Contributions and applications to survival analysis

Camila Fernandez

▶ **To cite this version:**

Camila Fernandez. Contributions and applications to survival analysis. Probability [math.PR]. Sorbonne Université, 2024. English. NNT : 2024SORUS230 . tel-04777087

HAL Id: tel-04777087
https://theses.hal.science/tel-04777087v1

Submitted on 12 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SORBONNE UNIVERSITÉ

Sorbonne Université

LPSM

École doctorale **École Doctorale Sciences Mathématiques de Paris Centre**

Unité de recherche **Laboratoire de Probabilités, Statistique et Modélisation**

Thèse présentée par **Camila Fernandez**

Soutenue le **12 juillet 2024**

En vue de l'obtention du grade de docteur de Sorbonne Université

Discipline **Mathématiques appliquées**

Spécialité **Statistiques**

Contributions and Applications to Survival Analysis

Thèse dirigée par	Olivier Wintenberger	directeur
	Pierre Gaillard	co-encadrant
	Chung Shue Chen	co-encadrant
	Alonso Silva	encadrant industriel

Composition du jury

Rapporteurs	Marianne Clausel	Université de Lorraine
	Francesca Pierri	Università degli Studi di Perugia
Examinateurs	Olivier Bouaziz	Université Paris Cité
	Elodie Brunel	Université de Montpellier
	Gregory Nuel	Sorbonne Université
Directeurs de thèse	Olivier Wintenberger	Sorbonne Université
	Pierre Gaillard	INRIA Grenoble
	Chung Shue Chen	Nokia Bell Labs
	Alonso Silva	Nokia Bell Labs

Colophon

Mémoire de thèse intitulé « Contributions and Applications to Survival Analysis », écrit par Camila Fernandez, achevé le 21 octobre 2024, composé au moyen du système de préparation de document LaTeX et de la classe yathesis dédiée aux thèses préparées en France.

Mots clés : analyse de survie, optimisation convexe en ligne, optimisation stochastique, apprentissage automatique, apprentissage en ligne

Keywords: survival analysis, online convex optimization, stochastic optimization, machine learning, online learning

Cette thèse a été préparée dans les laboratoires suivants.

Laboratoire de Probabilités, Statistique et Modélisation

Sorbonne Université
Campus Pierre et Marie Curie
4 place Jussieu
75005 Paris
France

☎ +33 1 57 27 93 16
Site `https://www.lpsm.paris/`

Nokia Bell Labs

Nokia France
12 rue Jean Bart
91300 Massy
France

Site `https://www.bell-labs.com/`

Inria

Centre Inria de l'Université Grenoble Alpes
655 Avenue de l'Europe
38334 Montbonnot CEDEX
France

Site `https://www.inria.fr/`

The right understanding of any matter
and a misunderstanding of the same
matter do not wholly exclude each other.

Franz Kafka

La exacta comprensión de una cosa y su
mala interpretación no se excluyen
totalmente.

Franz Kafka

Remerciements

Il y a trois ans et demi que cette aventure a commencé, une aventure remplie de joies et de déceptions, de rires et de larmes, de succès et de frustrations. Une aventure de trois ans et demi qui se termine maintenant, et où il ne me reste qu'à remercier toutes les personnes qui, de près ou de loin, ont fait partie de ce voyage. C'est un chapitre inoubliable qui englobe presque toute ma vie en France, qui m'a permis de grandir et d'apprendre et dont les souvenirs et les amitiés que j'ai forgés resteront pour toujours dans mon cœur. Merci à tous pour ces trois ans et demi de compagnie, d'affection, de soutien et de vie.

D'abord, je tiens à remercier Olivier Wintenberger, mon directeur de thèse, pour toute ta patience et ton dévouement. Pour m'avoir accueilli dans le laboratoire et m'avoir initiée au monde de la recherche. Cela a été un honneur de travailler avec toi, j'admire ta force académique et ta qualité humaine. Merci d'avoir toujours eu les mots justes et d'avoir dirigé ce navire lorsque nous pensions tous qu'il allait couler. Merci pour les invitations à Vienne et les dîners chez toi, pour les discussions de déjeuner et pour ta compréhension.

Pierre Gaillard, merci de toujours apporter de l'élégance à nos démonstrations, pour les cours sur l'empreinte carbone et les invitations à Grenoble. Merci pour ton immense soutien, tant dans les résultats mathématiques que dans mon bien-être personnel. Merci d'avoir géré l'extension de la thèse et pour tes mots d'encouragement.

Alonso Silva, thank you for trusting me with this project, for your commitment, and for not giving up even in the toughest times. Thank you also for welcoming me into the machine learning and survival analysis community, for all your coding advice and workshop suggestions, which have helped me grow as a professional and researcher.

Calvin Chen, thank you for welcoming me at Nokia and for your unwavering support and kindness. For your perseverance and meticulousness, for reading everything I write thoroughly, and for contributing the necessary rigor to ensure the accuracy and quality of our research.

I am grateful to the jury members, Francesca Pierri and Marianne Clausel, for taking the time to read my manuscript in detail and making valuable suggestions to improve it. I also include Elodie Brunel, Olivier Bouaziz, and Gregory Nuel for coming to listen to my presentation, for the interesting discussion, and for the rigorous evaluation of my work. Their expertise and observations have been essential for the enrichment of this work.

This thesis would not have been possible without Nokia Bell Labs, the laboratory that welcomed me when I was still an intern and did not know how to code in Python. Thanks to all the people I met in the corridors and with whom I shared meetings and lunches. Thank you, Fabio, for welcoming me in your team, and to all the other current and former members, Calvin, Alonso, Armen, Razane, Ayoub, Antonio, Francesco, Davide, Lou, Tianzhu, and Liuba, for making my time at Nokia infinitely enjoyable and fun.

In particular, there are two people who hold a special place in my heart. Gabriele, thank you for the Italian courses and guitar sessions, for sharing the cold office with me and always bringing joy, chocolate, and pizza into my life. Dalia, thank you for being my first friend on this journey and for including me in your circle as if we had known each other all our lives. Thank you for all the adventures and trips you have taken me on. I admire your strength, determination, and great heart, which have inspired me on several occasions. Thank you for all the shared ice cream and for our endless conversations that always leave me with some lesson about human life, adulthood, and love.

Merci à mes amis de l'équipe THOTH de l'INRIA à Grenoble, Paul, Jules, Julien et Pierre, d'avoir adapté leurs horaires à mes envies culinaires et de m'avoir suivi dans les diverses aventures que j'ai pu imaginer, en particulier, le parapente et le ski. Vous avez rendu tous mes voyages des moments de joie et de divertissement inoubliables.

Je veux aussi remercier toutes les personnes du LPSM qui ont participé directement et indirectement à la confection de cette thèse. Louise et Nathalie, pour leur énorme dévouement et l'affection avec laquelle elles s'occupent de tous nos besoins administratifs. Hugues, pour être le pilier qui maintient ce laboratoire en fonctionnement, merci de ne pas m'avoir banni du serveur lorsque j'ai occupé tous les CPU et d'être toujours disposé à résoudre mes doutes les plus naïfs. Merci aux permanents, Anna, Maxime, Claire, Erwan, Stéphane, Antoine, Arnaud, entre autres, pour leur gentillesse et pour créer l'ambiance propice à résoudre les mots croisés les plus diaboliques.

Merci à mes amis du LPSM, ceux qui sont partis et ceux qui restent. À Gloria, Niklas et Joseph, pour tous les déjeuners, les discussions et l'amour partagé pour le bon vin. À Francesco, Pierre, Alexis, Adeline, Yazid, Gabriel, Mathis, Eyal et Lucas, pour être les meilleurs compagnons dans cette traversée. À la nouvelle 'famille Wintenberger', Ferdinand, Paul, Grace, Antoine et Nina... et Romain, pour maintenir en vie le laboratoire et être toujours disponible pour écouter mes plaintes et sortir prendre un verre ou dîner le mythique 'poulet dégé' de Grace. Chez vous, j'ai trouvé un infini réconfort et beaucoup de compréhension, les éléments essentiels pour survivre aux années de thèse. J'espère que la vie continuera de nous réunir et nous réservera encore de nombreuses aventures.

Paul, merci d'être mon compagnon de voyages 2023, pour tout le fun que nous avons eu à Vienne et à Grenoble, de partager tes adresses de restaurants et ton enthousiasme à essayer de nouvelles choses, d'être mon professeur de ski, de toujours prendre soin de moi et de m'inclure dans ton grand cœur.

Antonio, merci pour nos conversations multilingues, d'être un fidèle compagnon de cinéma et pour tous les post-it avec des messages d'amour. De partager avec moi tes malaises et tes joies, pour tes câlins et tes mots d'encouragement et de motivation.

Merci à mes amis les plus fidèles qui m'ont accompagné tout au long de cette thèse, Miguel, Ariane, Iqraa et Ludovic. Pour la confiance et les confidences, pour les heures des ragots et de pause café. Vous êtes ce que j'ai de plus précieux de ces années de thèse ; j'ai apprécié chaque seconde à vos côtés, et c'est grâce à vous que cette expérience en a totalement valu la peine.

Miguel, merci de partager avec moi toutes tes histoires, d'apporter de la musique et de la bonne humeur à notre bureau, de m'encourager à rencontrer des gens et à sortir mon côté le plus sociable. Merci pour toutes les fêtes et les danses d'Anita ; le divertissement avec toi est toujours garanti. Merci pour toute la nourriture mexicaine et merci de partager nos âmes latines, toujours manquantes dans un laboratoire français.

Ariane, merci de m'inspirer avec ton 'sense of fashion', pour tes rires contagieux et ton sens de l'humour. De partager avec moi tes séries Netflix préférées et tes adresses à Paris et à Londres. Merci de me confier tes aventures, d'écouter les miennes attentivement et d'avoir toujours des mots de compréhension. Merci d'être venue de Londres pour ma soutenance de thèse et de continuer à me garder dans ta nouvelle vie.

Iqraa, merci pour ta douceur et ta compréhension avec laquelle tu écoutes encore et encore toutes mes histoires. Merci de te soucier de moi avec tant d'affection et pour toutes nos bêtises qui me ramènent un peu d'enfance. Nous sommes très différentes, mais nous parvenons à nous rencontrer et à nous comprendre profondément. *Mein tum se mohabbat karti hoon.*

Ludovic, merci pour tes adresses à l'intérieur et à l'extérieur du 10ème arrondissement, pour les cours quotidiens de français, les leçons de ski, de cuisine, de vélo et de politique française. Merci pour les voyages et les films d'auteur, d'essayer de me sensibiliser à la peinture de la Renaissance et de partager avec moi ta passion pour l'art et la musique. Merci d'être mon homme-portemanteau et d'attendre des heures le temps que je sois prête. Merci pour ton dévouement à notre relation et pour ton humour et légèreté qui allègent mes peurs et mes tristesses. Merci de me prendre la main au milieu du chaos et de me serrer dans tes bras la nuit lorsque je ne peux pas dormir. Merci de sécher mes larmes et d'enflammer mes joies avec des risas y sonrisas. Merci d'être mon compagnon à cette étape de la vie et de m'ouvrir les portes de ta maison, de ta culture et de ta vie. Merci pour ton immense soutien et compagnie qui rechauffent constamment mon cœur.

Merci à mes amis de l'extérieur du laboratoire, pour me faire sentir que la France est chez moi, pour tous les voyages de fin de semaine et les séances de sport. Catita, avec toi j'ai découvert une nouvelle amitié forgée par le quotidien, très semblable à la fraternité. Merci de supporter mon côté carré et autiste, pour toutes nos aventures et voyages, de me ramasser à la petite cuillère quand j'avais le cœur brisé et de m'ouvrir les portes de ta vie et de me présenter à tous tes amis. Cette thèse a été réalisée, en partie, grâce à ton soutien et ta compagnie. Merci, Gabriel, d'écouter volontairement et involontairement toutes mes histoires, de m'aider avec Lara et avec mes procédures administratives et économiques. Nico, merci de montrer de l'intérêt pour mes histoires les moins intéressantes, pour ton enthousiasme et ta motivation à explorer le monde et à faire du sport. J'admire ton intelligence, ta volonté et ton courage avec lesquels tu entreprends tous tes projets. Merci de m'inclure dans certains d'entre eux et pour tout les repas, le sport, les cocktails et les bières que nous avons encore à vivre. Yann, merci de m'avoir accueillie dans ta famille à plusieurs reprises. De partager avec moi un peu de ton enfant intérieur à travers tes jeus et tes blagues. Pour les heures passées a la bibli qui m'ont apporté du bonheur dans les jours d'étude. Pour tous les repas que nous nous sommes offerts, pour ta complicité et ton grand cœur qui m'ont réconforté à divers moments. Merci Adel, pour nos discussions interminables en prenant un verre, ou deux. De partager ta vision du monde et de l'amour. D'avoir toujours un argument juste et de me faire remettre en question mes propres pensées. Merci de m'écouter éternellement avec patience et compréhension. T'avoir rencontré et avoir construit cette vie ensemble est l'une des meilleures choses qui me soient arrivées en France. Merci aussi à tous mes autres amis, qui font partie de ma vie actuelle et avec qui j'ai passé des moments enrichissants : Rodrigo (Lara), Claire, Rodri, Alexis, Laurent, Valentin, Ewan, Martin, Edmond, Tony, Jean, Emilie, Cinzia, Jose, Cami (gringa), Nacho, parmi tant d'autres que je valorise immensément et qui m'ont soutenu de différentes manières durant cette période.

Merci à ma famille chilienne en France (les Chiliens à Paris), Isra, Claudio et Maka, de m'apporter de la chilenidad dans ma vie. Merci pour tous les completos et les panes con palta. Pour l'amour que vous me donnez et pour tous les rires et les fêtes. Votre compagnie stable me fait constamment sentir chez moi dans ce pays si loin de nos origines.

Merci à mes amies du lycée, Dani, Mai et Muriel, d'être mes sœurs de cœur et de continuer à grandir à mes côtés. Notre relation garde un morceau d'enfance et d'authenticité que je chéris profondément. Merci pour toutes les folies que nous avons imaginées ensemble et pour toutes celles qui nous restent encore à accomplir. Notre connexion surmonte toutes les barrières spatio-temporelles, et notre relation est ma plus belle histoire d'amour.

Je tiens à faire une mention spéciale aux amis de l'école de médecine de mon amie Muriel (y compris elle-même), pour avoir fourni la musique qui manquait à mes heures de sommeil et de démotivation, un véritable antidote antimiseria.

Merci à ma famille en général, qui m'a supportée tout au long de ma vie. Mes frères, mes parents, ma grand-mère, mes oncles, mes grands-oncles et mes cousins. De m'accepter telle que je suis (emmerdeuse dirait ma grand-mère). Pour tous les étés et les barbecues partagés. Pour tous les moments de joie et d'union qui font partie de mes souvenirs d'enfance et de mon identité. En particulier, je tiens à mentionner mes frères, pour les moments de jeux et de confidences. Nelsi, pour ta douceur et ta sensibilité, pour toujours écouter attentivement et offrir des câlins et de l'amour. Nandy, pour ta joie et ton énergie positive avec lesquelles tu affrontes la vie et que tu nous transmets. J'admire ton intelligence et ton courage avec lesquels tu poursuis tes rêves, je suis ta plus grande fan. Vala, les souvenirs de notre enfance ensemble sont mon trésor le plus précieux, je désire les temps où il n'y avait que toi et moi. Merci pour tes blagues incessantes, ta complicité et ta compagnie. Tu as été ma première meilleure amie et notre fraternité est la meilleure chose qui me soit arrivée dans la vie.

Enfin, je tiens à remercier mes parents. Papa, pour ton soutien inconditionnel, pour toujours m'encourager quand les choses ne se passent pas comme je le voudrais et pour ton humour stable qui me transmet la tranquillité dans les moments de chaos. J'admire l'humilité et la passion avec lesquelles tu affrontes tout ce que tu entreprends. Ton image et ton amour m'ont toujours poussée à être une meilleure personne et à poursuivre tous mes rêves. Et enfin, maman, merci pour ton amour inconditionnel et ton éducation. De m'enseigner les forces du travail et de la discipline. D'être l'exemple de la femme forte et indépendante. De m'inculquer l'amour de la lecture et de me montrer la liberté du savoir. Je suis qui je suis dans la partie la plus importante grâce à toi et je te dédie tous mes succès.

Agradecimientos

Hace tres años y medio que comenzó esta aventura, una aventura llena de alegrías y desilusiones, de risas y llantos, llena de éxito y de frustración. Una aventura de tres años y medio que llega a su fin y donde no me queda nada más que agradecer a todas las personas que desde la distancia y la cercanía espacio-temporal formaron parte de esta experiencia, un capítulo inolvidable que comprende casi toda mi vida en Francia, que me ha permitido crecer y aprender y cuyas memorias y amistades que he forjado perdurarán por siempre en mi corazón. Gracias a todos por estos tres años y medio de compañía, cariño, apoyo y vida.

En primer lugar, quiero agradecer a Olivier Wintenberger, mi director de tesis, por toda tu paciencia y dedicación. Por haberme acogido en el laboratorio y mostrarme el mundo de la investigación. Ha sido un honor para mí trabajar contigo, admiro tu fortaleza académica y tu calidad humana. Gracias por siempre tener las palabras justas y por haber liderado este barco cuando todos creímos que se iba a hundir. Gracias por las invitaciones a Viena y las cenas en tu casa, por las discusiones de almuerzo y por tu comprensión.

Pierre Gaillard, gracias por siempre aportar elegancia a nuestras demostraciones, por las clases sobre la huella de carbono y las invitaciones a Grenoble. Gracias por tu enorme apoyo tanto en los resultados matemáticos como en mi bienestar personal. Gracias por gestionar la extensión de la tesis y por tus palabras de aliento.

Alonso Silva, gracias por haber confiado en mí para este proyecto, por tu compromiso y por no rendirte incluso en los peores momentos. Gracias también por abrirme las puertas a la comunidad de machine learning y de survival analysis, por todos tus consejos de código y sugerencias de seminarios y workshops, los cuales me hicieron crecer como profesional e investigadora.

Calvin Chen, gracias por haberme acogido en Nokia y por tu incondicional apoyo y gentileza. Por tu constancia y meticulosidad, por leer minuciosamente todo lo que escribo y por aportar la rigurosidad clave para garantizar la precisión y la calidad de nuestra investigación.

Agradezco a los miembros del jurado, Francesca Pierri y Marianne Clausel, por haber tomado el tiempo de leer detalladamente mi manuscrito y hacer valiosas sugerencias para mejorarlo. Incluyo también a Elodie Brunel, Olivier Bouaziz y Gregory Nuel por haber venido a escuchar mi presentación, por la interesante discusión y por la evaluación rigurosa de mi trabajo. Su experticia y observaciones han sido esenciales para el enriquecimiento de este trabajo.

Esta tesis no habría sido posible sin Nokia Bell Labs, el laboratorio que me acogió cuando aún era practicante y no sabía programar en Python. Gracias a todas las personas que crucé en los pasillos y con quienes compartí reuniones y almuerzos. Gracias, Fabio, por haberme bienvenido en tu equipo, y a todos los otros miembros y ex-miembros, Calvin, Alonso, Armen, Razane, Ayoub, Antonio, Francesco, Davide, Lou, Tianzhu y Liuba por hacer mi paso por Nokia infinitamente ameno y disfrutable.

En particular, hay dos personas que tienen un lugar especial en mi corazón. Gabriele, gracias por los cursos de italiano y las sesiones de guitarra, por compartir la fría oficina conmigo y siempre aportar alegría, chocolate y pizza a mi vida. Dalia, gracias por haber sido mi primera amiga en esta travesía y haberme incluido en tu círculo como si nos conociéramos de toda la vida. Gracias por todas las aventuras y viajes en los que me has embarcado. Admiro tu fuerza, tu determinación y tu gran corazón que me han inspirado en diversas ocasiones. Gracias por todo el helado compartido y por nuestras interminables conversaciones que siempre me dejan alguna enseñanza sobre la vida humana, la adultez y el amor.

Gracias a mis amigos del equipo THOTH de INRIA en Grenoble, Paul, Jules, Julien y Pierre, por adaptarse a mis horarios y antojos de comida y por seguirme en las diversas aventuras que alguna vez se me ocurrieron, particularmente, el parapente y el ski. Ustedes hicieron de todos mis viajes un momento de alegría y diversión inolvidables.

Quiero también agradecer a todas las personas del LPSM que participaron directa e indirectamente en la confección de esta tesis. Louise y Nathalie, por su enorme dedicación y cariño con el que se ocupan de todas nuestras necesidades administrativas. Hugues, por ser el pilar que mantiene funcionando este laboratorio, gracias por no haberme baneado del servidor cuando ocupé todas las CPU y por siempre estar dispuesto a resolver mis más ingenuas dudas. Gracias a 'los permanentes', Anna, Maxime, Claire, Erwan, Stéphane, Antoine, Arnaud, entre otros, por su amabilidad y por crear el ambiente propicio para resolver los más endiablados puzzles.

Gracias a mis amigos del LPSM, los que ya se fueron y los que aún quedan. A Gloria, Niklas y Joseph, por todos los almuerzos, discusiones y el compartido amor por el buen vino. A Francesco, Pierre, Alexis, Adeline, Yazid, Gabriel, Mathis, Eyal y Lucas, por ser los mejores compañeros en esta travesía. A la nueva 'familia Wintenberger', Ferdinand, Paul, Grace, Antoine y Nina... y Romain, por mantener con vida el laboratorio y por siempre estar disponible para escuchar mis quejas y salir a tomar un trago o cenar el mítico 'poulet dégé' de Grace. En todos ustedes he encontrado infinito consuelo y comprensión, los elementos esenciales para sobrevivir los años de tesis. Espero que la vida nos siga encontrando y que nos depare aún muchas aventuras.

Paul, gracias por ser mi compañero de viajes 2023, por toda la diversión que tuvimos en Viena y en Grenoble, por compartir tus addresses de restaurantes y el entusiasmo por probar cosas nuevas, por ser mi profesor de ski, por siempre cuidar de mí y por hacerme un espacio en tu gran corazón.

Antonio, gracias por nuestras conversaciones multilingüísticas, por ser un leal compañero de cine y por todos los post-it con mensajes de amor. Por compartir conmigo tus malestares y alegrías, por tus abrazos y por tus palabras de aliento y motivación.

Gracias a mis amigos más fieles que me acompañaron a lo largo de toda esta tesis, Miguel, Ariane, Iqraa y Ludovic. Por la confianza y las confidencias, por las horas de chisme y de pause café. Ustedes son lo más importante que me dejó estos años de tesis, disfruté cada segundo a su lado y es gracias a ustedes que esta experiencia valió completamente la pena.

Miguel, gracias por compartir conmigo todas tus historias, por aportar música y buen humor a nuestra oficina, por alentarme a conocer gente y sacar mi lado más sociable. Gracias por todas las fiestas y los bailes de Anita ; la diversión contigo siempre está garantizada. Gracias por toda la comida mexicana y por compartir nuestras almas latinas, siempre en falta en un laboratorio francés.

Ariane, gracias por inspirarme con tu 'sense of fashion' y contagiarme con tus infinitas risas y sentido del humor. Por compartir conmigo tus series de Netflix preferidas y tus addresses en París y en Londres. Gracias por confiarme tus aventuras y escuchar las mías con atención, y por siempre tener palabras de comprensión. Gracias por venir desde Londres a mi defensa de tesis y por continuar teniéndome presente en tu nueva vida.

Iqraa, gracias por tu dulzura y comprensión con la que escuchas una y otra vez todas mis historias. Gracias por preocuparte por mí con tanto cariño y por todas nuestras travesuras que me traen un poco de niñez. Somos muy diferentes pero logramos encontrarnos y entendernos profundamente. *Mein tum se mohabbat karti hoon.*

Ludovic, gracias por tus addresses dentro y fuera del 10ème arrondissement, por los cursos diarios de francés, las lecciones de ski, cocina, bicicleta y política francesa. Por los viajes y las películas de autor, por intentar sensibilizarme a la pintura renacentista y por compartir conmigo tu pasión por el arte y la música. Gracias por ser mi hombre-perchero y esperar mil horas hasta que esté lista. Gracias por tu dedicación a nuestra relación, y por tu humor y ligereza que alivianan mis miedos y tristezas. Gracias por tomar mi mano en medio del caos y abrazarme en la noche cuando no puedo dormir. Gracias por secar mis lágrimas y avivar mis alegrías con risas y sonrisas. Gracias por ser mi compañero en esta etapa de la vida y por abrirme las puertas de tu casa, de tu cultura y de tu vida. Gracias por tu enorme apoyo y compañía que me llenan constantemente el corazón.

Gracias a mis amigos de fuera del laboratorio, por hacerme sentir que Francia es mi hogar, por todos los viajes de fin de semana y las sesiones de deporte. Catita, contigo descubrí una amistad nueva forjada por la cotidianidad, muy parecida a la hermandad. Gracias por soportar mi lado cuadrado y autista, por todas nuestras aventuras y viajes, por recogerme con una cuchara cuando tenía el corazón roto y por abrirme las puertas de tu vida y presentarme a todos tus amigos. Sin duda esta tesis fue lograda, en parte, gracias a tu apoyo y compañía. Gracias, Gabriel, por escuchar voluntaria e involuntariamente todas mis historias, por ayudarme con Lara y con mis procedimientos administrativos y económicos. Nico, gracias por demostrar interés por la más desinteresante de mis historias, por tu entusiasmo y motivación por salir a explorar el mundo y hacer deporte. Admiro tu inteligencia, decisión y coraje con los que emprendes todos tus proyectos. Gracias por incluirme en algunos de ellos y por toda la comida, deporte, cocktails y cervezas que aún nos quedan por vivir. Yann, gracias por haberme acogido en tu familia en múltiples ocasiones. Por compartir conmigo un poco de tu niño interior a través de tus juegos y bromas. Por las horas en la biblioteca que me trajeron felicidad en los días de estudio. Por todas las comidas que nos invitamos, por tu complicidad y gran corazón que me han reconfortado en diversos momentos. Gracias Adel, por nuestras inacabables discusiones tomando un trago, o dos. Por compartirme tu visión del mundo y del amor. Por siempre tener un argumento justo y hacerme cuestionar mis propios pensamientos. Gracias por eternamente escucharme con paciencia y comprensión. Haberte encontrado, y haber construido esta vida juntos es una de las mejores cosas que me pasó en Francia. Gracias a todos los demás amigos, que forman parte de mi vida actual y con los que he pasado momentos enriquecedores : Rodrigo (Lara), Claire, Rodri, Alexis, Laurent, Valentin, Ewan, Martin, Edmond, Tony, Jean, Emilie, Cinzia, Jose, Cami (gringa), Nacho, entre muchos otros que valoro inmensamente y que me han apoyado de distintas formas en esta aventura.

Gracias a mi familia chilena en Francia (les chiliens à Paris), Isra, Claudio y Maka, por traer chilenidad a mi vida. Gracias por todos los completos y los pan con palta. Por el amor que me entregan y por todas las risas y las fiestas. Su estable compañía me hace sentir constantemente en casa en este país tan lejos de nuestros orígenes.

Gracias a mis amigas del liceo, Dani, Mai y Muriel, por ser mis hermanas del alma y por continuar creciendo a mi lado. Nuestra relación guarda un pedazo de niñez y autenticidad que atesoro profundamente. Gracias por todas las locuras que hemos ideado juntas y por todas las que nos quedan aún por cumplir. Nuestra conexión supera todas las barreras espacio-temporales y nuestra relación es lejos mi mejor historia de amor.

Quiero hacer una mención especial a los amigos de la escuela de medicina de mi amiga Muriel (incluida), por haber proporcionado la música que le hacía falta a mis horas de sueño y desmotivación, un verdadero antídoto antmiseria.

Gracias a mi familia en general, que me han soportado a lo largo de toda mi vida. Mis hermanos, mis padres, mi abuela, mis tíos, mis tíos abuelos y mis primos. Por aceptarme tal cual soy (jodida diría mi abuela). Por todos los veranos y los asados compartidos. Por todos los momentos de alegría y unión que forman parte de mis memorias de infancia y de mi identidad. En especial, quiero mencionar a mis hermanos, por los momentos de juegos y confidencias. Nelsi, por tu dulzura y sensibilidad, por siempre escuchar con atención y ofrecer abrazos y amor. Nandy, por tu alegría y energía positiva con la que enfrentas la vida y que nos transmites. Admiro tu inteligencia y coraje con el que persigues tus sueños, soy tu más grande fan. Vala, los recuerdos de nuestra infancia juntas son mi más valioso tesoro, anhelo los tiempos en que solo existíamos tú y yo. Gracias por tus incesantes bromas, complicidad y compañía. Fuiste mi primera mejor amiga y nuestra hermandad es lo mejor que me ha pasado en la vida.

Finalmente, quiero agradecer a mis padres. Papá, por tu apoyo incondicional, por siempre darme ánimos cuando las cosas no salen como me gustarían y por tu humor estable que me transmite tranquilidad en momentos de caos. Admiro la humildad y pasión con la que enfrentas todo lo que emprendes. Tu imagen y amor siempre me han impulsado a ser mejor persona y a perseguir todos mis sueños. Y por último, mamá, gracias por tu incondicional amor y educación. Por enseñarme las fortalezas del trabajo y la disciplina. Por ser el ejemplo de mujer fuerte e independiente. Por inculcarme el amor por la lectura y mostrarme la libertad del conocimiento. Soy quien soy en la parte más importante gracias a ti y te dedico todos mis logros.

Contributions and Applications to Survival Analysis

Résumé

L'analyse de survie a suscité l'intérêt de diverses disciplines, allant de la médecine et de la maintenance prédictive à diverses applications industrielles. Sa popularité croissante peut être attribuée aux avancées significatives en matière de puissance de calcul et à la disponibilité accrue des données. Des approches variées ont été développées pour répondre au défi des données censurées, allant des outils statistiques classiques aux techniques contemporaines d'apprentissage automatique. Cependant, il reste encore une marge considérable pour l'amélioration. Cette thèse vise à introduire des approches innovantes qui fournissent des insights plus profonds sur les distributions de survie et à proposer de nouvelles méthodes avec des garanties théoriques qui améliorent la précision des prédictions.

Il est notamment remarquable de constater l'absence de modèles capables de traiter les données séquentielles, une configuration pertinente en raison de sa capacité à s'adapter rapidement à de nouvelles informations et de son efficacité à gérer de grands flux de données sans nécessiter d'importantes ressources mémoire. La première contribution de cette thèse est de proposer un cadre théorique pour la modélisation des données de survie en ligne. Nous modélisons la fonction de risque comme une exponentielle paramétrique qui dépend des covariables, et nous utilisons des algorithmes d'optimisation convexe en ligne pour optimiser la vraisemblance de notre modèle, une approche qui est novatrice dans ce domaine. Nous proposons un nouvel algorithme adaptatif de second ordre, SurvONS, qui assure une robustesse dans la sélection des hyperparamètres tout en maintenant des bornes de regret rapides. De plus, nous introduisons une approche stochastique qui améliore les propriétés de convexité pour atteindre des taux de convergence plus rapides.

La deuxième contribution de cette thèse est de fournir une comparaison détaillée de divers modèles de survie, incluant les modèles semi-paramétriques, paramétriques et ceux basés sur l'apprentissage automatique. Nous étudions les caractéristiques des ensembles de données qui influencent la performance des méthodes, et nous proposons une procédure d'agrégation qui améliore la précision et la robustesse des prédictions. Enfin, nous appliquons les différentes approches discutées tout au long de la thèse à une étude de cas industrielle : la prédiction de l'attrition des employés, un problème fondamental dans le monde des affaires moderne. De plus, nous étudions l'impact des caractéristiques des employés sur les prédictions d'attrition en utilisant l'importance des caractéristiques par permutation et les valeurs de Shapley.

Mots clés : analyse de survie, optimisation convexe en ligne, optimisation stochastique, apprentissage automatique, apprentissage en ligne

Laboratoire de Probabilités, Statistique et Modélisation
Sorbonne Université – Campus Pierre et Marie Curie – 4 place Jussieu – 75005 Paris – France

Abstract

Survival analysis has attracted interest from a wide range of disciplines, spanning from medicine and predictive maintenance to various industrial applications. Its growing popularity can be attributed to significant advancements in computational power and the increased availability of data. Diverse approaches have been developed to address the challenge of censored data, from classical statistical tools to contemporary machine learning techniques. However, there is still considerable room for improvement. This thesis aims to introduce innovative approaches that provide deeper insights into survival distributions and to propose new methods with theoretical guarantees that enhance prediction accuracy. Notably, we notice the lack of models able to treat sequential data, a setting that is relevant due to its ability to adapt quickly to new information and its efficiency in handling large data streams without requiring significant memory resources. The first contribution of this thesis is to propose a theoretical framework for modeling online survival data. We model the hazard function as a parametric exponential that depends on the covariates, and we use online convex optimization algorithms to minimize the negative log-likelihood of our model, an approach that is novel in this field. We propose a new adaptive second-order algorithm, SurvONS, which ensures robustness in hyperparameter selection while maintaining fast regret bounds. Additionally, we introduce a stochastic approach that enhances the convexity properties to achieve faster convergence rates.
The second contribution of this thesis is to provide a detailed comparison of diverse survival models, including semi-parametric, parametric, and machine learning models. We study the dataset characteristics that influence the methods performance, and we propose an aggregation procedure that enhances prediction accuracy and robustness. Finally, we apply the different approaches discussed throughout the thesis to an industrial case study: predicting employee attrition, a fundamental issue in modern business. Additionally, we study the impact of employee characteristics on attrition predictions using permutation feature importance and Shapley values.

Keywords: survival analysis, online convex optimization, stochastic optimization, machine learning, online learning

Table des matières

Chapitre 1

Introduction

1.1 Outline of the Thesis

The main goal of this thesis is to study different aspects of survival analysis, a statistical field that focuses on analyzing the time until an event of interest occurs. We aim to explore the use of online convex optimization algorithms, their robustness, and convergence guarantees in the context of censored data. Additionally, we want to study a stochastic approach that enhances the scope of regret analysis. Furthermore, we intend to focus on industrial applications of survival analysis tools, investigating both parametric and machine learning methods, as well as engaging in a discussion on scoring rules.

This thesis was prepared under a CIFRE contract with Nokia Bell Labs and in collaboration with Inria. Chapter 1 is dedicated to the introduction of the different subjects addressed in the thesis. Initially, we briefly explain what survival analysis is, including its historical approaches, and provide an overview of existing methods, comprising both statistical and machine learning approaches, complemented by an example using a classical dataset. Subsequently, we introduce scoring rules, an important domain within survival analysis. Additionally, we discuss the online setting and illustrate the challenges of selecting learning rates. Later, we present the problem of attrition prediction. Finally, we detail the explicit contributions of the thesis.

Chapter 2 presents the main mathematical contribution of the thesis. We introduce a detailed online setting for survival analysis and propose the use of online convex optimization (OCO) techniques to estimate a parametric survival function. In the context of OCO algorithms, we focus on the regret analysis and we identify issues with the Online Newton Step (ONS) regret bound when selecting the last known optimal learning rate. In Section 2.4, we present a stochastic setting that enables us to prove Theorem 2, which establishes logarithmic stochastic regret bounds. Additionally, we prove Corollary 1, which guarantees the convergence of the algorithm predictions to the optimal parameter. In the deterministic setting, we propose an aggregation algorithm, SurvONS 1, that wisely adapts the learning rate to enhance robustness while preserving fast convergence rates. This result is summarized in Theorem 3, which explicitly provides the algorithm regret bound. More details are provided in Section 1.5.

— Fernandez, C., Gaillard, P., de Vilmarest, J. and Wintenberger, O. (2024). Online learning approach for suvival analysis. *arXiv preprint arXiv :2402.05145*.

In Chapter 3, we conduct a detailed comparison of parametric and machine learning methods using two different scoring rules, and across three datasets. Additionally, we study the impact of optimizing the methods hyperparameters. Our goal is to understand which factors most significantly influence the performance of these methods, noting that they rank differently across datasets and when changing the evaluation score. To this end, we propose an experiment with simulated data to deepen the insights from the previous comparison. Finally, we suggest aggregating the methods by optimizing the parameters of a convex combination such that it minimizes the integrated Brier score (see Algorithm 2). This aims to enhance robustness in model performance and ensure consistent overall accuracy.

— Fernandez, C., Chen, C.S., Gaillard, P. and Silva, A. (2024). Experimental Comparison of Ensemble Methods and Time-to-Event Analysis Models Through Integrated Brier Score and Concordance Index. *arXiv preprint arXiv :2403.07460*.

In Chapter 4, we present an industrial application of survival analysis, focusing on the prediction of employee attrition. We implement and compare multiple survival analysis methods, including those proposed in the previous chapters. Additionally, we analyze the effect of covariates on the performance of the methods using two different strategies : permutation feature importance and Shapley values.

1.2 Background on Survival Analysis

1.2.1 Framework

We consider a group of subjects, also called individuals, and an event that occurs after a certain period. This event could be the failure of a machine, the recurrence of a disease, customer churn, the lifetime of a specific population, or employee attrition, among other examples. We assume that the event occurs at most once for any given subject. Additionally, we consider an arrival time, which could represent their admission to a hospital, the purchase date of a machine, or an employee's start time at a company, etc. The goal of survival analysis, and consequently of this thesis, is to predict the length of time until the event occurs.

Let us consider a homogeneous population of individuals, an event time, which is a single non-negative random variable T, and an arrival time, another non-negative random variable τ such that $T > \tau$. Each individual is associated with a given vector of characteristics $x \in \mathbb{R}^d$ of dimension $d > 0$, also known as explanatory variables, which can represent different attributes thought to influence survival. We consider each component of this vector to be a real, continuous number. Our aim is to understand the underlying distribution of T given the joint effect of x.

This setting is widely considered as a univariate rather than a multivariate technique due to the presence of a single response variable, the event time, despite the existence of multiple explanatory variables. Problems involving multivariate responses are discussed in Cox and Oakes [25].

The distribution of event times

In order to understand and describe the distribution of T we need to consider various functions. For each individual, we define the hazard function, representing the current risk of experiencing the event at time t given the individual has survived until then :

$$h(t|x) = \lim_{\Delta \to 0^+} \frac{\mathbb{P}(t \leqslant T \leqslant t + \Delta | T \geqslant t, x)}{\Delta}, \quad t \geqslant 0, x \in \mathbb{R}^d.$$

The hazard function measures at each time t, the tendency to experience the event time in the near future, thereby capturing the underlying dynamics of survival. We use the hazard function to study the distribution of survival times, specifically, we estimate the hazard function by assuming either a specific parametric or non-parametric model. Let us remark that the distribution of T also depends on τ, during this section we consider $\tau = 0$ and therefore we omit it in the notation of the probability law. We concentrate on modeling the hazard function, but other approaches are also possible, such as Fleming and Harrington [40], who model the problem as a counting stochastic process.

The hazard function is strongly related to the survival function, which represents the probability that an individual survives beyond a certain time t. More formally we define :

$$S(t|x) = \mathbb{P}(T \geqslant t|x), \quad t \geqslant 0, x \in \mathbb{R}^d,$$

the probability to not experience the event until time t. This function is the complement of the distribution function $F(t|x) = \mathbb{P}(T \leqslant t|x)$. In some cases, we will use the survival function to model the event time distributions. The mathematical relationship between the survival function $S(t|x)$ and the hazard function $h(t|x)$ is given as follows :

$$h(t|x) = -\frac{\partial \log(S(t|x))}{\partial t}, \quad t \geqslant 0, x \in \mathbb{R}^d.$$

This formulation allows us to deduce the hazard function when we have the survival function, and vice versa. Another important function in probability is the density :

$$f(t|x) = \lim_{\Delta \to 0^+} \frac{\mathbb{P}(t < T < t + \Delta | x)}{\Delta} \quad t \geqslant 0, x \in \mathbb{R}^d,$$

which is directly related to the hazard function and the survival probability as $f(t|x) = S(t|x)h(t|x)$. And finally, we define the cumulative risk :

$$\Lambda(t|x) = \int_0^t h(s|x)ds \quad t \geqslant 0, x \in \mathbb{R}^d.$$

In this thesis, we assume that the survival time is continuous, suggesting that we model the event as it can occur at any time point. However, time measurements in real-world scenarios are discrete, often recorded in days, months, or years, requiring a time discretization for practical applications. More details on continuous survival time can be found in Klein and Moeschberger [81], and on discrete time in Tutz [129].

Censorship

The main challenge of survival analysis is considering censored data. In many cases, the study might end before all individuals experience the event, or some might withdraw before the study concludes, leading to incomplete information. This scenario is referred to as right censoring and right censored individuals can represent a significant percentage of the entire dataset ; discarding them may lead to an underestimation of event times and introduce bias. In addition, there are two other types of censorship : Left censoring, which occurs when the arrival time is not observed for a proportion of individuals, and interval censoring, where the exact time of the event is not known, but it is known to occur within a certain interval. Throughout this thesis, we focus exclusively on the phenomenon of right censoring, which we will simply refer to as 'censoring'.

We consider the censored time as a non-negative random variable C, which is independent of T. We define $U = \min\{T, C\}$ the observed time, and $\delta = \mathbb{1}\{T \leqslant C\}$ the event indicator. Survival analysis addresses the challenge of modeling the underlying dynamics of survival times while considering censorship, i.e., the challenge of estimating the distribution of T when we only know the realizations of U, making the problem particularly hard to solve.

1.2.2 Existing methods

Non-parametric statistical methods

One of the oldest tools for analyzing survival in homogeneous populations is the life table, usually attributed to John Graunt [48]. This model uses a discrete-time framework, and therefore, we consider a discretization of time into $n > 0$ intervals $[0, a_1), ..., [a_{n-1}, a_n)$. We define $\mathtt{r}_t$ as the number of individuals at risk at the interval $[a_{t-1}, a_t)$, where 'at risk' means that the individuals have already arrived at the study and have not yet experienced the event or been censored. Additionally, we define $\mathtt{d}_t$ as the number of individuals experiencing the event at the interval $[a_{t-1}, a_t)$, and $\mathtt{c}_t$ as the number of individuals being censored during the interval. Let us note that in this case, t is used both as the index of the intervals and to denote the time variable in the hazard function. Life tables are characterized by their approach of not considering explanatory variables in the hazard function, which means the discrete function is expressed as :

$$h(t) = \mathbb{P}(T \in [a_{t-1}, a_t) | T \geqslant a_{t-1}), \quad t \geqslant 0.$$

The standard life table estimator of the hazard function is :

$$\widehat{h}(t) = \frac{\mathtt{d}_t}{\mathtt{r}_t - \mathtt{c}_t/2}, \quad t \geqslant 0,$$

where censoring is assumed to occur at the middle of the interval, a compromise between considering censoring at the end of the interval ($\widehat{h}(t) = \mathtt{d}_t/\mathtt{r}_t$) and at the beginning ($\widehat{h}(t) = \mathtt{d}_t/(\mathtt{r}_t - \mathtt{c}_t)$). An example of this approach can be seen in Fahrmeir et al. [35] who studied the duration of German unemployment. In addition, Lawless [85] and Greedwood [53] studied the distributional aspects of life tables and Hastie and Loader [61] and Loader [90] proposed different smoothing techniques.

Later, in 1958, Kaplan and Meier [76] proposed a non-parametric estimator of the survival probability based on the usual binomial estimate that computes at each time the proportion of survivors. This estimator assumes that time is observed on a continuous scale and it is used

in applications to compare estimates based on grouped data. Let us consider a population of individuals with their N ordered event times $t_0 < t_1 < \ldots < t_N$, which are realizations of T. For each event time t_i, there are associated $\mathtt{r}_i$, the number of people at risk during $[t_{i-1}, t_i)$, and $\mathtt{d}_i$, the number of events at time t_i. Additionally, it is important to note that not all individuals experience the event ; however, the estimator exclusively considers the times at which an event occurred. The Kaplan-Meier estimator is defined as follows :

$$\widehat{S}(t) = \prod_{i:t_i<t} \left(1 - \mathtt{d}_i/\mathtt{r}_i\right), \quad t \geqslant 0. \tag{1.1}$$

We notice that this estimator uses the proportion of observed events to the number of people at risk $\mathtt{d}_i/\mathtt{r}_i$, similar to the life table estimator when censoring is assumed to occur at the end of the interval. This estimator has been widely studied (see [105], [49] and [22]) and it remains a useful tool in this field.

Covariates effect

One of the first attempt to consider different characteristics of the individuals in the modeling of the survival function is the two-sample problem, which consists in considering two groups with different attributes, for example, patients with some disease following two different treatments. Example of this can be found in Mantel and Haenszel [96] and in Gehan [46], and later in its posterior extensions from Mantel [95] and Efron [33]. The limitation of these models is that they are restricted to the case in which there are not many possible combination of covariates. We illustrate this issue with a real dataset in Section 1.2.3.

One way to specify the link between explanatory variables and the survival model is to choose a parametrization of the hazard function. Generalized linear models framework offers a wide variety of models for binary data (see McCullagh and Nelder [97], Tutz [128] and Agresti [3]). The best known is the logit model that proposes to write the discrete hazard function as a logistic regression :

$$h(t|x) = \frac{\exp(\theta^\top x)}{1 + \exp(\theta^\top x)}, \quad \theta \in \Theta, t \geqslant 0, x \in \mathbb{R}^d,$$

where $\Theta \subseteq \mathbb{R}^d$ is the parametric family. This model is also known as the proportional continuation ratio model (Agresti [3]). Alternative models can be considered when describing the discrete hazard function, such as Gompertz [51], Gumbel, Probit or exponential distributions (see Tutz [129] for more details).

The most widely used continuous-time model was proposed by Cox [24] in 1972 and consist on writing the hazard as a multiplication of a non-parametric baseline function that depends on the time and an exponential function that contains the covariates effects :

$$h(t|x) = h_0(t) \exp(\theta^\top x), \quad \theta \in \Theta, t \geqslant 0, x \in \mathbb{R}^d, \tag{1.2}$$

where h_0 is the non-parametric baseline function. This model assumes that the ratio of the hazard rates is proportional, a condition that may not hold in some real datasets and can be seen as restrictive. However, its semi-parametric nature simplifies the analysis by eliminating the need to specify the form of the baseline hazard function. This characteristic is particularly beneficial for analyzing survival data without making prior assumptions about the distribution of event times. The Cox model estimates the parameter θ using the partial likelihood method, which maximizes the likelihood of the observed survival times. In addition, we remark the similarity of this model

TABLEAU 1.1 – Overview of statistical methods used in survival analysis, highlighting differences in their handling of covariates, time data types, and the specific estimators or models employed.

Method	Covariates	Time	Estimator/model
Standard life table	no	discrete	$\widehat{h}(t) = \frac{\mathtt{d}_t}{\mathtt{r}_t - \mathtt{c}_t/2}$
Kaplan-Meier	no	continuous	$\widehat{S}(t) = \prod_{i:t_i<t}(1 - \mathtt{d}_i/\mathtt{r}_i)$
Logit model	yes	discrete	$h(t\|x) = \frac{\exp(\theta^\top x)}{1+\exp(\theta^\top x)}$
Cox proportional hazards	yes	continuous	$h(t\|x) = h_0(t)\exp(\theta^\top x)$

to generalized linear models (GLM). Cox proportional hazards can be seen as a semi-parametric GLM where the exponential has the role of the link function (more details in McCullagh and Nelder [97]).

Several extensions of the Cox model have been developed to address more complex survival data structures. For instance, competing risks models [93] extend the Cox model to account for scenarios where individuals can experience one of several different types of events. This extension allows for the estimation of cause-specific hazard functions, providing a more detailed analysis in settings where events compete to occur. Another important extension is the multistate model [109], which generalizes the Cox model to handle transitions between multiple states over time. This model is particularly useful in clinical settings where patients can move through various health states, such as remission, relapse, or recovery. Both extensions retain the semi-parametric nature of the original Cox model while allowing for more flexible and realistic modeling of complex event structures. For more details on the Cox model and its possible extensions see Therneau and Grambsch [126].

Many other parametric and semi-parametric approaches exist, including the Weibull Accelerated Failure Time (AFT) model [136], the Aalen additive model [2], the log-normal model [113], among others. These methods will be discussed in detail in Chapter 3 and a resume table can be found in Tableau 1.1

Machine learning approaches

Nowadays, the importance of machine learning methods is increasing due to the significant advances in computing power and data availability. Survival analysis, traditionally reliant on statistical models, is also embracing these advancements. By incorporating machine learning, survival analysis can now tackle complex covariate interactions, manage high-dimensional data, and model non-linearity more effectively.

Machine learning models, initially designed for tabular data, have been adapted to handle censored data and to accommodate to multiple complex scenarios. The pioneering approach by Faraggi and Simon [36] introduced the use of neural networks to model survival data. This method served as a non-linear extension of the Cox proportional hazards model. Another popular technique in machine learning is boosting, designed to improve the accuracy of predictions by combining multiple weaker learners to create a strong predictive model. Examples of boosting techniques in survival analysis include the works of Binder et al. [11] and Ridgeway [112], which utilize Cox-type losses. Additionally, other authors, such as Baoshan et al. [94], have proposed adaptations of XGBoost [21] to survival data.

In 2008, Ishwaran et al. introduced Random Survival Forest [72], adapting the Random Forest method [15] for censored data. This adaptation involves using specific splitting criteria, like the

log-rank test, to better separate survival times across different groups. Recently, deep learning techniques like Deep Survival Analysis [111] and DeepSurv [77] have gained attention in the survival analysis community. These approaches use deep neural networks to optimize a non-linear loss derived from the partial likelihood of the Cox proportional hazard model, incorporating an additional regularization term. Thus, with the objective to provide personalized treatment recommendations based on medical data. Another method is DNNSurv [142], which simplifies survival analysis by transforming survival times into pseudo probabilities used as response variables in a neural network, effectively reducing the complexity to a standard regression problem. Additionally, DeepHit [87] emerges as a powerful tool capable of managing competing risks, a setting in which we consider several types of events that can influence the occurrence of the other events, without relying on the assumptions of the Cox model or any specific assumptions about the underlying stochastic process. A review of deep learning for survival analysis was proposed by Wiegrebe et al. [137].

Some machine learning approaches and its advantages will be discussed with details in Chapter 3. For more details on machine learning techniques in the context of survival analysis see Wang et al. [135] or Sonabend [122]. In the following section, we show an illustrative example using a well-known dataset.

1.2.3 Illustrating covariate effects with clinical data

In Chapter 3, we use the primary biliary cirrhosis (PBC) dataset to assess the performance of various models and to compare the effectiveness of our ensemble method. In this section, we provide some details about the PBC dataset, which is one of the most classical examples of survival data, made available by Therneau and Grambsch [125]. It presents the collected data of a study on the efficacy of using D-penicillamine as a treatment for primary biliary cirrhosis. The dataset consists of $N = 276$ patients with PBC, and the objective is to predict their lifetime. By the end of the data collection period, $59,8\%$ of the patients were still alive, leading to a significant percentage of censoring.

	trt	age	sex	ascites	hepato	spiders	edema	bili	chol	albumin	copper	alk.phos	ast	trig	platelet	protime	stage
0	1.0	58.765229	f	1.0	1.0	1.0	1.0	14.5	261.0	2.60	156.0	1718.0	137.95	172.0	190.0	12.2	4.0
1	1.0	56.446270	f	0.0	1.0	1.0	0.0	1.1	302.0	4.14	54.0	7394.8	113.52	88.0	221.0	10.6	3.0
2	1.0	70.072553	m	0.0	0.0	0.0	0.5	1.4	176.0	3.48	210.0	516.0	96.10	55.0	151.0	12.0	4.0
3	1.0	54.740589	f	0.0	1.0	1.0	0.5	1.8	244.0	2.54	64.0	6121.8	60.63	92.0	183.0	10.3	4.0
4	2.0	38.105407	f	0.0	1.0	1.0	0.0	3.4	279.0	3.53	143.0	671.0	113.15	72.0	136.0	10.9	3.0

FIGURE 1.1 – Clinical characteristics of patients with primary biliary cirrhosis (PBC dataset).

Each individual is associated with a vector of 17 characteristics, as shown in rows in Figure 1.1. The types of characteristics vary, ranging from age and sex to multiple biological indicators. These characteristics can be either categorical or numerical.

	status	time
0	True	400.0
1	False	4500.0
2	True	1012.0
3	True	1925.0
4	False	1504.0

FIGURE 1.2 – Observed time and status of patients with primary biliary cirrhosis (PBC dataset).

In Figure 1.2, we observe the target information, where 'status' indicates 'true' if the individual has died and 'false' if not. Meanwhile, 'time' represents the duration recorded at the end of the period, which could correspond to the time of death or the time the individual left the hospital.

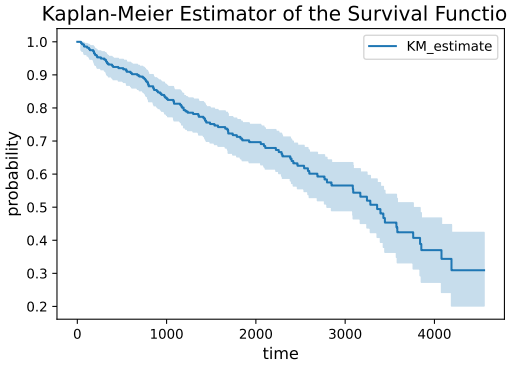


FIGURE 1.3 – Survival probability estimation for the PBC dataset using the Kaplan-Meier method. The curve represents the estimated survival probabilities, with the shaded blue area around the curve indicating the confidence interval, which highlights the precision of the estimates over time.

Figure 1.3 illustrates the survival probability estimated by the Kaplan-Meier method (1.1). The confidence interval shows that the survival probability prediction is more reliable at the beginning of the observed time period. An important advantage of the Kaplan-Meier estimator is that it does not assume any specific distribution for the event times, making it versatile and applicable to a broad range of datasets. This estimator relies on the proportion of events at each time point of interest, rather than on individual characteristics.

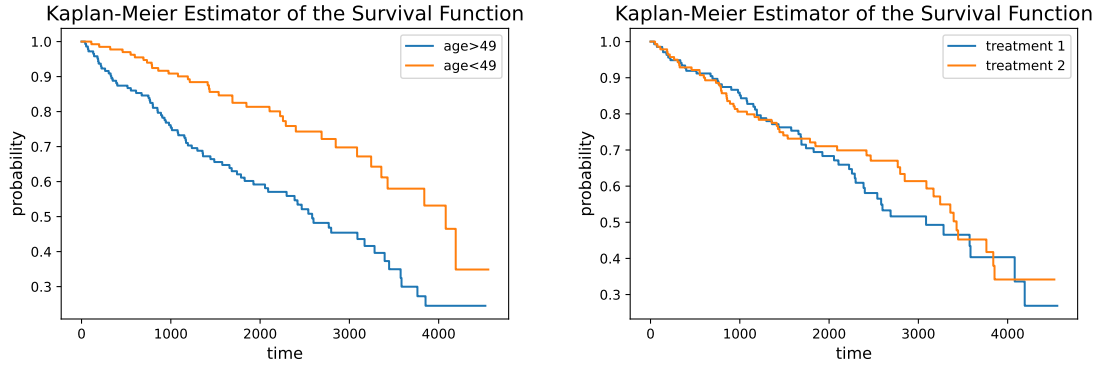


FIGURE 1.4 – Kaplan-Meier estimation of the PBC dataset comparing two age groups [left] and Kaplan-Meier estimation comparing two treatment groups [right].

To evaluate the effects of different features on the survival function, we categorize individuals into distinct groups based on these features and estimate the survival function separately for each group. This approach enables a targeted analysis of how specific characteristics influence survival outcomes, similar to the methodology employed in life tables techniques. In Figure 1.4 [left], we observe the difference in the survival function estimations between two age groups : the blue curve represents individuals over 49 years old, while the orange curve represents those under 49 years old. We note that older individuals consistently have a lower probability of survival compared to younger patients. In Figure 1.4 [right], we present the difference in the survival functions for two treatment groups. We observe that there is no consistent dominance, however, treatment 2 shows a slightly higher probability of survival compared to treatment 1 over a certain period of time. The main disadvantages of the Kaplan-Meier estimator are that it assumes the survival probability is homogeneous within the groups, in addition to the lack of covariate adjustment. This suggests the need for complementary methods or more complex models in certain research contexts.

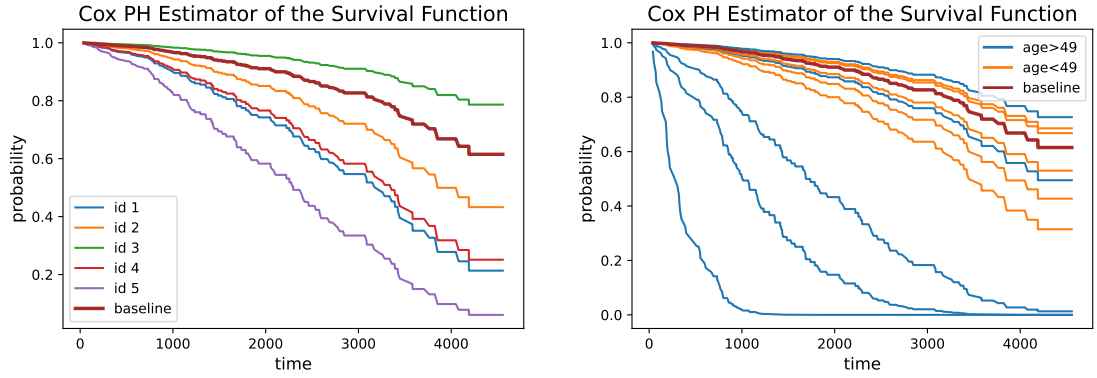


FIGURE 1.5 – Cox PH estimation of the PBC dataset comparing several individuals [left] and Cox PH estimation comparing two age groups [right].

Figure 1.5 [left] shows the survival function predictions of the Cox proportional hazards model for 5 individuals. Notably, the survival function varies among individuals due to covariate

dependency. This variation introduces greater analytical depth and enables personalized predictions. In addition, we remark that the Cox proportional hazards model uses the likelihood principle for the parametric estimation and, in a second step, a non-parametric approach for the baseline function. We show the baseline function in brown, which remains the same across all individuals. On the right, Figure 1.5 contrasts predictions for individuals older than 49 (in blue) with those younger than 49 (in orange). We observe that older individuals typically show a lower survival probability, aligning with the Kaplan-Meier estimates (Figure 1.4). However, some older individuals exhibit a higher survival probability than some younger patients, which highlights the influence of other covariates on survival predictions. This illustrates how the interaction of multiple characteristics significantly affects model outcomes. We show the baseline function in brown, which remains constant across all individuals. This illustrates the multiplicative effect of the exponential term on the survival function.

Throughout the thesis, we use three open datasets : Primary Biliary Cirrhosis (PBC), German Breast Cancer Study Group 2 (GBCSG2) [117], and Kaggle Telecom Churn (TLCM) [71]. Particularly, in Chapter 3, we employ these datasets to compare the performance of various methods, including the Cox proportional hazards model. Each dataset will be described in detail in the corresponding chapters.

1.2.4 Scoring rules

One important question, not only in survival analysis but also in general modeling, concerns how well the model predicts the target. In the case of regression problems, the most commonly used score is the mean squared error (MSE), which has its roots in the work of Gauss and Legendre [124]. A straightforward approach would be to predict the event times $\hat{t}_1, \dots, \hat{t}_N$ of the N individuals and compare this prediction with the real times $t_1, \dots, t_N$ using the estimator of the MSE :

$$\frac{1}{N}\sum_{i=1}^{N}(t_i - \hat{t}_i)^2 \,.$$

Assessing accuracy by using time point predictions in many cases is not satisfactory [103]. A more common approach is to consider the estimator of the survival function $\widehat{S}(t|x)$, which is seen as predictions of the event status $\mathbb{1}\{T > t\}$. The MSE in this case is :

$$\mathbb{E}\left[\left(\mathbb{1}\{T > t\} - \widehat{S}(t|x)\right)^2\right], \quad t \geqslant 0, x \in \mathbb{R}^d \,.$$

However, in the presence of censorship, we do not observe the realization of the event time t_i for every individual. Therefore, it is not straightforward to estimate the MSE directly. The MSE must be estimated by carefully choosing a method for censoring adjustment. Further discussions on the estimation of the MSE using survival data can be found in Graf et al. [52] and Gerds and Schumacher [47].

As the MSE can be hard to estimate, the survival analysis community developed alternative performance measures. Recent work includes that of Cwiling et al. [27], who proposed using the restricted mean survival time (RMST) to evaluate the goodness-of-fit of survival models. They estimate the mean squared error of an RMST estimator using inverse probability censoring weighting. Another example is the work by Qi et al. [110], who proposed a method for estimating the mean absolute error (MAE) based on the predictions of event times and found a way to

evaluate the accuracy of these metrics using semi-synthetic data. Another possible approach involves the use of statistics such as the Pearson statistic or the deviance statistic. For more details, refer to Tutz [129].

We briefly present the performance measures that are considered throughout this thesis : the concordance index [60], the integrated Brier score [47], the likelihood [39], and the receiver operating characteristic (ROC) curve. Selecting an appropriate scoring rule is a non-trivial task, requiring careful consideration of the strengths and limitations of each measure. Specifically, in Chapter 2 we use the likelihood as a loss function to measure the regret of the algorithms. Then, in Chapter 3, our focus will be on comparing the performance of multiple models and examining how model rankings vary between the concordance index and integrated Brier score. Finally, in Chapter 4, we apply this comparison, including the ROC curve, to a real industrial case. A review of statistical methods for evaluating the performance of survival predictions can be found in [102].

Concordance index : This score was proposed by Harrel et al. [60] and it quantifies how well the model predicts the ordering of the event times, in other words, it measures the concordance in between the predicted risk and the actual outcomes. This score is better for higher values. It is a non-parametric measure, implying that it does not assume a specific distribution of the survival times, enhancing its flexibility in various applications. However, it is insensitive to calibration, a model could have a high concordance index but still produce poorly calibrated risk probabilities.

Integrated Brier score : The Brier score was first proposed by Brier [16] with the aim to evaluate the accuracy of weather forecast. Later, Gerds and Schumacher [47] proposed methods to consistently estimate the expected Brier score in survival models, contributing significantly into formalizing this score in the survival community. The Brier Score, measures the mean squared difference between observed outcomes and predicted probabilities at a specific time point. The Brier score is defined as :

$$\mathrm{BS}(t) = \frac{1}{N}\sum_{i=1}^{N} w_i \left(\mathbb{1}\{u_i \geqslant t\} - \widehat{S}(t|x)\right)^2, \quad t \geqslant 0\,,$$

where $u_i = \min\{c_i, t_i\}$, a realization of $\min\{C, T\}$, is the observed time of individual i, and w_i denotes the weight of individual i, which is associated with censorship. For more details see on this formula see Section C1.2. Under certain hypothesis this score is a consistent estimator of the MSE [47]. The Integrated Brier Score is obtained by averaging the Brier Scores over a range of time points, usually from the start of the study to the end or until a time of particular interest $b > 0$:

$$\mathrm{IBS} = \frac{1}{b}\int_0^b \mathrm{BS}(t)dt\,.$$

This provides a summary measure of the model performance across the follow-up period. A model has better accuracy if the integrated Brier score is lower.

Likelihood : The likelihood is a fundamental concept in statistics and statistical modeling, representing the probability of observing the data given a set of parameters for a specified model. Specifically, if you have a statistical model that describes how your data is generated, the likelihood of the model parameters given the observed data is a function that quantifies how likely it is to observe the given data for different parameter values. We suppose that the survival function is a function of a specified parametric family $\Theta \subseteq \mathbb{R}^d$, thus, we write it as $S(t|x_i, \theta)$. Given f the

density of the event time distribution T, the likelihood is :

$$\ell(\theta) = \prod_{i=1}^{N} f(u_i|x_i,\theta)^{\delta_i} S(u_i|x_i,\theta)^{1-\delta_i}, \quad x_i \in \mathbb{R}^d \,,$$

where $\delta_i = \mathbb{1}\{t_i \leqslant c_i\}$ is the event indicator of individual i. While likelihood itself is a measure of fit rather than predictive accuracy, derived measures such as the log-likelihood can be used in cross-validation settings to assess out-of-sample predictive accuracy. Models that achieve higher log-likelihood values on validation data are considered to have better predictive accuracy.

ROC curve : The ROC curve measures the performance of a binary classifier model at varying threshold values. It is particularly suitable in scenarios where the classification threshold is not clear. In order to measure accuracy with the ROC curve in survival analysis, we consider the prediction task to be the status indicator δ. We set different probability thresholds to decide the status prediction, and for each threshold, we compute the number of false positives (FP), true positives (TP), false negatives (FN), and true negatives (TN). Then, we define the false positive rate and the true positive rate :

$$FPR = \frac{FP}{FP + TN}, \quad TPR = \frac{TP}{TP + FN} \,.$$

The ROC curve shows the variation of the true positive rate against the false positive rate as the classification threshold is varied. The faster the TPR grows with respect to the FPR, the better. This curve can be quantitatively summarized by the Area Under the Curve (AUC). A higher AUC value indicates better predictive accuracy. This is a very well-known score within the industrial environment, and it has been adapted in multiple ways to handle the time-dependent survival outcomes [67, 75].

Illustrating censorship bias

We follow the data simulation procedure outlined in Section 1.3.1, with an arrival time of $\tau = 0$, and we fit a Cox proportional hazards model to compare each score with its non-censored counterpart. We simulate data 50 times, with approximately 50% censorship, and split the data, allocating 75% for training and 25% for validation. We then compare the score values in two scenarios within the validation set : one considering the presence of censorship and the other with the complete information that is known from the data simulation.

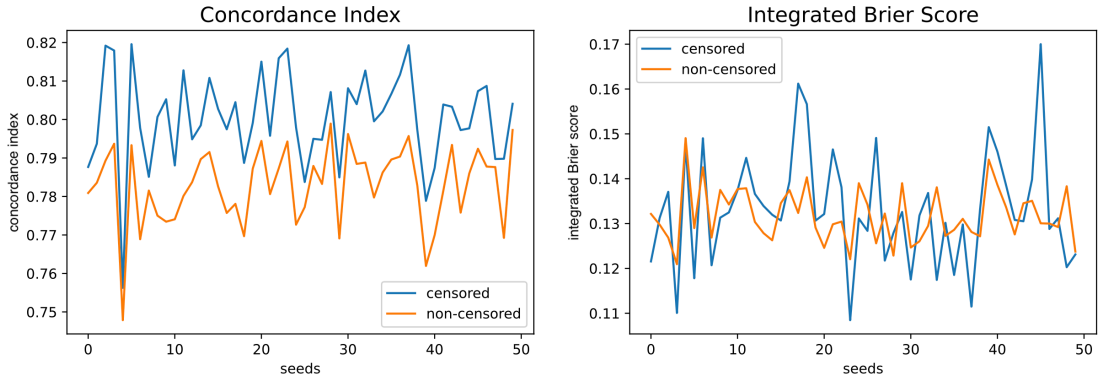


FIGURE 1.6 – Concordance index comparison across multiple dataset splits [left] and integrated Brier score comparison across multiple dataset splits [right]. We use simulated data to asses the impact of considering censorship in the validation set.

In Figure 1.6 [left], we observe the bias of the concordance index when considering censorship. In the presence of censorship, the score is artificially higher, consistently overestimating the accuracy of the model across the 50 data simulations. In Figure 1.6 [right], the difference is less pronounced, but we observe that the censored case exhibits greater variance.

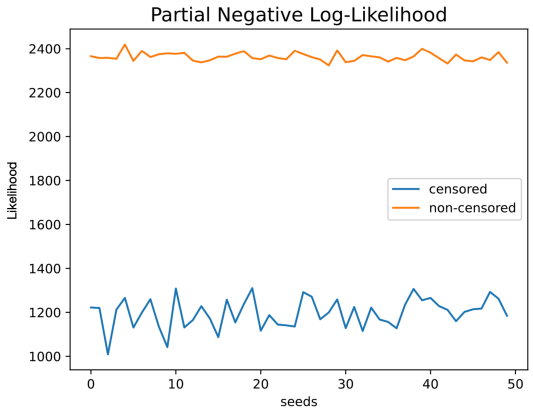


FIGURE 1.7 – Partial negative log-likelihood comparison across multiple dataset splits. We use simulated data to asses the impact of considering censorship in the validation set.

In Figure 1.7, we observe the partial negative log-likelihood. In this context, since we are considering the negative log-likelihood, lower values indicate higher accuracy. The likelihood approach is limited by its requirement for a parametric model. Because the Cox proportional hazards model is semi-parametric, we can only compute the partial log-likelihood, which corresponds to the parametric component of the Cox model. Similarly to what we observe with the concordance index, the censored case exhibits greater accuracy. However, when computing the likelihood with complete information, the accuracy is lower. This highlights the potential bias introduced by censorship and the importance of selecting an appropriate accuracy measure to

assess the performance of the methods. Let us note that we do not include the ROC curve in this analysis because, in the non-censored case due to our approach, all status values are 1, leading to no false positives or true negatives, which makes computing the FPR impossible.

1.3 Online Convex Optimization for Survival Analysis

One important contribution of this thesis is the application of online convex optimization techniques to estimate survival distributions, a field that has previously been unexplored. To achieve this, we assume an exponential shape for the hazard function and that there exists $\theta \in \Theta$ such that :

$$h(t|x, \tau, \theta) = \exp(\theta^\top x)\mathbb{1}\{t \geqslant \tau\}, \quad \theta \in \Theta, t \geqslant 0, x \in \mathbb{R}^d, \tag{1.3}$$

where τ, the arrival time, is no longer assumed to be zero. The parameter family $\Theta \subseteq \mathbb{R}^d$ is defined as convex and bounded, and our objective is to estimate the parameter θ. For this purpose, we consider a sequential setting in which we have an horizon time $n \geqslant 1$ and a time partition $(t-1, t]$ for $t = 1, \ldots, n$, that is independent of the observations. We consider a set of N individuals, each associated with an observed time $u_i \sim \min\{T, C\}$, where T and C are assumed to be conditionally independent given τ_i and the covariate vector x_i. Our focus is on methods based on the maximum likelihood principle. We remark that our model (1.3) corresponds to the parametric term of the Cox proportional hazards model (1.2).

An online convex optimization problem involves, considering at each iteration t, a loss function $\ell_t : \Theta \to \mathbb{R}$ to be minimized. The online convex optimization algorithm will choose for each t, a parameter θ_t, suffer a loss of $\ell_t(\theta_t)$ and observe the gradient $\nabla\ell_t(\theta_t)$. A general step for updating the parameter $\theta_t \in \Theta$ in an online convex optimization algorithm can be described as :

$$\theta_{t+1} = \theta_t - \frac{1}{\gamma} P_t \nabla\ell_t(\theta_t), \quad t \geqslant 1\,,$$

where γ is the learning rate and P_t is a preconditioning matrix, which it is used to scale the gradient and can help in accelerating the convergence. In some cases, P_t might be the identity matrix, simplifying the update rule. In addition, we can consider second order algorithms by specifying P_t depending on the second derivative $\nabla^2\ell_t(\theta_t)$ or an approximation of it. For a given time horizon n, the objective of the algorithm is to minimize the regret function :

$$\text{Regret}_n = \sum_{t=1}^{n} \ell_t(\theta_t) - \min_{\theta\in\Theta} \sum_{t=1}^{n} \ell_t(\theta), \quad n \geqslant 1\,,$$

where ℓ_t is a convex function. The interest of online convex optimization is to develop algorithms that efficiently minimize cumulative loss over time, achieving near-optimal performance with as few iterations as possible. This setting is applicable in scenarios where decisions need to be made sequentially and adaptively, such as in stock market trading, online advertising, machine learning model updates based on streaming data, and attrition prediction.

There are several methods developed for optimizing convex functions in an online setting, such as Online Gradient Descent (OGD) [145], Online Newton Step (ONS) [65], Adam [79], and AdaGrad [32], among others (see Hazan [64]). The regret bound of these algorithms is strongly related to the curvature of the loss function. In particular, we consider the following property :

Definition 2. (Exp-concavity) A convex function $\ell : \Theta \to \mathbb{R}$ is μ-exp-concave if the function $p(\theta) := \exp(-\mu\ell(\theta))$ is concave.

The exp-concavity of loss functions allows for more aggressive learning rates that adjust more rapidly to the observed data, while still maintaining control over the growth of cumulative regret. This is because the additional curvature provided by exp-concavity, with respect to convexity, helps in more accurately predicting the outcome of future decisions based on past data, thus reducing the possibility of making large errors. In addition, it is often possible to design algorithms that achieve logarithmic regret bounds ($\mathcal{O}(\log(n))$). This is significantly better than what can be guaranteed for general convex functions, where regret bounds might be linear or sublinear (for example $\mathcal{O}(\sqrt{n})$).

Online Newton Step, an online version of Newton-Raphson method [140], which uses second order information to update the predicted parameters, assures a logarithmic regret bound for exp-concave functions. More explicitly, for μ-exp-concave losses $(\ell_t)_{t\geqslant 1}$ with bounded domain (Θ of diameter D) and gradients (bounded by G). We consider a learning rate $\gamma = 1/2\min\{1/GD, \mu\}$, then the regret bound of ONS will be

$$\text{Regret}_n \leqslant \frac{d\log(2nG^2\gamma^2D^2)}{\gamma} \quad n \geqslant 1\,.$$

Given the fast convergence rate of ONS, we chose this algorithm to optimize the negative log-likelihood ℓ_t of the exponential model (1.3), using its nice convex characteristics to estimate the parameters of the hazard function. We observe that the choice of the learning rate proposed by Hazan [65] is strongly related to the exp-concavity constant ; therefore, the regret bound of ONS is highly influenced by this constant. We note that if the exp-concavity constant is small, the learning rate will also be small, and the regret bound may become arbitrarily large. No guarantee exists regarding the lower bound of the exp-concavity constant.

We discuss this issue and its possible solutions in detail in Chapter 2. We propose an aggregation adaptive algorithm that addresses the problem of regret bound dependency on the exp-concave property, enhancing robustness while maintaining fast regret bounds. Additionally, we introduce a solution to bound the regret in the stochastic setting. We present an example using simulated data in the following section.

1.3.1 Illustrating the effects of learning rate choices

In this section, we simulate data to illustrate the issue of the learning rate and how it influences the cumulative loss of the ONS algorithm. We considered $N = 2000$ samples and $n = 500$ iterations. Then, considering $d = 3$, for each individual i we take a realization x_i of a multivariate random normal of dimension $(N, d-1)$ with parameters

$$\eta = \begin{pmatrix} 1 \\ -2 \end{pmatrix}, \qquad \Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix},$$

and we add an intersect column that transforms x_i into a matrix of dimension (N, d). The real value for θ will be set as $\theta^* = (-0.5, -0.8, 0.3)$. We sample the arrival times τ as a uniform between 0 and n (denoted by $\mathcal{U}(0, n)$), we set T to follow an exponential distribution of rate $\exp((\theta^*)^\top x)$, and C to follow a uniform distribution between 0 and 0.35 :

$$T \sim \tau + \exp(\exp((\theta^*)^\top x)), \qquad C \sim \tau + \mathcal{U}(0, 0.35).$$

The parameters of the censored distribution were adjusted in order to have around 50% of censorship and the other parameters of the data simulation were chosen arbitrarily. We selected two learning rates for our experiment : a small rate, $\gamma_1 \approx 0.03$, and a larger rate, $\gamma_2 \approx 10$, to implement the ONS algorithm. We repeat this experiment 100 times and we present the average results within the 100 simulations.

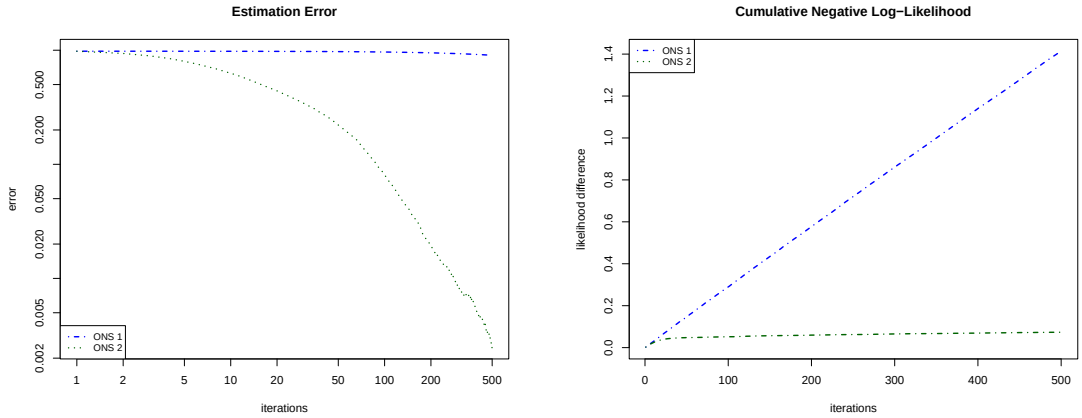


FIGURE 1.8 – Quadratic error of the parameter estimations of ONS algorithm [left] and cumulative negative log-likelihood difference [right]. We show the variation of ONS results when choosing different learning rates.

In Figure 1.8 [left], we compare the quadratic errors in the estimations made by the ONS algorithm with learning rates γ_1 (ONS 1) and γ_2 (ONS 2). We note that ONS 2 exhibits a lower estimation error compared to ONS 1, suggesting that a larger learning rate (γ_2) results in more accurate estimations. Figure 1.8 [right] highlights this findings showing the difference between the cumulative negative log-likelihood of the estimations and the real parameters. In addition, we observe that the cumulative error of ONS 2 seems to grow at a logarithmic scale. This Figure illustrates how sensitive is the algorithm to the learning rate selection.

1.3.2 Stochastic setting

In Chapter 2 we use the stochastic approach proposed by Wintenberger [139], which involves assessing stochastic risk to determine the regret of the algorithms. We consider a filtration $\mathcal{F}_t, t \geqslant 1$ and an $\mathcal{F}_t$ adapted sequence of random loss functions ℓ_t for $t \geqslant 1$. At each iteration t an algorithm predicts θ_t and incurs the random conditional risk $L_t(\theta_t) := \mathbb{E}[\ell(\theta_t)|\mathcal{F}_{t-1}] = \mathbb{E}_{t-1}[\ell(\theta_t)]$. For a time horizon n, the stochastic regret we aim to minimize is :

$$\text{Risk}_n := \sum_{t=1}^{n} L_t(\theta_t) - \min_{\theta \in \Theta} \sum_{t=1}^{n} L_t(\theta), \quad t \geqslant 1 \,.$$

In this context, we can study the stochastic regret bounds of online convex optimization algorithms such as ONS. Similarly to the deterministic case, their convergence is strongly related to the curvature of the losses. We define the stochastic counterpart of the exp-concavity property.

Definition 4. (Stochastic exp-concavity) A sequence of random loss functions (ℓ_t) for $t \geqslant 1$ are μ-stochastically exp-concave if

$$L_t(\theta_1) \leqslant L_t(\theta_2) + \nabla L_t(\theta_1)^\top(\theta_1 - \theta_2) - \frac{\mu}{2}\mathbb{E}_{t-1}[(\nabla \ell_t(\theta_1)^\top(\theta_1 - \theta_2))^2], \quad \forall \theta_1, \theta_2 \in \Theta, t \geqslant 1, a.s.$$

It was proved that ONS achieves logarithmic stochastic regret for a sequence of μ-stochastically exp-concave random losses with a learning rate of $\mu/3$ [139]. The stochastically exp-concave property extends the range of applicable loss functions, particularly accommodating those from the survival analysis setting, which exhibits poor exp-concavity. Moreover, this setting enhances convexity properties, notably, the conditional risk demonstrates more favorable convex characteristics compared to the initial loss functions, which allows us to demonstrate that the stochastic exp-concavity constant is lower bounded, ensuring that the algorithm attains better theoretical guarantees for the stochastic regret.

1.4 Application to Attrition Prediction

One important application of survival analysis in the industrial sector is predicting employee attrition, which significantly affects the operational efficiency and strategic planning of organizations. By forecasting attrition, companies can take preventive actions to retain employees, thereby reducing turnover expenses and retaining high-performing employees. Furthermore, it allows companies to proactively manage their human resources, leading to enhanced financial performance, improved employee morale, and strengthened organizational cohesion.

The problem of attrition prediction has evolved significantly over time. Initially, studies on employee turnover were rooted in industrial psychology and management theories, focusing on worker satisfaction and its impact on productivity, and therefore, indirectly addressing attrition (see Herzberg [68]). As the field of organizational behavior matured, researchers began to examine turnover more directly, linking it to organizational commitment and job stress (see Lang et al. [84]). These studies underscored the multifaceted nature of attrition, incorporating psychological, sociological, and environmental factors.

The advent of data science and machine learning has transformed attrition prediction in recent decades. Advanced statistical models and algorithms now enable the analysis of large datasets to identify patterns and predictors of turnover. Notable contributions include classical statistical methods by Bennett et al. [9], the adaptation of machine learning techniques by Ajit P. [4] and Frye B. et al. [45], along with survival analysis approaches by Morita, J. et al. [100] and Jin, Z. et al. [73].

Today, attrition prediction is an interdisciplinary field, incorporating insights from human resources, organizational psychology, data science, and economics. This evolution reflects an ongoing quest to understand and mitigate turnover, leveraging both theoretical insights and technological advancements to foster organizational resilience and employee satisfaction. We discuss different approaches to the attrition prediction problem in Chapter 4 and we apply the survival analysis techniques discussed throughout the thesis to a real industrial case.

1.5 Contributions of the Thesis

The main mathematical contribution of the thesis is presented in Chapter 2 which is oriented towards solving the issue exposed in Section 1.3. In the first place we detail the mathematical framework that allows the use of online convex optimization algorithms to predict survival curves. We consider a horizon time n and a time partition $(t-1, t]$ for $t = 1, \dots, n$, and under the assumption of the exponential model detailed in equation (1.3) we give an explicit expression of its likelihood in the sequential setting. This function is used as the loss to assess regret, and we provide its formulation in this section so it can be cited in the theorem statement :

$$\ell_t(\theta) = \sum_{i=1}^{N} -y_{it}\theta^\top x_i(u_i) + r_{it}\int_{\tau_i \vee (t-1)}^{u_i \wedge t} \exp(\theta^\top x_i) ds\,, \qquad \theta \in \Theta,\ t = 1, \dots n, \tag{1.4}$$

where we remind $u_i \wedge t = \min\{u_i, t\}$ and $\tau_i \vee (t-1) = \max\{\tau_i, t-1\}$, and we define $y_{it} := \delta_i \mathbb{1}\{t-1 < u_i \leqslant t\}$ and $r_{it} := \mathbb{1}\{\tau_i \leqslant t, u_i > t-1\}$. In this expression we explicitly observe the contribution of censored and non-censored individuals to the likelihood. We do the following assumption :

Assumption 1. There exists $D, G > 0$ such that for all $t \geqslant 1$ and $\theta \in \Theta$, $\|\theta\| \leqslant D$ and $\|\nabla \ell_t(\theta)\| \leqslant G$.

This assumption is crucial in the context of online convex optimization to assure algorithm convergence, stability, and performance, and it is widely used in this field [64]. We use it to bound the regret of our algorithm. The following property will also have a significant role in assuring theoretical guarantees for the OCO algorithms.

Definition 5. (Directional derivative condition – DDC) We say that a function $\ell : \Theta \to \mathbb{R}$ satisfy the directional derivative condition for a constant $\gamma > 0$ if for any pair $\theta_1, \theta_2 \in \Theta$

$$\ell(\theta_2) \geqslant \ell(\theta_1) + \nabla \ell(\theta_1)(\theta_2 - \theta_1) + \frac{\gamma}{2}\left(\nabla \ell(\theta_1)(\theta_2 - \theta_1)\right)^2. \tag{1.5}$$

This condition is related to the curvature of the loss function and is weaker than the exp-concavity property (see Definition 2), which is usually considered an alternative to strong convexity. We previously discussed the issue of the learning rate for the ONS algorithm : when the exp-concavity property is weak, the learning rate proposed by Hazan [65] becomes too small, leading to large regret bounds.

1.5.1 Stochastic approach

The first solution we propose is to study the survival problem through a stochastic approach. We model the arrival time τ as a homogeneous Poisson process with intensity λ, and for each $t \geqslant 1$ we define the count variable $N_t = \sum_{i=1}^{\infty} \mathbb{1}\{\tau_i \leqslant t\}$, which allows us to define the stochastic loss :

$$\ell_t(\theta) := \sum_{i=1}^{N_t} -y_{it}\theta^\top x_i + r_{it}\exp(\theta^\top x_i)\big((u_i \wedge t) - (\tau_i \vee (t-1))\big), \quad \theta \in \Theta, t \geqslant 1\,. \tag{1.6}$$

The stochastic nature of the losses, particularly the stochastic risk L_t, enhances the convexity properties compared to deterministic losses. This allows us to establish a lower bound on the

stochastic exp-concavity constant, preventing the regret from exploding. In addition, we make a design assumption necessary to prove the strong convexity of L_t, which we include in this section for completeness.

Assumption 2. There exist $A > 0$ such that $\mathbb{E}[xx^\top \mathbb{1}\{T \leqslant C\}(1-T)_+ | \tau = 0] \succcurlyeq AI_d$.

And we prove the following theorem that states logarithmic stochastic regret for ONS.

Theorem 2. Given $\varrho > 0, n \geq 1$ and the stochastic losses $(\ell_t)_{t=1,2,\dots}$ from Equation (1.6), then under Assumption 2, a bounded domain of diameter D and hyperparameter γ, the stochastic exp-concavity constant, the ONS algorithm has logarithmic stochastic regret with probability $1 - 4\varrho$. Specifically, $\text{Risk}_n = \mathcal{O}(\log(n/\varrho)/\gamma)$, for $n \geq 1$.

To prove this theorem, we need to establish a lower bound for the stochastic exp-concavity constant. This constant is crucial as it not only helps demonstrate that the regret bound is logarithmic, but also ensures that it does not explode. This is in contrast to Hazan's regret bound, where there is no guarantee regarding the smallness of the exp-concavity constant. Additionally, we prove the following theorem that assures the convergence of the algorithm prediction to the real parameter.

Corollary 1. Given $\varrho > 0, n \geq 1$ and the stochastic losses $(\ell_t)_{t=1,2,\dots}$ from Equation (1.6), we consider θ_t the ONS prediction at time t and $\bar{\theta}_n$ the average prediction $\bar{\theta}_n = \frac{1}{n}\sum_{t=1}^n \theta_t$. Defining the optimal parameter

$$\theta^* = \arg\min_{\theta \in \Theta} \sum_{t=1}^n L_t(\theta),$$

then, under Assumption 2, a bounded domain of diameter D and hyperparameter γ, with probability $1 - 4\varrho$ we have :

$$||\bar{\theta}_n - \theta^*||^2 \leqslant \mathcal{O}\left(\frac{\log(n/\varrho)}{\gamma n}\right), n \geq 1.$$

Let us note that if we assume the exponential model from Equation (1.3), θ^* turns out to be the parameter of the model. In this corollary, we demonstrate convergence to this 'real' parameter, which in turn minimizes the expected value of the negative log-likelihood in a stationary framework. Both of the stochastic results are possible thanks to the strong convexity of the risk functions L_t, which is not the case in the deterministic setting.

1.5.2 Deterministic setting

The second solution we propose is a new aggregation algorithm, SurvONS described in Algorithm 1, which adaptively selects the learning rate while maintaining control over the regret bound. We prove that this algorithm has bounded regret in the following theorem.

Theorem 3. Let $n \geq 1$, $(\ell_t)_{t=1,\dots,n}$ be the sequence of losses defined in (1.4), that are assumed to satisfy Assumption 1 and (1.5) with constants $\gamma_t \in (0, 1/GD)$. Let $K \geqslant 1$ and $\Gamma \in (0, 1/(4GD))^K$. Then, Algorithm 1 with hyperparameters Γ and $\mathcal{E} = 1/(\Gamma D)^2$, satisfies the regret upper-bound :

$$\text{Regret}_n \leqslant \min_{\gamma \in \Gamma} \left\{ \frac{2\log(K) + 5d\log(n)}{\gamma} + \gamma G^2 D^2 n_\gamma \right\},$$

where $n_\gamma := \sum_{t=1}^n \mathbb{1}\{\gamma_t < \gamma\}$, $\gamma > 0$.

The regret bound proposed in this theorem does not depend on the exp-concavity property, contrary to Hazan's bound. The regret remains bounded and achieves a trade-off between the suboptimal choices of the learning rate (small constants) and the frequency at which the algorithm selects the user-specified constant γ, rather than the optimal γ_t determined at iteration t. This helps to compensate for the increase of regret.

We observe that aggregation methods improve robustness in hyperparameter selection, however, maintaining a fast rate of convergence under this scheme is challenging. Additionally, we find that the parametric approach in this context is difficult to fit due to the lack of a strong convexity property in the survival loss. Finally, let us note that there are no guarantees on the minimum number of iterations that OCO algorithms require to find an optimal solution, and this number could potentially be large.

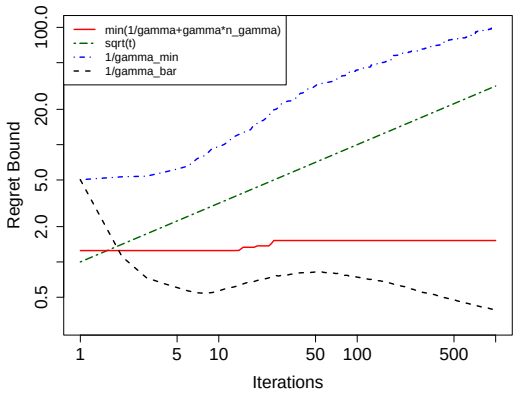


FIGURE 1.9 – Comparison of regret bound orders (up to logarithmic factors) for multiple online methods using simulated data.

In summary, Figure 1.9 shows the theoretical regret bounds (up to logarithmic factors) for Online Gradient Descent (OGD) in green, ONS in blue, SurvONS in red, and the stochastic approach in black [1]. We observe the benefits of considering an average estimation of γ, which represents the stochastic setting, and the advantages of SurvONS, both of which have lower regret bounds. The simulation framework for this experiments is presented in Chapter 2.

1.5.3 Ensemble methods and applications

The second contribution of this thesis is the experimental comparison of various parametric and machine learning models using two scores, with the objective of understanding which factors most significantly affect the performance of the models and the behavior of the scores. . This comparison was conducted using three datasets : PBC [125], which was described earlier, GBCSG2 [117], and TLCM [71]. We carried out simulation experiments and we observed that the ranking of the methods depends primarily on the dataset and the alignment between its distribution properties and the assumptions of the model. The results of this experiment are detailed in Chapter 3.

1. See the codes at the GitHub repository : `https://github.com/camferna/Online-Learning-Approach-for-Survival-Analysis`

Additionally, we propose an aggregation algorithm, described in Algorithm 2, to enhance robustness and performance. Specifically, for each method $j = 1, \dots, K = 5$, we consider its estimation of the survival probability $\hat{S}_j : \mathbb{R}^+ \to [0, 1]$, and the ensemble estimation will be the convex combination :

$$\hat{S}(t|x_i) = \sum_{j=1}^{K} \lambda_j \hat{S}_j(t|x_i) \quad \text{such that} \quad \sum_{j=1}^{K} \lambda_j = 1 \,.$$

where the weights λ_j are obtained by minimizing the integrated Brier score with an exponential gradient descent procedure. This ensemble algorithm allows us to guarantee competitive performance regardless of the dataset.

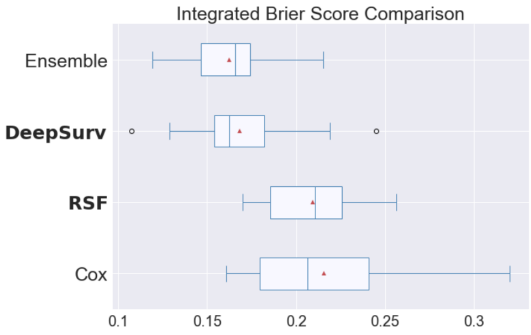


FIGURE 1.10 – Box plot comparison of the ensemble method using the integrated Brier score across multiple dataset splits on the primary biliary cirrhosis dataset.

Figure 1.10 shows the performance of the ensemble method measured by the integrated Brier score on the PBC dataset [2]. We observe that it outperforms the other methods. Additionally, our experiments generally indicate that machine learning methods, such as DeepSurv and RSF, perform better and adapt more effectively to varying data distributions.

1.5.4 Employee attrition prediction

Finally, in Chapter 4, we apply the methods proposed throughout the thesis to a real industrial case : predicting employee attrition. We use a real dataset consisting of around 10000 employees of whom 3% had left the company by the end of the study. Notably, this results in a high percentage of censoring, which adds complexity to the analysis. We compare multiple survival analysis methods, including SurvONS and the ensemble method from Chapter 3, and engage in a discussion on the influence of features on model performance.

2. The code can be found in the repository : `https://github.com/camferna/Ensemble-Methods-and-Time-to-Event-Analysis-Models`

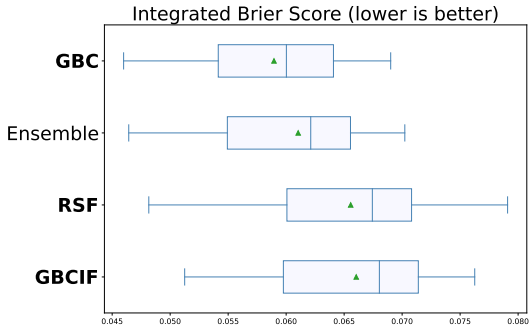


FIGURE 1.11 – Box plot comparison of the ensemble method using the integrated Brier score across multiple dataset splits on the attrition dataset.

Figure 1.11 shows the performance of the ensemble method measured by the integrated Brier score on the attrition dataset [3]. We observe that it outperforms the other methods except for gradient boosting Cox (GBC).

Future work could explore further the parameter family bounds and their impact on algorithm performance, the integration of state-space models, and the application of continuous ranked probability scores in survival data analysis. Additionally, expanding the aggregation methods to incorporate time-varying weights and diverse optimization techniques could refine our approach, especially in adapting machine learning methods for censored tabular data. These areas promise substantial improvements in the robustness and efficacy of survival analysis techniques. While significant strides have been made in adapting online convex optimization to censored data and in the other aspects explored in this thesis, there is still a lot of room for improvement.

3. The attrition dataset is confidential due to data privacy agreements. Consequently, the code cannot be publicly shared.

Algorithm 1 SurvONS

Input : $(\ell_t)_{t=1,2,\dots}, D > 0, n \geqslant 1$, grids $\Gamma, \mathcal{E}$

Initialization : for each γ_k in Γ : $\theta_0(\gamma_k) \in \Theta$, $\pi_{0,k} = \frac{1}{K}$, $\hat{\theta}_0 \in \Theta$, $A_0^{-1} = \mathcal{E}^{-1}\mathbb{1}_d$

for iteration $t = 1, \dots, n$ **do**

Update : $\hat{\theta}_t = \sum_{k=1}^{K} \pi_{t,k}\, \theta_t(\gamma_k)$

Observe :

$$\nabla \ell_t(\hat{\theta}_t) = \sum_{i=1}^{N} -y_{it} x_i(u_i) + r_{it} \int_{\tau_i \vee t-1}^{u_i \wedge t} \exp(\hat{\theta}_t^T x_i(s)) x_i(s) ds$$

$$\nabla^2 \ell_t(\hat{\theta}_t) = \sum_{i=1}^{N} r_{it} \int_{\tau_i \vee t-1}^{u_i \wedge t} \exp(\hat{\theta}_t^T x_i(s)) x_i(s) x_i(s)^T ds$$

$$\mu_t = \frac{\nabla \ell_t(\hat{\theta}_t)^T \nabla^2 \ell_t(\hat{\theta}_t) \nabla \ell_t(\hat{\theta}_t)}{||\nabla \ell_t(\hat{\theta}_t)||^4}$$

$$\gamma_t = 2 \frac{-\frac{1}{\mu_t} \log(1 + \mu_t ||\nabla \ell_t(\hat{\theta}_t)|| D) + ||\nabla \ell_t(\hat{\theta}_t)|| D}{(||\nabla \ell_t(\hat{\theta}_t)|| D)^2}$$

for $\gamma_k \in \Gamma$ **do**

Observe :

$$\tilde{\gamma}_t = \max\{\gamma_t/4, \gamma_k\}$$

$$\nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) = \nabla \ell_t(\hat{\theta}_t)(1 + \tilde{\gamma}_t \nabla \ell_t(\hat{\theta}_t)(\theta_t(\gamma_k) - \hat{\theta}_t))$$

Recursion :

$$A_t^{-1} = A_{t-1}^{-1} - \frac{A_{t-1}^{-1} \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k))^T A_{t-1}^{-1}}{1 + \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) A_{t-1}^{-1} \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k))^T}$$

$$\theta_{t+1}(\gamma_k) = \mathrm{Proj}_t \left(\theta_t(\gamma_k) - \frac{1}{\gamma_k} A_t^{-1} \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) \right)$$

end for

Update : $\pi_{t+1,\cdot} = \pi_t \exp\left(-\Gamma \nabla \ell_t(\hat{\theta}_t)^T (\hat{\theta}_t - \theta_t(\Gamma)) - \Gamma^2 (\nabla \ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\Gamma)))^2 \right)$

end for

return $\hat{\theta}_n$

Chapitre 2

Online Learning Approach for Survival Analysis

Abstract

We introduce an online mathematical framework for survival analysis, allowing real time adaptation to dynamic environments and censored data. This framework enables the estimation of event time distributions through an optimal second order online convex optimization algorithm—Online Newton Step (ONS). This approach, previously unexplored, presents substantial advantages, including explicit algorithms with non-asymptotic convergence guarantees. Moreover, we analyze the selection of ONS hyperparameters, which depends on the exp-concavity property and has a significant influence on the regret bound. We propose a stochastic approach that guarantees logarithmic stochastic regret for ONS. Additionally, we introduce an adaptive aggregation method that ensures robustness in hyperparameter selection while maintaining fast regret bounds. The findings of this paper can extend beyond the survival analysis field, and are relevant for any case characterized by poor exp-concavity and unstable ONS. Finally, these assertions are illustrated by simulation experiments.

2.1 Introduction

On the one hand the primary objective of survival analysis is to estimate the time until a critical event occurs, often referred to as survival time or failure time. Examples of such events include customer churn [91], machine failures [19], and employees' attrition [99]. Survival analysis is particularly suited for scenarios where the occurrence of the event may not be observed for all individuals in the dataset. This phenomenon arises when data collection happened before the event occurred, or individuals left the study before experiencing the event, and is called right censoring. As neglecting the censored data is restrictive, it is essential to consider censorship in estimating event time distributions to avoid bias and underestimation. For each individual i with

event time t_i, we define the survival probability function as

$$S_i(t) = \mathbb{P}(t_i \geq t), \qquad t \geq 0.$$

On the other hand convex optimization aims to find the minimum of a convex function over a convex set. It can be extended to an online approach in which the dataset becomes available in sequential order and is used to update the estimations of the algorithms at each step. This setting is suitable when the dataset is rapidly evolving over time, allowing for efficient processing of large volumes of data. Online convex optimization is a broad field with diverse applications such as online portfolio selection in finance, signal processing, communication, and machine learning algorithms ; see Hazan [63] and references therein.

In this paper, we propose the application of online convex optimization algorithms to survival analysis. The combination of these two approaches has not been explored before. Our method offers significant advantages, including explicit algorithms with non-asymptotic convergence guarantees, making it a promising tool for the survival analysis field.

Specifically, we estimate a parametric survival probability function S_i using online convex optimization algorithms : let Θ be a non-empty, convex, compact set in $\mathbb{R}^d$, and ℓ_t the negative log-likelihood of the individuals at risk during the interval $(t-1, t]$, $t \geq 1$. The performance of online convex optimization algorithms is measured with the regret

$$Regret_n := \sum_{t=1}^{n} \ell_t(\theta_t) - \min_{\theta \in \Theta} \sum_{t=1}^{n} \ell_t(\theta), \qquad n \geq 1,$$

which indicates how close the cumulative loss is to the optimal solution. A smaller regret implies better performance, and our objective is to bound its growth with respect to n as slowly as possible.

One of the most widely used algorithms in online convex optimization is the Online Newton Step (ONS) of Hazan et al. [65], renowned for its fast regret convergence rate for exp-concave loss functions. This second-order algorithm relies on a hyperparameter known as the learning rate, whose optimal selection is directly dependent on the exp-concavity properties of the loss functions. The exp-concavity constant plays a fundamental role in the theoretical regret analysis of ONS.

We give a detailed mathematical framework for online survival analysis data and we implement the ONS method to optimize the negative log-likelihood of the exponential model. We note that the ONS algorithm requires a careful selection of the learning rate to ensure robust performance. However, certain choices, such as the learning rate proposed by Hazan et al. [65], might lead to an explosive increase in regret, particularly when applied to the survival losses ℓ_t. Therefore, proper selection of the learning rate is essential in our application.

We discuss various strategies for selecting the learning rate hyperparameter. The first contribution involves applying the stochastic setting from Wintenberger [139] to the survival case. This setting enhances convex properties by assessing stochastic risks rather than cumulative losses, allowing us to attain theoretical guarantees for the stochastic regret that is strongly related to the exp-concavity properties on average. Consequently, this provides the convergence of the algorithm estimations to the real parameter under well-specification. Secondly, in the deterministic setting, we propose to apply ONS to an auxiliary function that recursively adapts the learning rate in response to updates in the exp-concavity constant. We introduce the algorithm SurvONS, an aggregation procedure which ensures a logarithmic regret bound and robustness

in hyperparameter selection over a fixed grid. This provides a new compromise in the context of second-order algorithms : the algorithm either performs well on average (as in the case of BOA [138]) or performs well for certain iterations (as in our case with SurvONS). It is important to emphasize that this algorithm is applicable not only to the survival case but also to any case where the exp-concavity properties are poor and the original versio of ONS is unstable. Finally, we conduct experiments using simulated data to examine our algorithm's behavior under different constraints. We discuss the choice of the grid, and we observe that the combination of multiple ONS allows us to use larger grids in SurvONS than in BOA-ONS [138].

The literature in survival analysis is considerable. The approaches range from non-parametric methods, such as the one proposed by Kaplan and Meier in 1958 [76], to semi-parametric methods like Cox proportional hazards [24], and more recent machine learning applications. For instance, Ishwaran proposed an adapted random forest for censored data inx [72]. Another example is DeepSurv, which was introduced by Katzman in [77]. DeepSurv utilizes deep learning techniques to estimate the log-risk function in the Cox model. From a theoretical perspective, Arjas and Haara [8] proposed a dynamic setting called discrete-time logistic regression. In this model, events are always treated in the order in which they occurred in real time. The authors provided an asymptotic normality result for the maximum likelihood estimator of the regression coefficients. The discrete model is a suitable choice when events are observed at discrete time points ; see Tutz [129]. Building upon Arjas and Haara's framework, Fahrmeir [34] introduced a state-space approach for analyzing discrete-time survival data. This approach includes the estimation of time-varying covariate effects achieved by maximizing posterior densities through the use of Kalman Filter algorithms. Christoffersen [23] provided a method for discretising continuous event times when the instantaneous hazard follows an exponential shape. In a similar setting we provide adaptive estimators with non-asymptotic guarantees for the first time.

2.2 Background on parametric inference

2.2.1 Notation

We consider a set of N individuals denoted by $i \in \{1, \dots, N\}$, each associated with an arrival time $\tau_i \geqslant 0$. Such time could represent when a patient enters the hospital, a client joins the company, or simply when an individual enrolls in the study. Every individual has a unique event time t_i, which is a positive random variable. By definition, we have $t_i \geq \tau_i$ almost surely (a.s). We also define c_i, which marks the cessation of observation for the individual i ; this time is referred to as the censored time. For instance, this might be applicable in cases where the observation period has a predetermined ending. In a more general context, c_i can be a positive random variable satisfying $c_i \geq \tau_i$ a.s. Given that some individuals are censored before the event occurs, and vice versa, it is natural to define the observed time as $u_i := \min\{t_i, c_i\}$. We also define the event indicator $\delta_i := \mathbb{1}\{t_i \leqslant c_i\}$, which provides a way to discern whether an event has happened or if it is censored. For each individual $i \in \{1, \dots, N\}$, we obseve the random variables $(u_i, \delta_i) \in \mathbb{R}_+ \times \{0, 1\}$. Furthermore, we suppose that both t_i and c_i are independent across all individuals.

Explanatory variables are defined to give context through time to each of the individuals, and these will be represented by left continuous functions $x_i : \mathbb{R}_+ \to \mathbb{R}^d$. The explanatory variables $x_i(t) \in \mathbb{R}^d$ combine covariates of the individual $i \in \{1, \dots, N\}$ at time $t \geq 0$. It's important to note that we use the variable t to refer to time in general, while t_i represents the

specific event time of individual i. We assume that given x_i, a short notation for $(x_i(t))_{t\geq 0}$, the times t_i and c_i are conditionally independent. Additionally, we suppose t_i follows a continuous distribution of density $f(t|x_i,\tau_i)$ and c_i a continuous distribution of density $g(t|x_i,\tau_i)$. We have $g(t|x_i,\tau_i) = f(t|x_i,\tau_i) = 0$ for all $t < \tau_i$ since $t_i, c_i \geq \tau_i$ a.s.

In addition, we suppose that g satisfies the following property :

$$\forall t \geqslant \varepsilon > 0 : g(t|x_i,\tau_i) = g(t-\varepsilon|x_i,\tau_i-\varepsilon)\,.$$

Note that this assumption is also necessary for the density function f. However, as we will know its specific shape, the property is inherently satisfied. Finally, we denote by I_d the identity matrix of dimension d.

2.2.2 Survival probability

The objective of survival analysis is to predict the length of time until a specified event occurs. Consequently, it is necessary to estimate the distribution of these events. We define the survival probability function of individual i to be the complement of the cumulative distribution, that is, $S(t|x_i,\tau_i) = 1 - \int_{\tau_i}^{t} f(s|x_i,\tau_i)ds$, which can also be expressed as the probability of surviving up to time t :

$$S(t|x_i,\tau_i) = \mathbb{P}(t_i \geqslant t|x_i,\tau_i)\,, \qquad t \geq 0\,.$$

To estimate this function, it is common to assume a particular shape for the hazard function. The hazard function is defined as :

$$H(t|x_i,\tau_i) = -\frac{\partial}{\partial t}\log(S(t|x_i,\tau_i))\,, \qquad t \geq 0\,,$$

which represents the instantaneous risk of the event occurring at time t. Notably, we can derive the survival function from the hazard function :

$$S(t|x_i,\tau_i) = \exp\left(-\int_0^t H(s|x_i,\tau_i)ds\right)\,, \qquad t \geq 0\,.$$

For more details on event times distributions, refer to Cox and Oakes [25].

2.2.3 Likelihood

In order to estimate the survival probability we suppose the hazard function is a function of a specified parametric family Θ given the explanatory variables. The parameters will be determined following the likelihood principle observing $(u_i,\delta_i) \in \mathbb{R}_+ \times \{0,1\}$ and knowing (x_i,τ_i). As usual we implicitly make the assumption of non-informative censoring (see Kalbfleisch et al. [74]), which means that the censored distribution does not involve the parameter θ.

As mentioned earlier, some models assume a specific shape for the hazard function, such as additive, exponential, logistic or Weibull (see Cox and Oakes [25]). In this paper, we assume that the hazard function is exponential, and we detail this assumption below.

Definition 1 (Log-linear regression model for the Hazard function). We assume that there exist a vector $\theta \in \mathbb{R}^d$, such that the hazard function satisfies for all $t \geqslant 0$ and all $x_i : \mathbb{R}_+ \to \mathbb{R}^d$,

$$H(t|x_i,\tau_i) := h\big(\theta^T x_i(t)\big)\mathbb{1}\{t \geq \tau_i\}\,, \qquad t \geq 0\,,$$

where $h : x \in \mathbb{R} \mapsto \exp(x)$ is the response function.

By using this exponential model we obtain a formula to compute the negative log-likelihood which is the function that we aim to minimize.

Proposition 1. Under the exponential model from Definition 1 and omitting additional constants, the negative log-likelihood function $\ell : \Theta \to \mathbb{R}$ can be written in the following way :

$$\ell(\theta) = \sum_{i=1}^{N} -\delta_i \theta^T x_i(u_i) + \int_{\tau_i}^{u_i} \exp(\theta^T x_i(s))ds\,. \tag{2.1}$$

We call this function the complete log-likelihood and the proof of this proposition is detailed in Appendix B1.

2.2.4 Sequential likelihood optimization

We consider a horizon time n and a time partition $(t-1, t]$ with discrete time $t = 1, 2, \dots$ that is independent of the observations $(u_i, \delta_i)_{1 \leq i \leq N}$. In many real-life situations, data continues to evolve ; new patients may arrive, some patients may leave, and the optimization algorithm may need to update its estimation as new information becomes available. This is the focus of our work : to update online convex optimization algorithms for sequential survival data.

For individual i we define $y_{it} := \delta_i \mathbb{1}\{t-1 < u_i \leqslant t\}$ which indicates whether an event is observed for individual i during the interval $(t-1, t]$ or not. Additionally, we denote the risk indicator as $r_{it} := \mathbb{1}\{\tau_i \leqslant t, u_i > t-1\}$ for event i in the interval $(t-1, t]$. Then, we define the log-likelihood on the interval $(t-1, t]$ by the expression

$$\ell_t(\theta) := \sum_{i=1}^{N} -y_{it}\theta^T x_i(u_i) + r_{it} \int_{\tau_i \vee (t-1)}^{u_i \wedge t} \exp(\theta^T x_i(s))ds\,, \qquad \theta \in \Theta,\ t = 1, 2, \dots\,, \tag{2.2}$$

where we remind $u_i \wedge t = \min\{u_i, t\}$ and $\tau_i \vee (t-1) = \max\{\tau_i, t-1\}$. Let us notice that, analogous to Equation (2.1), the contribution to the log-likelihood of an individual that experiences an event in the interval $(t-1, t]$ is given by $\theta^T x_i(u_i) + \int_{\tau_i \vee (t-1)}^{u_i} \exp(\theta^T x_i(s))ds$, and the contribution of an individual that is censored in the interval $(t-1, t]$—either by $u_i = c_i$ or by t—is $\int_{\tau_i \vee (t-1)}^{u_i \wedge t} \exp(\theta^T x_i(s))ds$. If an individual is not yet present in the interval, i.e., $\tau_i > t$, or its observed time has passed before the beginning of the interval ($u_i \leqslant t-1$), its contribution to the log-likelihood is zero.

Finally, the log-likelihood up to time n is given by :

$$\ell^n(\theta) := \sum_{t=1}^{n} \ell_t(\theta), \qquad \theta \in \Theta.$$

It is important to notice that if n is sufficiently large, i.e., $n \geq u_i$ for every $1 \leq i \leq N$, and all the events have been observed, the complete log-likelihood of Equation (2.1) corresponds to the sum of all the interval contributions. Therefore, $\ell(\theta) = \ell^n(\theta)$, $\theta \in \Theta$, for n sufficiently large when N is finite.

2.3 Online convex optimization

2.3.1 Setting

A convex optimization problem consists of approximating the minimum of a convex function over a convex set. This problem can be extended to a recursive setting where, at each iteration t, a convex optimization algorithm predicts the parameter θ_t and incurs a loss of $\ell_t(\theta_t)$. This approach is particularly good in situations where the data evolves over time, requiring fast adaptation and decision making. We apply this methodology to survival analysis, introducing a novel perspective in a field traditionally dominated by batch processed data.

The online convex optimization algorithm aims to minimize its regret at any horizon time $n \geq 1$:

$$Regret_n := \sum_{t=1}^{n} \ell_t(\theta_t) - \min_{\theta \in \Theta} \sum_{t=1}^{n} \ell_t(\theta).$$

In this paper, we aim to optimize the losses $\ell_t(\theta)$ from Equation (2.2). To apply online convex optimization algorithms, we must first assume that $\Theta \subseteq \mathbb{R}^d$ is a non-empty, convex, bounded, and closed set. Subsequently, we verify the convexity of the objective function. Here the choice of the response function h is crucial. For $h(x) = \exp(x)$ the cost function $\ell_t(\theta)$ is defined in Equation (2.2) for every iteration t. We derive its gradient and Hessian :

$$\nabla \ell_t(\theta) = \sum_{i=1}^{N} -y_{it} x_i(u_i) + r_{it} \int_{\tau_i \vee t-1}^{u_i \wedge t} \exp(\theta^T x_i(s)) x_i(s) ds, \tag{2.3}$$

and

$$\nabla^2 \ell_t(\theta) = \sum_{i=1}^{N} r_{it} \int_{\tau_i \vee t-1}^{u_i \wedge t} \exp(\theta^T x_i(s)) x_i(s) x_i(s)^T ds \succcurlyeq 0. \tag{2.4}$$

The positive semi-definite Hessian confirms the convexity of the losses. Additionally, we formalize the boundedness assumption.

Assumption 1 (Bounded domain and gradient). There exists $D, G > 0$ such that for all $t = 1, 2, \ldots$ and $\theta \in \Theta$, $\|\theta\| \leqslant D$ and $\|\nabla \ell_t(\theta)\| \leqslant G$.

One of the most ancient algorithms for online convex optimization is named "follow the leader" (FTL), and it consists of choosing, at each iteration t, the point that optimizes the cumulative loss up to $t-1$. This algorithm does not satisfy any non-trivial regret guarantee for linear losses. However, under some modifications, like the randomized version proposed by Hannan [57], it can achieve an $\mathcal{O}(\sqrt{n})$ regret bound. Additionally, the approach from Cesa-Bianchi and Lugosi [18], where the losses are strongly convex, achieves a logarithmic regret in the number of iterations.

In 2003, Zinkevich [145] proposed a sequential version of the gradient descent algorithm (OGD), which satisfies a uniform regret bound of $\mathcal{O}(\sqrt{n})$ for an arbitrary sequence of convex cost functions and under the previous conditions (bounded gradients and domain). Later, Hazan et al. [65] proved that Zinkevich's algorithm attains a $\mathcal{O}(\log(n))$ regret for an arbitrary sequence of strongly convex functions (with bounded first and second derivatives). They also introduced an online version of the Newton-Raphson method, which they named the Online Newton Step (ONS), and demonstrated that it also achieves logarithmic regret. More algorithms and details can be found in Hazan [63].

We implement the ONS algorithm to minimize the negative log-likelihood and study the selection of its hyperparameters along with its regret bounds.

2.3.2 Exp-concavity and directional derivative condition

To ensure a logarithmic regret bound, the loss function must satisfy specific conditions. First, we review the definition of exp-concavity.

Definition 2. (Exp-concavity) A convex function $\ell : \Theta \to \mathbb{R}$ is μ-exp-concave iff the function $p(\theta) := \exp(-\mu\ell(\theta))$ is concave.

This property is fundamental in the regret analysis and replaces the strong convexity condition required by the OGD algorithm. This means that the ONS algorithm requires a weaker hypothesis on the losses $(\ell_t)_{t=1,2,\dots}$, to achieve logarithmic regret. Furthermore, we introduce a study based on this weaker condition, which is essential to derive the regret bound described by Hazan [63] in survival analysis.

Definition 3. (Directional derivative condition – DDC) We say a function $\ell : \Theta \to \mathbb{R}$ satisfy the directional derivative condition for a constant $\gamma > 0$ if for any pair $\theta_1, \theta_2 \in \Theta$

$$\ell(\theta_2) \geqslant \ell(\theta_1) + \nabla\ell(\theta_1)(\theta_2 - \theta_1) + \frac{\gamma}{2}\left(\nabla\ell(\theta_1)(\theta_2 - \theta_1)\right)^2. \tag{DDC}$$

To determine the directional derivative constant γ, we must first compute the exp-concavity constant μ.

Lemma 1. A twice differentiable function $\ell : \Theta \to \mathbb{R}$ is μ-exp-concave iff

$$\nabla^2\ell(\theta) \succcurlyeq \mu\nabla\ell(\theta)\nabla\ell(\theta)^T, \qquad \theta \in \Theta. \tag{2.5}$$

This holds with

$$\mu \leq \min_{\theta\in\Theta} \frac{\nabla\ell(\theta)^T\nabla^2\ell(\theta)\nabla\ell(\theta)}{||\nabla\ell(\theta)||^4}.$$

This lemma provides us a way for calculating the exp-concavity constant μ. The proof of Lemma 1 can be found in Appendix B2. Given a μ-exp-concave function ℓ, we can also determine its directional derivative constant γ. We have the following bound :

Lemma 2. A μ-exp-concave function $\ell : \Theta \to \mathbb{R}$, satisfying Assumption 1, admits a directional derivative constant $\gamma > 0$ satisfying

$$\gamma \leq \min_{\theta\in\Theta} \frac{-\frac{2}{\mu}\log(1 + \mu||\nabla\ell(\theta)||D) + ||\nabla\ell(\theta)||D}{(||\nabla\ell(\theta)||D)^2}.$$

We note that this lower bound improves upon the upper bound provided by Hazan [63] :

$$\gamma \leq \frac{1}{2}\min\left\{\frac{1}{GD}, \mu\right\},$$

and the proof of Lemma 2 can also be found in Appendix B2.

2.3.3 Online Newton Step

The ONS algorithm is an online analogue of the Newton-Raphson method ; see Ypma [140] for more details. The Newton-Raphson algorithm moves in the direction of the inverse of the Hessian multiplied by the gradient. For exp-concave loss functions ℓ_t with $t = 1, 2, \ldots$, we can replace the Hessian matrix with an approximation of it :

$$A_t = \sum_{k=1}^{t} \nabla \ell_k(\theta_k) \nabla \ell_k(\theta_k)^T.$$

At each iteration, the algorithm updates the estimation of the parameter as follows :

$$\theta_{t+1} = \theta_t - \frac{1}{\gamma} A_t^{-1} \nabla \ell_t(\theta_t)\,,$$

where γ is an algorithm hyperparameter denoting the learning rate and its optimal selection aligns with the DDC constant. This might lead to a point outside the convex set Θ and so we need to project it back. This projection is somewhat different than the standard projection as it is characterized by the norm defined by A_t instead of the Euclidean norm. The iteration step of the algorithm is :

$$\theta_{t+1} = \mathrm{Proj}_t\Big(\theta_t - \frac{1}{\gamma} A_t^{-1} \nabla \ell_t(\theta_t)\Big)\,,$$

where $\mathrm{Proj}_t(\theta^*) \in \arg\min_{\theta \in \Theta} (\theta - \theta^*)^T A_t (\theta - \theta^*)$.

Let us remark that ONS requires to invert a large matrix A_t, and in order to avoid expensive calculations, we consider the Sherman-Morrisson formula [120] which provides a recursion for A_t^{-1} from $A_0^{-1} := (1/\epsilon) I_d$:

$$A_t^{-1} = A_{t-1}^{-1} - \frac{A_{t-1}^{-1} \nabla \ell_t(\theta_t) \nabla \ell_t(\theta_t)^T A_{t-1}^{-1}}{1 + \nabla \ell_t(\theta_t) A_{t-1}^{-1} \nabla \ell_t(\theta_t)^T}\,, \qquad t = 1, 2, \ldots.$$

We formally describe the Online Newton Step algorithm 3 in Appendix B2. Hazan [63] proved the following regret bound of ONS.

Theorem 1 (Hazan [63]). Let us consider the losses $\ell_t : \Theta \to \mathbb{R}$ μ-exp-concave and satisfying Assumption 1. Then, Algorithm 3 with hyperparameters $\gamma = \frac{1}{2}\min\{\frac{1}{GD}, \mu\}$ and $\epsilon = (\gamma D)^{-2}$ satisfies $Regret_n \leqslant \gamma^{-1} d \log(2nG^2\gamma^2 D^2)$ for any $n \geqslant 4$.

Let us remind that we want to apply ONS algorithm to the losses $(\ell_t)_{t=1,2,\ldots}$ described in Equation (2.2), where we assume the exponential model defined in 1. The exp-concavity property is fundamental in the regret analysis of ONS. We first notice that we can work under (DDC) rather than μ-exp-concavity, focusing our work on the study of the constant γ which is the hyperparameter of ONS, rather than on μ. Then we see that the choice of this constant is very sensitive to variations in the gradients, which depend on the number of people at risk at each time. If μ is small, which can happen when the gradient of the loss is small, the choice of $\gamma = \frac{1}{2}\min\{\frac{1}{GD}, \mu\}$ proposed by Hazan [63] will also be small, potentially exploding the regret bound and causing issues with the convergence. Additionally, to properly tune the hyperparameter γ we need to know the exp-concavity constant in advance, but this constant might depend on the gradient of the losses that are not known before running the algorithm. Adjusting γ is not trivial and we provide some insights in the following sections.

2.4 Stochastic setting

The first solution we propose is to use a stochastic approach to bound the regret of ONS. We present the general stochastic setting introduced by Wintenberger [139] and apply one of its results to the survival case. The main difficulty in sequential survival analysis is the intrinsic time dependence in the loss functions $(\ell_t)_{t=1,2,\dots}$. Indeed, even if the individuals are iid, the log-likelihoods ℓ_t are dependent because of the individuals that are at risk during consecutive time intervals $(t-1,t]$ for $t=1,2,\dots$.

2.4.1 Stochastic Model

We model the arrival times $\tau_i \geq 0$ as a homogeneous Poisson process with intensity λ; see Kingman [80] for a reference textbook on the subject. For each $t>0$, we define the count random variable $N_t := \sum_{i=1}^{\infty} \mathbb{1}\{\tau_i \leqslant t\}$ which represents the number of individuals that arrive before t, and τ_{N_t} represents the arrival time of the last individual arriving before t. We assume a constant rate λ, such that $\mathbb{E}[N_t] = \lambda t$, indicating the average number of individuals arriving at time t. Additionally, in this section, we consider the covariate functions to be constant, i.e., $x_i(t) = x_i$ for all $t>0$, and that they follow, independently, the distribution of a random variable X. In this stochastic setting we rewrite the loss function :

$$\ell_t(\theta) = \sum_{i=1}^{N_t} -y_{it}\theta^T x_i + r_{it}\exp(\theta^T x_i)\big((u_i \wedge t) - (\tau_i \vee (t-1))\big), \tag{2.6}$$

where we replaced N by N_t in Equation (2.2). It is important to note that the derivation of this expression is based on the assumption of the exponential model 1. Now, we want to apply ONS to optimize this loss and study what happens with its stochastic regret.

For each iteration $t=1,2,\dots$, we consider the stochastic loss ℓ_t and the filtration $\mathcal{F}_t$ of σ-algebras such that the predictions of the online learning algorithm θ_t and the past losses $(\ell_s)_{s=1}^{t-1}$ are $\mathcal{F}_{t-1}$-measurable. To simplify notation, we use $\mathbb{E}_t[\cdot]$ to represent the conditional expectation given $\mathcal{F}_t$, denoted as $\mathbb{E}[\cdot|\mathcal{F}_t]$. In this context, our objective is to minimize the stochastic regret at any horizon time $n \geq 1$:

$$Risk_n := \sum_{t=1}^{n} L_t(\theta_t) - \min_{\theta\in\Theta} \sum_{t=1}^{n} L_t(\theta)\,,$$

where $L_t(\theta_t)$ is the conditional risk, defined as $L_t(\theta_t) := \mathbb{E}_{t-1}[\ell_t(\theta_t)]$ for $t=1,2,\dots$. Let us notice that in our case, where the stochastic losses ℓ_t are defined in Equation (2.6), the σ-algebra $\mathcal{F}_t$ is generated by y_{is}, x_i, τ_i, and $u_{is} = \min\{u_i, s\}$ for all $i=1,\dots,N_{t-1}$ and $s=1,\dots,t-1$.

The main difference with the setting presented in Section 2.3 is the use of the conditional risk L_t instead of the loss functions ℓ_t in the calculation of regret. This allows us to relax the convexity conditions imposed on ℓ_t and instead focus on the convexity properties of L_t.

2.4.2 Stochastically Exp-Concavity

It was proved in Wintenberger [139] that the ONS algorithm achieves a $\mathcal{O}(\log(n))$ stochastic regret bound under a stochastic exp-concavity condition for ℓ_t which is described below.

Definition 4 (Stochastic exp-concavity). A sequence of random functions $(\ell_t)_{t=1,2,\dots}$ is said to be γ stochastically exp-concave with respect to a filtration $\mathcal{F}_t$ if for all $\theta_1, \theta_2 \in \Theta$ and $t = 1, 2, \dots$

$$L_t(\theta_1) \leqslant L_t(\theta_2) + \nabla L_t(\theta_1)^T(\theta_1 - \theta_2) - \frac{\gamma}{2}\mathbb{E}_{t-1}\left[(\nabla \ell_t(\theta_1)^T(\theta_1 - \theta_2))^2\right], \qquad a.s.$$

Let us note that this property corresponds to the stochastic counterpart of the directional derivative condition (DDC). This property plays a crucial role in the proof of Theorem 7 of Wintenberger [139], which establishes the logarithm stochastic regret bound. However, the losses ℓ_t defined in (2.6) do not satisfy this property. Nevertheless, we demonstrate that the events where this inequality is not fulfilled have a small probability and therefore, we can still bound the stochastic regret. In addition, we need to make the following design assumption.

Assumption 2. There exist $A > 0$ such that $\mathbb{E}[xx^\top \mathbb{1}\{T \leqslant C\}(1-T)_+ | \tau = 0] \succcurlyeq AI_d$.

This assumption is not trivial, and it is not always satisfied ; however, when all the individuals experience an event and $T \leqslant 1$, it corresponds to a classical design. When $t \geq 1$ an alternative analyses is required.

2.4.3 Stochastic Regret

To apply Theorem 7 from Wintenberger [139], the losses need to satisfy certain hypothesis, among which are stochastic exp-concavity and a stochastic bound on the gradients of the losses. We prove that our losses ℓ_t, whose do not satisfy exactly the conditions of Theorem 7, still leads ONS algorithm to achieve a logarithmic stochastic regret. We present the result in the following theorem.

Theorem 2. Given $\varrho > 0, n \geq 1$ and the stochastic losses $(\ell_t)_{t=1,2,\dots}$ from Equation (2.6), then under Assumption 2, a bounded domain of diameter D and hyperparameter γ, the stochastic exp-concavity constant, the ONS algorithm has logarithmic stochastic regret with probability $1 - 4\varrho$. Specifically, $Risk_n = \mathcal{O}(\log(n/\varrho)/\gamma)$, $n \geq 1$.

The proof of Theorem 2 can be found in Appendix B3 and the explicit regret bound in Equation (B.4). To finish, we prove the following corollary.

Corollary 1. Given $\varrho > 0, n \geq 1$ and the stochastic losses $(\ell_t)_{t=1,2,\dots}$ from Equation (2.6), we consider θ_t the ONS prediction at time t and $\bar{\theta}_n$ the average prediction $\bar{\theta}_n = \frac{1}{n}\sum_{t=1}^n \theta_t$. Defining the optimal parameter

$$\theta^* = \arg\min_{\theta \in \Theta} \sum_{t=1}^n L_t(\theta),$$

then, under Assumption 2, a bounded domain of diameter D and hyperparameter γ, with probability $1 - 4\varrho$ we have :

$$||\bar{\theta}_n - \theta^*||^2 \leqslant \mathcal{O}\left(\frac{\log(n/\varrho)}{\gamma n}\right), n \geq 1.$$

This corollary ensures the convergence of the algorithm predictions to the real parameter, which is possible thanks to the strong convexity of the risk functions L_t. It is important to remark that this does not hold in the deterministic setting. The proof can be found in Appendix B3 and the explicit bound in Equation (B.5).

2.5 Survival ONS algorithm

As mentioned earlier, the choice of γ has a significant influence on the algorithm's performance, particularly regarding the regret bound. To avoid convergence issues and address the challenge posed by the small optimal constant proposed by Hazan [63], we propose an adaptive setting that allows us to select the most suitable learning rate at each step while maintaining control over the regret bound. We introduce SurvONS (Algorithm 1), a survival version of MetaGrad from van Erven et al. [133], that uses Bernstein Online Aggregation (BOA, introduced in Wintenberger [138]) to aggregate multiple ONS applied to an adaptive auxiliary function. SurvONS strategically selects larger learning rates to handle sub-optimal parameters. The key difference between our algorithm and MetaGrad lies in the approach to updating the adaptive learning rate. We explain this algorithm in detail throughout this section.

2.5.1 Recursive adaptation to the constants

We present first the recursive adaptation of the constants μ and γ. Lemma 1 provides a bound for the exp-concavity constant μ, and Lemma 2 offers a bound for the directional derivative constant γ based on μ. We aim to apply this approach to $\ell = \ell_t$ for all $t = 1, 2, \ldots$, and recursively obtain μ_t and $\gamma_t(\mu_t)$ such that they satisfy the bounds of Lemma 1 and Lemma 2.

In Hazan's approach, as described in [63], the idea is to select a universal constant μ that renders all the functions $(\ell_t)_{t=1,2,\ldots,}$ μ-exp-concave. The natural choice would be to take :

$$\mu := \min_{t\in\{1,\ldots,n\}} \mu_t^*, \qquad \text{where} \qquad \mu_t^* := \min_{\theta\in\Theta} \frac{\nabla\ell_t(\theta)^\top \nabla^2\ell_t(\theta)\nabla\ell_t(\theta)}{||\nabla\ell_t(\theta)||^4},$$

is the bound given by Lemma 1. With this configuration, we guarantee exp-concavity for every function. However, the challenge of minimizing over the parameter set Θ in the definition of μ_t^* might be more intricate than minimizing the loss function ℓ_t. In addition, we can not know the constant in advance because ℓ_t is revealed at the t-th iteration only in our online setting.

To solve this problem, we define at each time $t = 1, 2, \ldots$ an adaptive estimation of the exp-concavity constant :

$$\mu_t := \frac{\nabla\ell_t(\theta_t)^\top \nabla^2\ell_t(\theta_t)\nabla\ell_t(\theta_t)}{||\nabla\ell_t(\theta_t)||^4},$$

and, similarly from Lemma 2,

$$\gamma_t(\mu_t) := \frac{-\frac{2}{\mu_t}\log(1+\mu_t||\nabla\ell_t(\theta_t)||D) + 2||\nabla\ell_t(\theta_t)||D}{(||\nabla\ell_t(\theta_t)||D)^2},$$

where θ_t is the parameter predicted by the algorithm at time t. Let us notice that this choice of $\mu_t \geqslant \mu$ and $\gamma_t(\mu) \geqslant \gamma(\mu)$ assures the exp-concavity and the directional derivative condition for ℓ_t close to θ_t at time t. We sometimes refer to $\gamma_t(\mu_t)$ as γ_t when the specification is not necessary.

2.5.2 SurvONS

Now, we have an adaptive way to choose μ_t and γ_t that preserves the exp-concavity properties at each iteration. However, this choice might not be optimal, in some iterations the gradient $\nabla\ell_t$ can be close to zero due to the lack of individuals at risk, and this might lead to numeric problems setting μ_t and γ_t. Thus we propose an intermediate choice of the learning rate. Given a user specified constant $\gamma > 0$, we define for each time $t = 1, 2 \dots$:

$$\tilde{\gamma}_t := \max\{\gamma_t(\mu_t)/4, \gamma\}\,,$$

a value that chooses a portion of the optimal directional derivative condition constant $\gamma_t/4$ when it is not too small, and the user specified constant γ when the quarter of the optimal constant decreases under γ. This choice $\tilde{\gamma}_t$ is a trade off in between choosing the optimal directional derivative condition constant and a worse constant when the optimal one is susceptible to bring convergence problems.

In order to keep the logarithmic regret bound we cannot directly use the adaptive choice of the constant as the algorithm's learning rate. Instead, it was proposed by van Erven et al. [133] to optimize an adaptive auxiliary function. Let us consider $\hat{\theta}$ such that $\ell_t(\hat{\theta})$ and $\nabla\ell_t(\hat{\theta})$ have been observed and $\gamma > 0$, we define the directional derivative function :

$$\hat{\ell}_{t,\gamma}(\theta) := \ell_t(\hat{\theta}) + \nabla\ell_t(\hat{\theta})(\theta - \hat{\theta}) + \frac{\gamma}{2}\left(\nabla\ell_t(\hat{\theta})(\theta - \hat{\theta})\right)^2, \qquad \theta \in \Theta\,, \qquad t = 1, 2, \dots\,. \tag{2.7}$$

We prove that this function satisfies the directional derivative condition for a different constant $\hat{\gamma}$.

Lemma 3. Let $\gamma > 0$, $\Theta \subseteq \mathbb{R}^d$ of diameter $D > 0$, $\hat{\theta} \in \Theta$ and $\ell_t : \Theta \to \mathbb{R}$ the log-likelihood defined in Equation (2.2). Then, the function $\hat{\ell}_{t,\gamma}$ from (2.7) satisfies for every $\theta_1, \theta_2 \in \Theta$:

$$\begin{aligned}\hat{\ell}_{t,\gamma}(\theta_2) \geqslant \hat{\ell}_{t,\gamma}(\theta_1) + \nabla\hat{\ell}_{t,\gamma}(\theta_1)(\theta_2 - \theta_1) \\ + \frac{\gamma}{2(1 + \tilde{\gamma}\nabla\ell_t(\hat{\theta})(\theta_1 - \hat{\theta}))^2}\left(\nabla\hat{\ell}_{t,\gamma}(\theta_1)(\theta_2 - \theta_1)\right)^2,\end{aligned}$$

and thus, the function $\hat{\ell}_{t,\gamma}$ has directional derivative constant $\hat{\gamma}$ with $\hat{\gamma} := \frac{\gamma}{2(1+\gamma D||\ell_t(\hat{\theta})||)^2}$.

The proof of Lemma 3 is presented in Appendix B4. The idea of the algorithm is to use ONS routine to optimize the functions $\hat{\ell}_{t,\gamma} = \hat{\ell}_{t,\tilde{\gamma}_t}$, i.e., the auxiliary function with $\gamma = \tilde{\gamma}_t$, which adapt at each step according to the current optimal γ_t and the algorithm predictions θ_t.

In addition, to obtain an algorithm that is robust for the choice of the learning rate, we propose an aggregation procedure which applies ONS and combines it with multiple choices of the learning rate γ. To formalize this idea, we consider a grid $\Gamma = \{\gamma_i\}_{i=1,\dots,K}$ and $\mathcal{E} = \{\epsilon_i\}_{i=1,\dots,K}$ such that $\epsilon_i = \frac{1}{(\gamma_i D)^2}$ for all $i = 1, \cdots, K$. Then, at each iteration $t = 1, \dots, n$, and for each $i = 1, \dots, K$, we define $\tilde{\gamma}_{it} = \max\{\gamma_t/4, \gamma_i\}$ and we aggregate ONS applied to $(\hat{\ell}_{t,\tilde{\gamma}_{it}})_{t=1,2,\dots}$. The aggregation is held by BOA algorithm of Wintenberger [138], which is a recursive procedure that considers exponential weights with a second order refinement. The algorithm SurvONS is described in Algorithm 1 and it is important to notice that the difference between SurvONS and MetaGrad is the choice of the constant $\tilde{\gamma}$.

Aggregation methods allow us to avoid bad choices of γ and therefore, the convergency issues. Let us remind that we consider the exponential model 1. We prove that the regret of Algorithm 1 is bounded.

Algorithm 1 SurvONS

Input : $(\ell_t)_{t=1,2,\dots}, D > 0, n \geqslant 1$, grids $\Gamma, \mathcal{E}$
Initialization : for each γ_k in Γ : $\theta_0(\gamma_k) \in \Theta$, $\pi_{0,k} = \frac{1}{K}$, $\hat{\theta}_0 \in \Theta$, $A_0^{-1} = \mathcal{E}^{-1}\mathbb{1}_d$
for iteration $t = 1, \dots, n$ **do**

Update : $\hat{\theta}_t = \sum_{k=1}^{K} \pi_{t,k}\, \theta_t(\gamma_k)$

Observe :
$$\nabla \ell_t(\hat{\theta}_t) = \sum_{i=1}^{N} -y_{it} x_i(u_i) + r_{it} \int_{\tau_i \vee t-1}^{u_i \wedge t} \exp(\hat{\theta}_t^T x_i(s)) x_i(s) ds$$
$$\nabla^2 \ell_t(\hat{\theta}_t) = \sum_{i=1}^{N} r_{it} \int_{\tau_i \vee t-1}^{u_i \wedge t} \exp(\hat{\theta}_t^T x_i(s)) x_i(s) x_i(s)^T ds$$
$$\mu_t = \frac{\nabla \ell_t(\hat{\theta}_t)^T \nabla^2 \ell_t(\hat{\theta}_t) \nabla \ell_t(\hat{\theta}_t)}{||\nabla \ell_t(\hat{\theta}_t)||^4}$$
$$\gamma_t = 2 \frac{-\frac{1}{\mu_t} \log(1 + \mu_t ||\nabla \ell_t(\hat{\theta}_t)||D) + ||\nabla \ell_t(\hat{\theta}_t)||D}{(||\nabla \ell_t(\hat{\theta}_t)||D)^2}$$

for $\gamma_k \in \Gamma$ **do**

Observe :
$$\tilde{\gamma}_t = \max\{\gamma_t/4, \gamma_k\}$$
$$\nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) = \nabla \ell_t(\hat{\theta}_t)(1 + \tilde{\gamma}_t \nabla \ell_t(\hat{\theta}_t)(\theta_t(\gamma_k) - \hat{\theta}_t))$$

Recursion :

$$A_t^{-1} = A_{t-1}^{-1} - \frac{A_{t-1}^{-1} \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k))^T A_{t-1}^{-1}}{1 + \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) A_{t-1}^{-1} \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k))^T}$$

$$\theta_{t+1}(\gamma_k) = \mathrm{Proj}_t \left(\theta_t(\gamma_k) - \frac{1}{\gamma_k} A_t^{-1} \nabla \hat{\ell}_{t,\tilde{\gamma}_t}(\theta_t(\gamma_k)) \right)$$

end for
Update : $\pi_{t+1,\cdot} = \pi_t \exp\left(-\Gamma \nabla \ell_t(\hat{\theta}_t)^T (\hat{\theta}_t - \theta_t(\Gamma)) - \Gamma^2 (\nabla \ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\Gamma)))^2 \right)$
end for
return $\hat{\theta}_n$

Theorem 3. Let $n \geq 1$, $(\ell_t)_{t=1,\dots,n}$ be the sequence of losses defined in (2.2), that are assumed to satisfy Assumption 1 and (DDC) with constants $\gamma_t \in (0, 1/GD)$. Let $K \geqslant 1$ and $\Gamma \in (0, 1/(4GD))^K$. Then, Algorithm 1 with hyperparameters Γ and $\mathcal{E} = 1/(\Gamma D)^2$, satisfies the regret upper-bound :

$$Regret_n \leqslant \min_{\gamma \in \Gamma} \left\{ \frac{2\log(K) + 5d\log(n)}{\gamma} + \gamma G^2 D^2 n_\gamma \right\},$$

where $n_\gamma := \sum_{t=1}^{n} \mathbb{1}\{\gamma_t < \gamma\}$, $\gamma > 0$.

This theorem provides a regret bound that proposes a trade-off between the bad choices of γ and the frequency with which the algorithm selects γ over γ_t, thereby compensating for the regret increment. The proof of Theorem 3 can be found in Appendix B4. Let us notice that this analysis is also valid for MetaGrad algorithm [133], and Theorem 3, which was developed for the survival losses (2.2), holds equally true for any loss satisfying Assumption 1 and (DDC).

TABLEAU 2.1 – Regret bound orders (up to logarithmic factors) after n iterations of different online optimization algorithms.

	OGD	ONS	SurvONS	ONS($\bar{\gamma}$)
Regret bound	$\sqrt{n}$	$\frac{1}{\min_{1\leqslant t\leqslant n}\gamma_t}$	$\min_\gamma\left\{\frac{1}{\gamma}+\gamma n_\gamma\right\}$	$\frac{1}{\bar{\gamma}_n}$

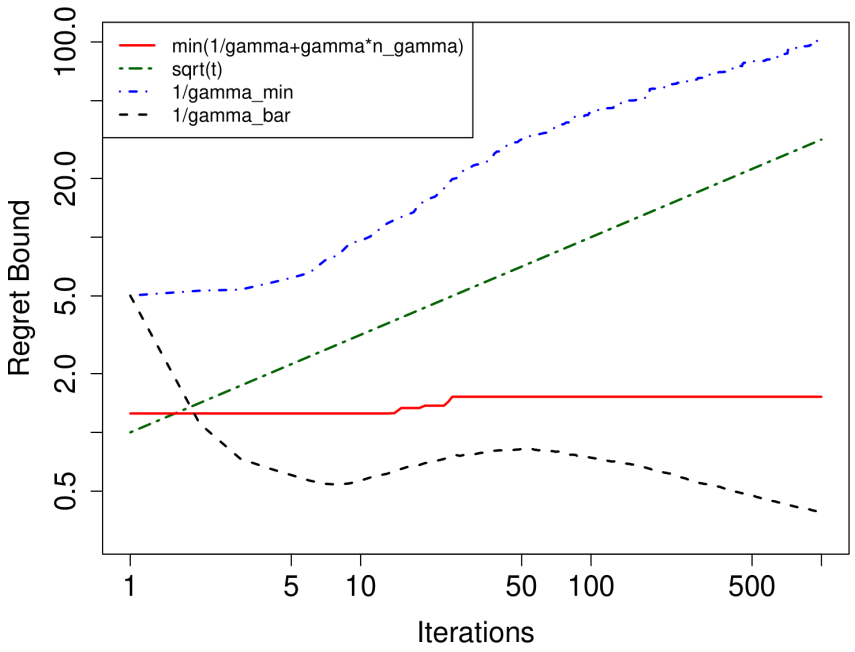


FIGURE 2.1 – Comparison of regret bound orders (up to logarithmic factors) for multiple online methods using simulated data.

2.5.3 Theoretical regret bounds comparison

We show in Figure 2.1 the differences between the regret bound orders, in order to illustrate the importance of the constant adaptation $\bar{\gamma}_t$ in SurvONS, and the interest of the stochastic setting. We compare the theoretical regret bound orders of ONS [65] with the optimal hyperparameter γ_t, OGD [145], SurvONS 1, and ONS with an average hyperparameter $\bar{\gamma}_t = \sum_{s=1}^{t}\gamma_s$, representing the stochastic approach. The bounds are detailed in Tableau 2.1.

In this comparison, we omit constants and logarithmic terms. We estimate γ_t with SurvONS, and we use these estimations to construct the bounds. The simulation framework for this experiment is detailed in Section 2.6. The graph is presented in log-log scale.

Figure 2.1 traces the regret behavior of the different algorithms (see Tableau 2.1). Without an explicit calculation of the stochastic constant, we show the interest of considering an average case through plotting the average constant $\bar{\gamma}_t$. We observe that although in theory, the bound of ONS appears better than the bound of OGD ($\mathcal{O}(\log(n)/\gamma)$ v/s $\mathcal{O}(\sqrt{n})$), when γ_t goes to 0, the bound

of ONS is not $\mathcal{O}(\log(n))$, but $\mathcal{O}(\log(n)/\min_t \gamma_t)$. A similar finding in logistic regression has been made rigorous by Hazan et al. [66] with the help of lower bounds matching $\mathcal{O}(\log(n)/\min_t \gamma_t)$. In practical applications, it is essential to consider more detailed analyses that remain robust in scenarios where $\min_t \gamma_t$ goes to 0, which is what we propose with SurvONS and the stochastic approach.

2.6 Simulation experiments

In this section we present simulation results of our method [1]. We considered a number of individuals $N = 10\,000$ and a number of iterations $n = 1\,000$. Then we sample a multivariate random normal of dimension $(N, d-1)$ with $d = 4$ and mean vector and covariance matrix :

$$\eta := \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \qquad \Sigma := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

We add an intersect column that transforms the matrix into one of dimension (N, d). This matrix corresponds to the covariates information $\{x_i\}_{i=1}^N$, which does not depend on time. The real parameter θ^* is set randomly following a $\mathcal{N}(0, I_d)$ distribution. We sample the arrival times τ_i as a uniform between 0 and n and we simulate T_i and C_i following an exponential distribution of rate $\exp(\theta^{*T} x_i)$,

$$T_i \sim \tau_i + \exp(\exp(\theta^{*T} x_i)), \qquad C_i \sim \tau_i + \exp(\exp(\theta^{*T} x_i)).$$

For more details on the common use of exponential distributions in survival analysis we refer to Selvin [118]. We repeat this procedure 100 times, and the results are the average curves over the 100 data simulations. Additionally, we consider two exponential grids for the aggregation methods of size $K = 10$. First, we test a random grid, consider the SurvONS predictions θ_t for $t = 1, \dots, n$, and define :

$$G := \max_{t=1,\dots,n} ||\nabla \ell_t(\theta_t)|| .$$

This process is repeated multiple times to ensure the stability of G estimation. Second, we choose 10 equidistant points and then we generate the grids by considering the exponential of each point, such that :

$$\Gamma_1 := (1/\sqrt{n}, \dots, 1/4GD) , \qquad \Gamma_2 := (1/GD, \dots, 10/GD) ,$$

where D is adjusted a posteriori such as $D := 1.1||\theta^*||$. Throughout this section we compare the results of the two choices of grid Γ_1 and Γ_2.

We observe in Figure 2.2 the distribution of the average γ_t estimations that we obtained from SurvONS. The average for Γ_1 is 1.24 and 1.64 for Γ_2. The similarity between both estimations elucidates the proximity of the graphs in Figure 2.1, which is unsurprising given that the directional derivative constant is inherent to the loss function and does not depend on the algorithm or the selected grids.

We compare SurvONS, described in Algorithm 1, with the BOA-ONS proposed by Wintenberger [138]. Additionally, we fit several ONS and OGD with constant learning rate equal to each γ in the grid, and then we select the one that performs better to include in the comparison.

1. See the codes at the GitHub repository : `https://github.com/camferna/Online-Learning-Approach-for-Survival-Analysis`

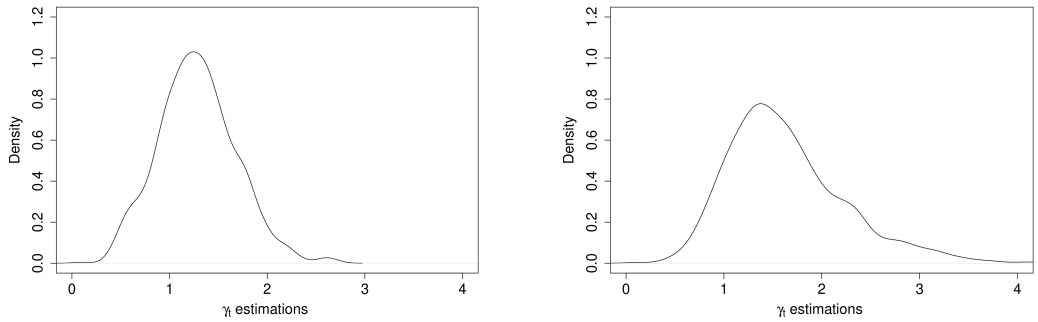


FIGURE 2.2 – Density of γ_t estimation obtained by Algorithm 1, with Γ_1 [left] and Γ_2 [right]

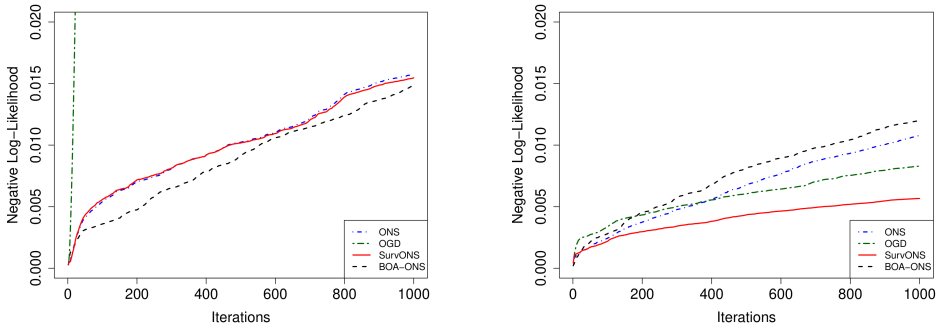


FIGURE 2.3 – Cumulative negative log-likelihood comparison of different online methods with hyperparameters in grid Γ_1 [left] and Γ_2 [right]

Remark that this procedure overestimates the performances of ONS and OGD. We show the average cumulative difference between the negative log-likelihood of the estimations and the real parameters in Figure 2.3.

In Figure 2.3, we observe that SurvONS (in purple) does not outperform BOA-ONS (in black) with the Γ_1 grid. However, the scenario changes with the second grid, Γ_2, where SurvONS proves to be more effective than the other methods. This unexpected result arises from the fact that the Γ_1 grid falls within the theoretical limits. Nevertheless, we observe a consistent improvement in performance for all algorithms when considering a larger grid. This discrepancy could arise from either an overestimation of the constant G or the presence of outlier points exhibiting extremely large gradients. Nonetheless, given the similarity in the constant γ_t estimation across the two grids, shown in Figure 2.2, we recommend opting for larger grids, ranging from 4 to 40 times the theoretical bound of $1/4GD$.

In addition, Figure 2.4 presents the quadratic error, where we consider the cumulative average of the estimations. Specifically, given a sequence of algorithm predictions $(\theta_s)_{s=1}^t$, the cumulative average is defined as $\bar{\theta}_t := t^{-1}\sum_{s=1}^t \theta_s$. Let us remind that the curves depicted represent the average of 100 instances obtained from simulating 100 datasets. The figure is in log-log scale.

Figure 2.4 corroborates the result of Corollary 1, which establishes the convergence of the

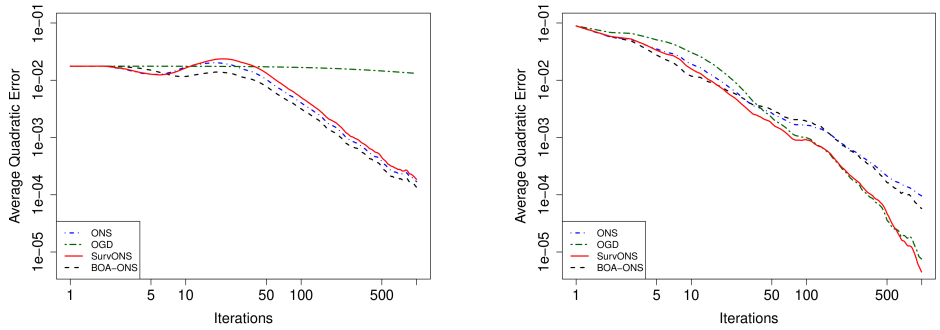


FIGURE 2.4 – Average quadratic error comparison of different online methods with hyperparameters in Γ_1 [left] and Γ_2 [right]

estimations θ_t to the real parameter θ^* when using the ONS algorithm. The findings of Wintenberger in [139], which demonstrate the $\mathcal{O}(\log(n))$ stochastic regret of BOA-ONS, together with the insights from Figure 2.4, suggest the potential to extend a similar corollary to both BOA-ONS and SurvONS. Furthermore, Corollary 1 can be easily extended to BOA-ONS by replacing the application of Theorem 2 with Theorem 4 from [139].

2.7 Conclusions

In this paper, we presented a detailed mathematical framework for online survival data, analyzing the regret of Online Newton Step and its sensitivity to the learning rate. Notably, we found that tuning this parameter is challenging, and the regret bound is highly sensitive to its adjustment. Our first contributions is introducing a stochastic setting to ensure that ONS achieves logarithmic stochastic regret in the survival context. Additionally, we proposed an adaptive method, SurvONS, which aggregates ONS with different learning rates. Adaptive methods, commonly used in first-order algorithms like AdaGrad [32] or Adam [79], offer a promising avenue for enhancing second-order algorithms. Our approach leverages adaptive strategies to improve efficiency and convergence, extending its applicability beyond the online survival domain. The regret analysis of SurvONS strategically selects larger learning rates to address sub-optimal parameters. In conclusion, aggregation methods enhance robustness in selecting algorithm hyperparameters ; however, achieving and maintaining fast rates remains a non-trivial task.

Finally, in the simulation experiments, we compared two grid choices. Figure 2.2 shows that γ_t estimations closely align within the grids, and the second grid produces values that do not approach zero to the same extent as the first grid. Additionally, Figure 2.3 indicates that choosing larger values for the learning rate grid accelerates convergence, suggesting the preference for larger grids.

Chapitre 3

Experimental Comparison of Ensemble Methods and Time-to-Event Analysis Models

Abstract

Time-to-event analysis is a branch of statistics that has increased in popularity during the last decades due to its many application fields, such as predictive maintenance, customer churn prediction and population lifetime estimation. In this paper, we review and compare the performance of several prediction models for time-to-event analysis. These consist of semi-parametric and parametric statistical models, in addition to machine learning approaches. Our study is carried out on three datasets and evaluated in two different scores (the integrated Brier score and concordance index). Moreover, we show how ensemble methods, which surprisingly have not yet been much studied in time-to-event analysis, can improve the prediction accuracy and enhance the robustness of the prediction performance. We conclude the analysis with a simulation experiment in which we evaluate the factors influencing the performance ranking of the methods using both scores.

3.1 Introduction

Time-to-event analysis is popular in medical research for predicting the lifetime of populations. It is also widely used in many fields in order to predict the time until a certain critical event occurs, which may be the recurrence of a disease, the customer churn in business management and operation research, recidivism in social science and psychology, the failure of machines in industrial engineering, etc. One of the most important characteristics of time-to-event analysis, which makes a significant difference from classical regression problems [144], [7], is a phenomenon known as censorship, and specifically, in this paper we treat the problem of right censorship. Right censorship arises from the fact that a study may finish before all the samples reach the

critical event or because some of the individuals have withdrawn from the study before it ends. As a result, not all the samples may have reached their failure time during the observed period, such that there will be a subset of them whose observed time will represent a lower bound for the critical time.

Many different models have been proposed in order to predict survival times. One of the most widely used ones was proposed by Cox [24] in 1972 ; this is a semi-parametric model which is composed of an unknown baseline hazard function that depends on the time and the effect of the covariates given by an exponential function. Extension of the Cox model can be found in Therneau and Grambsch [126]. In addition, generalized linear models (GLMs) have also played a crucial role in time-to-event analysis, providing a flexible framework for modeling survival data by relating the mean of the response variable to the linear predictors through a link function (more details in [97]). Later, other parametric techniques were proposed such as Aalen additive model [1], Weibull AFT [136] and the log-normal model [113]. Recently, machine learning methods have attracted much attention and many non-parametric model-based machine learning techniques for time-to-event analysis have been proposed, such as gradient boosting Cox [112], random survival forest [72] and survival support vector machine [107]. Later, deep neural network-based models such as DeepSurv [77], DeepHit [87] and DNNSurv [142] have significantly advanced the field. For more details on classical machine learning methods for time-to-event analysis, review Wang et al. [135], and more details on deep learning-based methods, refer to Wiegrebe et al. [137]. In this paper, we present a comparison of several of these models through two different scores, the concordance index [60] and the integrated Brier score [47], and among different types of data sets with the objective to study how the different models behave and compare their effectiveness.

Ensemble methods are learning algorithms that combine different models by optimizing certain weighting procedures in order to obtain a predictor that will be the combination of multiple learners. One of the main advantages of ensemble methods is the fact that they can inherit the good properties of each of the predictors and use them whenever it is most suitable, for example, if we have a dataset that behaves better for a particular type of models, then the weighting procedure will privilege this type of models and thus leads to an increment of accuracy that is independent of the chosen dataset. Note that this can be extended to time-varying weighting by which we can also take advantage of the fact that there are some models that vary their performance over time or over the distribution (see [10]), where we ponder differently the methods that are better for predicting distribution tails and the ones for predicting the center of the distributions. Ensemble methods are well known and used in many applications of data analytics and machine learning [141]. Classical examples in time-to-event analysis include tree-based models such as random survival forest [72] and the adaptive kernel survival estimator [20], boosted models such as gradient boosting Cox [112] and XGBoost [21], as well as combinations and variations of both models, like those studied by Hothorn et al. [70]. These methods aim to combine weak learners to enhance robustness, but they are not tailored to perform aggregation of generic models from different sources. This technique has not yet been widely explored in time-to-event analysis. Van der Laan et al. [132] proposed a method called the "Super Learner," which uses cross-validation techniques to create a weighted combination of multiple learners. This method was adapted to survival data by Golmakani and Polley [50], under the assumption that the individual algorithms are based on proportional hazards. Additionally, Debray et al. [30] studied model averaging and stacked regressions of existing clinical prediction models (CPM), particularly in scenarios with limited validation data. Their study is confined to the clinical field and aims to leverage well-known pre-trained clinical models.

The existing literature lacks clean performance comparison between time-to-event analysis methods and how to calibrate parameters. Van Wieringen et al. [134] reviewed the performance

of different methods applied to the particular case of gene expression data. The methods that are able to handle this type of problem, where the number of features exceeds by far the number of samples, are very specific and do not necessarily represent the general case of survival analysis problems.

Contributions. The main contribution of this paper is to give a detailed comparison of different and diverse time-to-event analysis methods using two widely used scores. The above gives us a detailed comparative study of the time-to-event analysis models and their different advantages and disadvantages. To this end, we compare the performance using three datasets and we study the impact of optimizing the hyperparameters through a randomized search. We observe that the method ranking varies across each dataset, making it challenging to select the most appropriate model without prior knowledge. To address this issue, we propose combining these different methods to enhance robustness across datasets. This is carried out by optimizing the parameters of a convex combination of the methods described in Section 3.2.1, such that the integrated Brier score is minimized. Finally, we conduct simulation experiments aimed at deepening insights from the dataset comparison and studying the factors influencing method performance ranking. We generate data using three different techniques under three scenarios : increasing the number of samples, reducing the number of features, and augmenting the percentage of censorship.

Paper outline. First, we present the preliminaries and definitions for our study, together with the implemented methods : Cox proportional hazard, Gradient boosting Cox, Random survival forest, Weibull accelerated failure time, Aalen's additive and DeepSurv. In Section 3.3, we exhibit our implementation of ensemble methods. In Section 3.4, we present the three datasets (Primary biliary cirrhosis, German breast cancer and Telecom churn) used for our study. Section 3.5 shows the comparison of the various techniques and their numerical results. Besides, we show the performance of the ensemble method. In Section 3.6, we present the simulation experiment and finally, Section 3.7 contains some concluding remarks.

3.2 Preliminaries

The main objective of time-to-event analysis is to estimate the distribution of survival times. Given a set of N subjects with its respective vector of covariates of dimension d, $x_i = \{x_i^1, \dots, x_i^d\} \in \mathcal{X},\ i \in \{1, \dots, N\}$, we assume that x_i follows the distribution of a random variable X_i. Let T_i and C_i be a non-negative random variable denoting the survival and censored time, respectively. Then, we define the observed time as $Y_i = \min\{T_i, C_i\}$ and we will write $\Delta_i = \mathbb{1}\{T_i \leqslant C_i\}$ for the survival indicator. Under these conditions, a subject of the dataset will be described by $(x_i, y_i, \delta_i) \in \mathcal{X} \times \mathbb{R} \times \{0, 1\}$ assumed to be a realization of the random variable (X_i, Y_i, Δ_i). In addition, we define the set of individuals at risk as $\mathcal{R}(t) = \{i \in \{1, \dots, N\} : y_i > t\}$. Let us remark that we consider that all the individuals are present at time $t = 0$. Then, the probability to survive at time t for subject i of the dataset is given by :

$$S(t|x_i) = \mathbb{P}(T_i > t|X_i = x_i).$$

In order to estimate the survival probability, many parametric and semi-parametric models

assume a particular shape of the hazard function, which is defined for all $t > 0$ as :

$$\begin{aligned} h(t|x_i) &= -\frac{\partial}{\partial t}\log(S(t|x_i)) \\ &= \lim_{dt\to 0}\frac{\mathbb{P}(t \leqslant T_i < t+dt | T_i \geqslant t, X_i = x_i)}{dt}. \end{aligned}$$

We can retrieve the survival probability function by integrating the exponential of the hazard function

$$S(t|x_i) = \exp\left(-\int_0^t h(u|x_i)du\right).$$

Each model will give us an estimator $\hat{S}$ of the survival probability S. In addition, we define the mortality risk of an individual by a function $R : \mathcal{X} \to \mathbb{R}_+$, which will be used later to compute the concordance index. The mortality risk must satisfy $R(x_i) > R(x_j)$ if $\mathbb{P}(T_i < T_j) > 1/2$, i.e. if individual i has a higher mortality risk than individual j. Note that R is not uniquely defined and only the ranking matters. Each model will define and estimate (by providing a function $\hat{R}$) the mortality risk differently and we give the details separately in Section C2. In addition, to measure the goodness of fit of each model, we consider two scores. Concordance Index [60] is a rank score that measures the ability of the model to correctly provide a reliable ranking of the survival times. And secondly, the integrated Brier score [47], which measures the calibration of the models by averaging the square distances between the observed survival status and the predicted survival probability. We give more details about both scores in Section C1.

3.2.1 Methods and their implementation

We consider six methods in our study. These are Cox proportional hazard [24], gradient boosting Cox [112], random survival forest [72], Weibull AFT [136], Aalen additive [1] and DeepSurv [77]. There exist many other methods for survival analysis, such as life tables [26], different versions of cox regressions [12], [62], linear regressions [127], Bayesian network classifier based methods [43] and support vector machine [78], see [135] for more details. Nevertheless, we choose the six methods mentioned above because they are the most popular and widely used techniques, they include parametric, semi-parametric and machine learning approaches, and on the other hand, the diversity of their structure is very relevant and has a key role in ensemble methods. Note that in our implementation and the comparative study, we adopted the methods from the standard libraries : Scikit-survival [106], Lifelines [29] and PySurvival [41]. More details about the methods can be found in Section C2.

3.3 Ensemble Methods

The main objective of ensemble methods is to combine the predictions of multiple estimators in order to improve generalizability and robustness and to obtain more reliable and accurate predictions. One has to derive effective combination rules or design powerful algorithms to boost performance. Ensemble methods consist of both empirical [58] and theoretical [116] approaches. It can be proved that weak learners can be boosted into strong learners through ensemble methods by combining multiple estimators. Applications of ensemble methods [143] can be found

in many fields, such as computer vision, computer security, aided medical diagnosis, credit card fraud detection, weather forecasting, predictive maintenance, etc.

There are three reasons why it is possible to construct very good ensemble methods [31]. First, from a statistical point of view, a learner algorithm can be seen as a procedure to identify the best hypothesis space $\mathcal{H}$. When there is a small amount of data available, the algorithm may find many spaces that fit with the same accuracy. By aggregation, ensemble methods, however, can reduce the risk of choosing the wrong learner. Secondly, ensemble methods have computational advantages because learning algorithms can get stuck in local optimum solutions, and even when there is enough training data, it can still be challenging to find the best hypothesis. This issue can be addressed by running multiple learners from different starting points. Thirdly, in most applications of machine learning, the truth cannot be represented by any of the hypotheses in the $\mathcal{H}$ space. However, by forming a weighted version of the elements of $\mathcal{H}$, it is possible to expand the space of representable functions.

In this paper, we use a gradient descent optimization algorithm to set the parameters of the convex combination of the six methods described in Section 3.2.1. Assuming that we have K procedures to estimate the survival probability function, let us set $\hat{S}_k$ as the estimator proposed by the k-th method. We want to find the parameters $\lambda_k \geqslant 0$ such that

$$\hat{S}(t|X) = \sum_{k=1}^{K} \lambda_k \hat{S}_k(t|X),$$

minimizes the integrated Brier score provided that $\sum \lambda_k = 1$. In order to do this, we optimize the weights λ_k in a subset of the data $\mathcal{D}$ of size n. We consider the gradient vector as the descent direction, which follows the definition of integrated Brier score (see Section C1.2), the j-partial derivative is given by :

$$\frac{\partial IBS(\hat{S}, \mathcal{D})}{\partial \lambda_j} = \frac{1}{\tau n} \sum_{i=1}^{n} \int_0^{\tau} W_i(t) \cdot 2\big(\mathbb{1}\{y_i > t\} - \sum_{k=1}^{K} \lambda_k \hat{S}_k(t|x_i)\big) \cdot (-S_j(t|x_i))\ dt. \tag{3.1}$$

The gradient descent algorithm is presented in Algorithm 2.

Algorithm 2 Exponential Gradient Descent

1: **Require :** T number of iteration, $\eta > 0$ learning rate
2: **Initialization :** $\lambda(0) = (1/K, \ldots, 1/K)$
3: **for** each iteration $t = 1, \ldots, T$ **do**
4: Define $Z_t = \sum_{k=1}^{K} \lambda_k(t) \exp(-\eta D f_k)$,
5: where $D f_k = \frac{\partial IBS(\hat{S}, \mathcal{D})}{\partial \lambda_k}$ defined in (3.1).
6: Update $\lambda_k(t+1) = \frac{\lambda_k(t) \exp(-\eta D f_k)}{Z_t}$ for all $k = 1, \ldots, K$.
7: **end for**

Here, we consider η a constant learning rate with initial λ equitably distributed. The iteration process is repeated until it reaches a maximum number that is set as 10000. We estimate the

optimal aggregation weights each time when we fit the methods in a cross-validation process of five folds. It is important to mention that using a gradient descent algorithm for optimizing the parameters is possible thanks to the fact that the integrated Brier score function is convex, which is not the case for the concordance index.

3.4 Datasets

We study three different datasets, whose general properties are summarized in Tableau 3.1.

Tableau 3.1 – Characteristics of the datasets used in our study.

	Samples	Features	Censored	% Censorship
PBC [125]	276	17	165	59.8 %
GBCSG2 [117]	686	8	387	56.4 %
TLCM [71]	7043	19	5174	73 %

3.4.1 Primary Biliary Cirrhosis (PBC)

Mayo Clinic Primary Biliary Cirrhosis dataset was made available by Therneau and Grambsch [125] and it is for studying the effects of the drug D-penicillamine on the lifetime of patients with PBC. This dataset has 276 samples and 17 covariates such as age, presence of ascites, cholesterol, etc. There are 165 patients who did not die at the end of the study (59.8%) and that corresponds to censored data.

3.4.2 German Breast Cancer Study Group 2 (GBCSG2)

German Breast Cancer Study Group was made available by Schumacher et al. [117] and it is used for studying the effects of hormone treatment on breast cancer recurrence. The dataset has 686 samples and 8 covariates, such as age, hormonal therapy, menopausal status, etc. There are 387 patients who did not get cancer again (56.4%), corresponding to censored data.

3.4.3 Kaggle Telco Churn (TLCM)

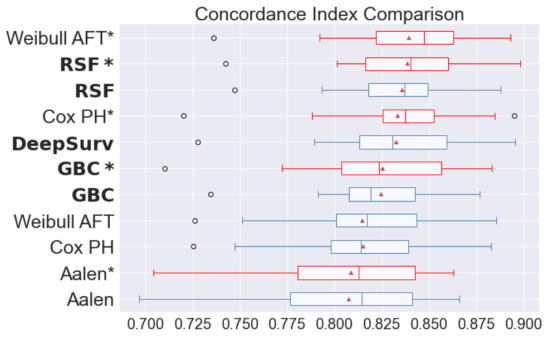
Kaggle Telco Churn dataset was made available in 2008 by Kaggle and it is a sample dataset from IBM [71]. It is used for studying the different causes of customer churn in a fictional telecommunication enterprise. The dataset has 7043 samples and 19 features such as gender, partner, dependents, phone service, etc. This dataset has 5174 clients who have not churned at the end of the study (73%) and that corresponds to censored data.

3.5 Comparison Results

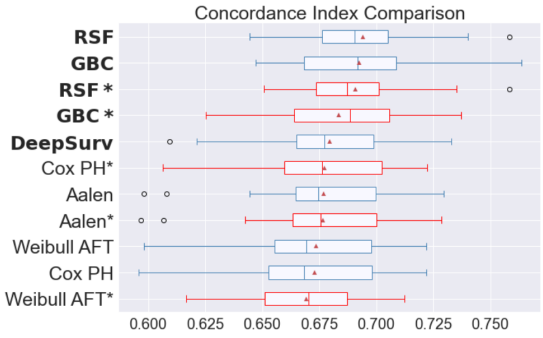
In the following section, we compare the six methods described in Section 3.2.1 through concordance index and integrated Brier score, respectively. Besides, we compare their results

with that of the deployed ensemble method. For each dataset, the scores were computed 25 different times corresponding to 25 partitions (training/validation) of the dataset. This number was chosen arbitrarily in order to maintain a reasonable number of iterations without making the process too computationally expensive. Results are shown by the box plots below. Note that among Figure 3.1a to 3.4, there are some methods with their names marked with an asterisk and their boxes colored by red, which is to indicate the implementation of a randomized search of the parameters conducted by a cross-validation process, whereas the unmarked (and blue) corresponds to adjust the method with the default parameters described in Section C2. In addition, the machine learning techniques were bolded to differentiate them from the semi-parametric and parametric methods.

3.5.1 Concordance index comparison



(a) Box plot comparison across multiple dataset splits using the concordance index on the primary biliary chirrosis dataset.



(b) Box plot comparison across multiple dataset splits using the concordance index on the German breast cancer dataset.

FIGURE 3.1

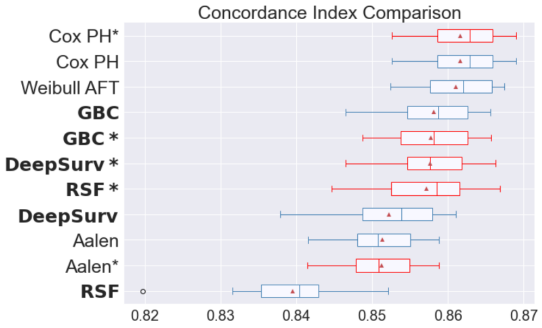


FIGURE 3.2 – Box plot comparison across multiple dataset splits using the concordance index on the telecom churn dataset.

Figure 3.1a shows the concordance index comparison under the PBC dataset. The methods are shown in decreasing order of their obtained mean score. Note that the mean score value is marked

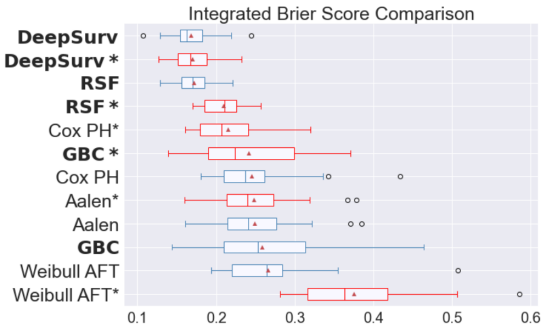
by the red triangle in each box plot. We can observe that Weibull AFT with the randomized search of the parameters (denoted by Weibull AFT*) is the method that outperforms the others, followed by random survival forest with the randomized search of the parameters (RSF*), random survival forest (RSF) and Cox proportional hazard with the randomized search of the parameters (Cox PH*). We can also see that the randomized search of the parameters works well for all the methods (see Weibull AFT* vs. Weibull AFT, RSF* vs. RSF, Cox PH* vs. Cox PH, GBC* vs. GBC, and Aalen* vs. Aalen, respectively). In particular, Weibull AFT* and Cox PH* obtain an increment of 2.9% and 2.5% against Weibull AFT and Cox PH, respectively.

Figure 3.1b shows the concordance index comparison result under the GBCSG2 dataset. Here, the method with the best performance is the random survival forest (RSF), followed by gradient boosting Cox (GBC). Unlike the result under the PBC dataset, we cannot observe an increment in the performance when implementing the randomized search of the parameters on RSF, GBC and the other, except for Cox proportional hazard, implementing the randomized search of the parameters (i.e., Cox PH*) has a slight increment of 0.7%. Figure 3.2 shows the concordance index comparison result under the TLCM dataset. We see that the Cox proportional hazard method (both Cox PH* and Cox PH) outperforms the others, followed by Weibull AFT, whose performance is close to Cox's. In this dataset, we observe that the randomized search does not contribute significantly to improving the performance of the methods, except for the case of random survival forest (RSF) where there is a 2% increment by RSF* when compared with RSF. Weibull AFT* and DeepSurv* were not considered in the graph because they underperformed compared to the other models, and in addition, their performance value was out of the bounds of the figure.

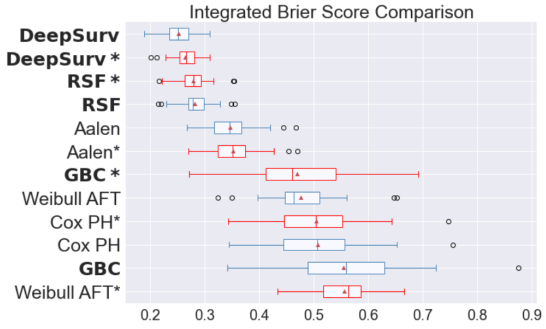
In general, we can observe that for the first two datasets (PBC and GBCSG2), machine learning methods (RSF, RSF*, GBC, GBC* and DeepSurv) perform very well, while parametric methods (Cox PH, Weibull AFT and Aalen additive) are left behind.

This is not the case for the TLCM dataset where Cox HP and Weibull AFT are leading. In addition, we would like to remark the fact that the performance of each method, and its ranking, depends on the dataset. Some methods will perform better for certain types of datasets than others. This may be due to the different characteristics of the datasets, such as the number of covariates, the percentage of censorship, and the total number of observations, together with the assumptions about the hazard function structure and how these assumptions fit the real distribution of each dataset.

3.5.2 Integrated Brier score comparison



(a) Box plot comparison across multiple dataset splits using the integrated Brier score on the primary biliary chirrosis dataset.



(b) Box plot comparison across multiple dataset splits using the integrated Brier score on the German breast cancer dataset.

FIGURE 3.3

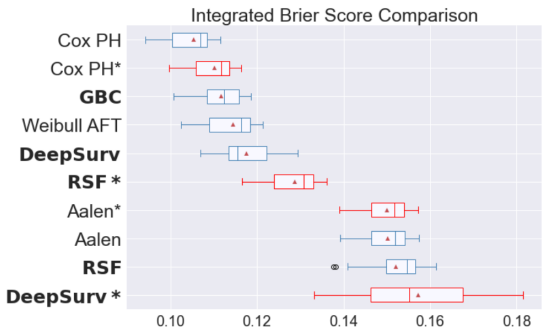


FIGURE 3.4 – Box plot comparison across multiple dataset splits using the integrated Brier score on the Telecom churn dataset.

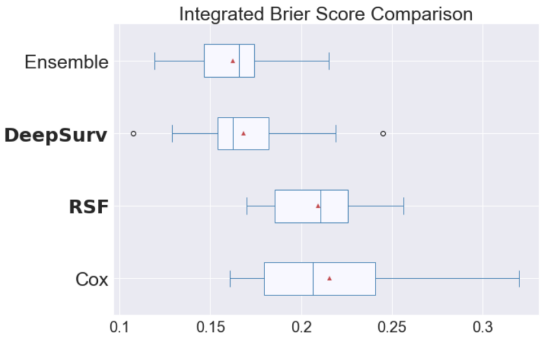
Figure 3.3a shows the integrated Brier score comparison under the PBC dataset. In this figure, as in the case of the concordance index, the methods are displayed in the increasing order of performance, which in this case corresponds to decreasing integrated Brier score. Here, we observe that DeepSurv outperforms the other methods (for the IBS score, the lower the better), followed by DeepSurv* and RSF. We see that there is a clear predominance of machine learning techniques (DeepSurv, DeepSurv*, RSF and RSF*). Similarly, for the GBCSG2 dataset, in Figure 3.3b, DeepSurv outperforms the other methods, followed by DeepSurv*, RSF*, and RSF. Note that Aalen additive has a performance of 23% worse than that of RSF. In this case, we can also say that machine learning techniques (DeepSurv, DeepSurv*, RSF and RSF*) have better results than the other methods.

Figure 3.4 shows the integrated Brier score comparison under the TLCM dataset. Contrary to the previous cases, Cox PH method is the lead. In Figure 3.4, we can appreciate a slight predominance of parametric approaches (Cox PH, Cox PH* and Weibull AFT). We can see that when the amount of censored data is larger, machine learning techniques (DeepSurv and RSF) do not outperform the classical parametric methods.

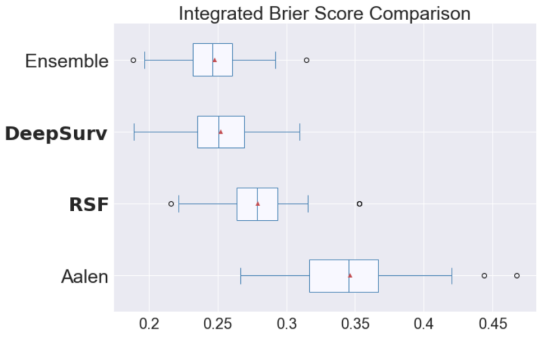
Finally, we would like to remark that for a given dataset the results for the concordance index and integrated Brier score differ. This is not surprising in this case due to the nature of the two scores, that is very different in between them. Some models can give good ranked results while calibrating very poorly and vice-versa. Discussions about how to choose an appropriate score have taken place in the past and there is no consensus in the community [134].

3.5.3 Ensemble methods comparison

In the following, we show the result of our deployed ensemble method. Each aggregation is set according to Section 3.3 for optimizing the parameters of a convex combination of the six methods (described in Section 3.2.1).



(a) Box plot comparison of the ensemble method using the integrated Brier score across multiple dataset splits on the primary biliary cirrhosis dataset.



(b) Box plot comparison of the ensemble method using the integrated Brier score across multiple dataset splits on the German breast cancer dataset.

FIGURE 3.5

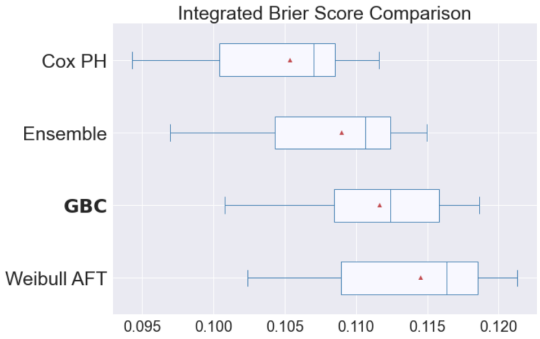


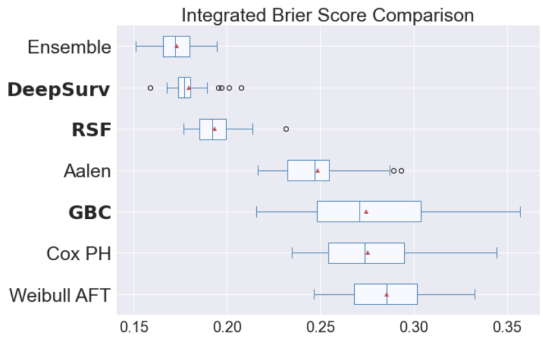
FIGURE 3.6 – Box plot comparison of the ensemble method using the integrated Brier score across multiple dataset splits on the Telecom churn dataset.

Figure 3.5a shows the integrated Brier score comparison result under the PBC dataset. We observe that the ensemble method through gradient descent outperforms DeepSurv by 3%. Similarly, in Figure 3.5b for the GBCSG2 dataset, we find that the ensemble method outperforms the best predictor among the six and obtains a performance improvement of 1.6%. Finally, Figure

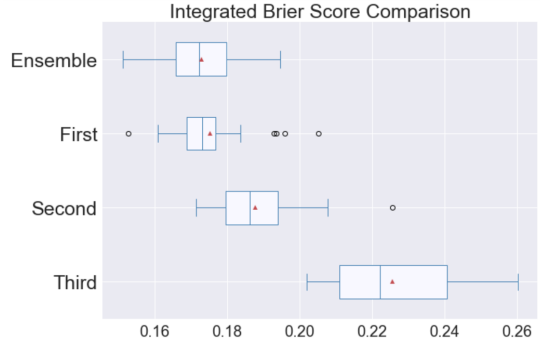
3.6 shows the integrated Brier score comparison result under the TLCM dataset. The ensemble method does not improve the performance, whereas the best estimator is the Cox PH which has a performance of 3.8% better than that of the ensemble method.

In addition, we show the overall performance of each method by averaging the scores obtained by each under the three datasets so as to estimate their overall performance.

Figure 3.7a shows the comparison among all the techniques, including the deployed ensemble method. We see that, in the overall score, the ensemble method outperforms the best predictor by 3.4%. In Figure 3.7b, we show the result obtained by averaging the best scores obtained by the six methods (described in Section 3.2.1) in each of the three datasets (they are DeepSurv for PBC and GBCSG2 and Cox PH for TLCM, see Figures 3.3a, 3.3b and 3.4, respectively) to obtain a global score, which corresponds to the average of the best scores among the six algorithms without using the ensemble method. We similarly average the second best scores obtained by the six methods among the three datasets. Finally, this is also applied to the third best scores in the same way. The results are labeled as "First", "Second" and "Third" in Figure 3.7b, respectively. We see that the ensemble method improves by 1% the performance of the "First" score and has shown its effectiveness.



(a) Box plot comparison across multiple dataset splits using the integrated Brier score comparison among the three datasets.



(b) Box plot comparison across multiple dataset splits using the integrated Brier score overall comparison among the three datasets.

FIGURE 3.7

3.6 Simulation Experiments

To deepen the insights from Section 3.5, we conducted experimental simulations with the goal of comparing the ranking of the methods under different dataset configurations; thus, to understand why some methods perform better than others. The first two methods were based on R libraries, coxed [59] and simsurv [17], which we chose due to their user-friendly functionality. Specifically, we appreciate coxed for its ability to easily specify the percentage of censorship, and we value simsurv for its capability to generate data from a variety of parametric survival distributions not limited to the Cox model. In both simulation cases we assume a particular shape of the hazard function, Cox proportional hazards and Weibull AFT respectively. The third method was developed by us following the logic of the truck dataset from O. Grisel and V. Maladiere [54], with the objective of complexifying the distribution from which we sample the data. We give a further explanation in Section 3.6.1. This method was carried out in Python. Let

us note that we chose three different distributions to sample data, each yielding a distinct dataset. The presented results are the averages obtained from 100 simulations. We do the comparison using the concordance index, and a similar analysis is presented in Section 3.6.5 using the integrated brier score.

3.6.1 Python dataset simulation

Following the truck dataset simulation from [54], we first generate a specified number of features d for each of the N individuals. These features include normally distributed $\mathcal{N}(1, 0.3)$ values, uniform $\mathcal{U}(0, 1)$ values and categorical features of 3 categories. Next, we define three types of failure, as mentioned in [54] : initial assembly failure, operation failure and fatigue failure. Although our method aims to be more general than the truck problem, we maintain the distributions specified in the cited reference. Each type of failure is modeled by a different Weibull curve with parameter λ. The first type of failure has a decreasing hazard with $\lambda = 0.003$, while the other two types have hazard rates that increase, with $\lambda = 3$ and $\lambda = 6$, respectively. The influence of the features on each of the failure types will vary in each experiment, depending on the number of features considered. To continue, we sample events of the three types for each individual and we choose the first one that occurs, or none if no event has taken place (censored case). Finally, we incorporate non-informative uniform censoring, where the parameters of the uniform distribution vary for each simulation case. The length of the uniform interval is what provides us with control over the percentage of censorship [1].

3.6.2 Number of samples

In this section, we compare the behavior of the methods as the number of samples increases. We consider 12 features and 50% of censorship. We vary the number of samples over a grid in between 50 and 2000 to study the impact of the number of samples in the performance of the different methods. The results are presented in the following figures.

(a) Concordance index comparison of the increasing sample size simulation with Coxed library.

(b) Concordance index comparison of the increasing sample size simulation with Simsurv library.

FIGURE 3.8

1. The codes containing all the dataset simulation can be found in the repository : `https://github.com/camferna/Ensemble-Methods-and-Time-to-Event-Analysis-Models`

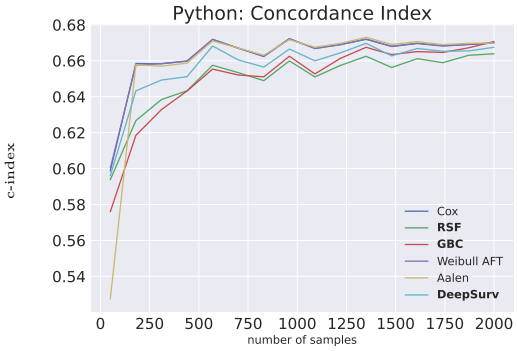
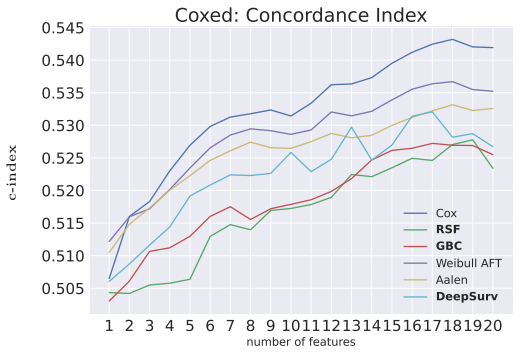


FIGURE 3.9 – Concordance index comparison of the increasing sample size simulation with Python.

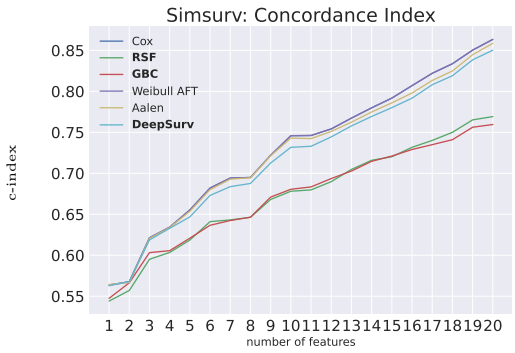
We observe in Figures 3.8a, 3.8b, and 3.9 that the concordance index improves as the number of samples increases. Additionally, we observe in Figure 3.8a that Cox proportional hazard consistently outperforms the other methods, regardless of the number of samples. Subsequently, the order is not very clear, but random survival forest and gradient boosting consistently show lower performance. In Figures 3.8b and 3.9, we observe a consistent outperformance of Cox proportional hazards and Weibull AFT, closely followed by Aalen additive hazards. Random survival forest and gradient boosting underperform compared to the other methods in both figures. In conclusion, the ranking of the models performance appears to depend on the shape of the underlying distribution used to sample the event times and not on the number of samples of the dataset.

3.6.3 Number of features

In this section, we compare the behavior of the methods as the number of features decreases. With a fixed 50% of censorship and 1000 samples, we start the analysis with 20 features and progressively remove one feature at each step. The results are presented in the following figures.



(a) Concordance index comparison of the decreasing number of features simulation with Coxed library.



(b) Concordance index comparison of the decreasing number of features simulation with Simsurv library.

FIGURE 3.10

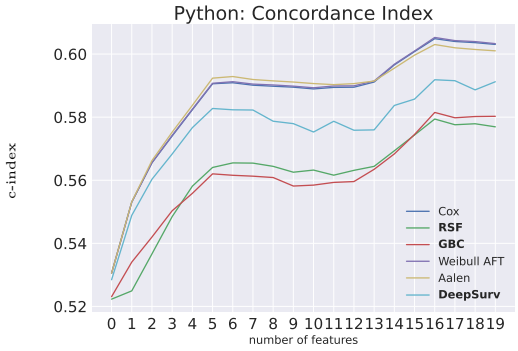


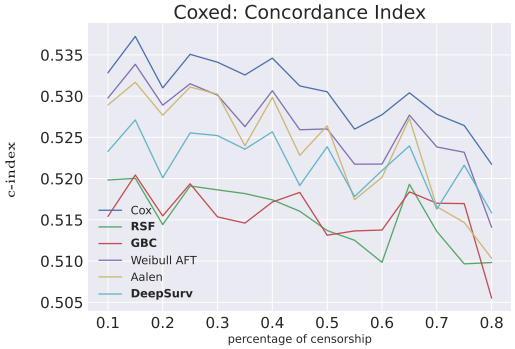
FIGURE 3.11 – Concordance index comparison of the decreasing number of features simulation with Python.

We note in Figures 3.10a, 3.10b, and 3.11 that the concordance index improves as the number of features increases. This behavior align with our expectations since the initial model is constructed with 20 features, and the subsequent removal of features results in a reduction of information. Moreover, we observe in Figure 3.10a that, as in Figure 3.8a, Cox proportional hazard consistently outperforms the other methods, regardless of the number of features. The same holds for Figure 3.10b and 3.11, where the best performance is shared by Cox proportional hazard, Weibull AFT, and Aalen additive hazards. Following the conclusion of Section 3.6.2, the ranking of the models depends mainly on the shape of the distribution used to generate the data, rather than on the number of features.

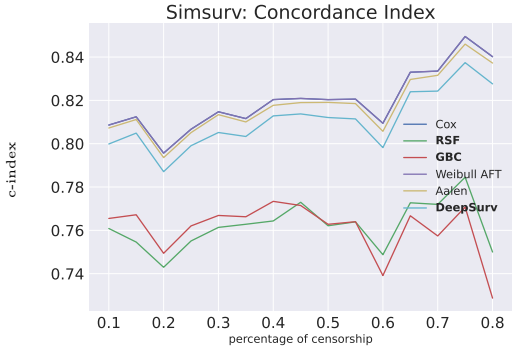
3.6.4 Percentage of censorship

In this section, we compare the behavior of the methods as the percentage of censorship increases. We fix the number of samples at 1000 and the number of features at 12. The results

are presented in the following figures.



(a) Concordance index comparison of the increasing percentage of censorship simulation with Coxed library.



(b) Concordance index comparison of the increasing percentage of censorship simulation with Simsurv library.

FIGURE 3.12

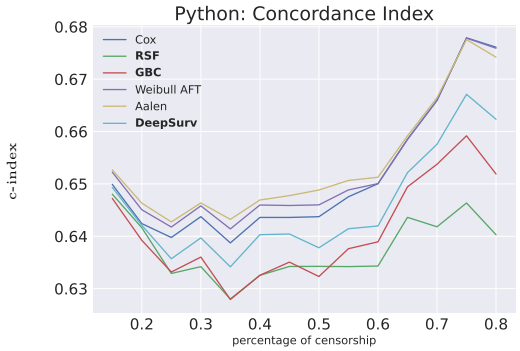


FIGURE 3.13 – Concordance index comparison of the increasing percentage of censorship simulation with Python.

In Figure 3.12a, we observe a decline in performance as the percentage of censorship increases. This is in line with the notion that higher levels of censorship result in reduced available information, consequently leading to diminished performance. However, a contrasting pattern emerges in Figure 3.12b and 3.13, where we actually observe an improvement in performance towards the end of the curves. We believe that this phenomenon is attributed to a bias in the concordance index when the percentage of censorship is high. One solution to address this issue is presented by Uno et al. [130], where they introduced a weighted version of the score. In addition, we observe in Figure 3.12a that Cox proportional hazards outperforms the other methods, followed by Weibull AFT and Aalen additive. This same pattern is evident in Figures 3.12b and 3.13, where these three models lead in terms of performance. Notably, the ranking of the methods remains consistent even as the percentage of censorship increases, reinforcing the conclusion from the previous sections. The primary factor influencing the performance change of the methods is

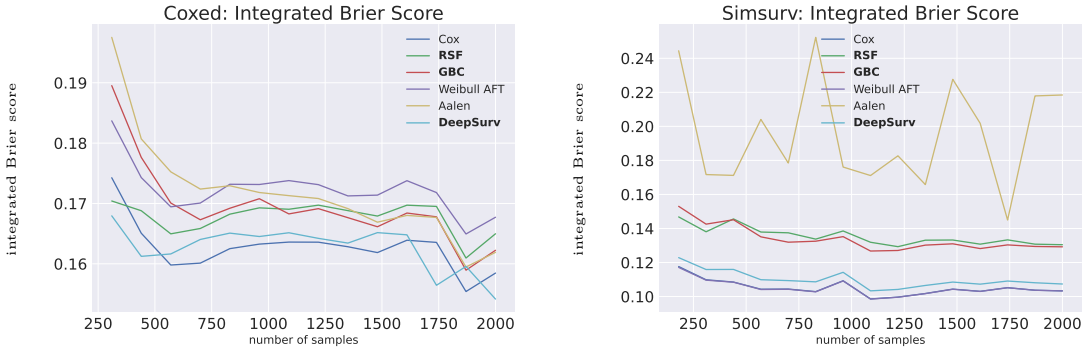
the congruence between the model assumptions and the actual distribution of event times, with improved fit leading to better performance.

3.6.5 Integrated Brier score

In this section, we simulate data using three different techniques. The first set of events is generated by sampling a Cox proportional hazards model, the second by following a Weibull distribution, and the third involves a combination of Weibull distributions. The objective is to compare how the ranking of the methods varies across three experiments. Thus, to understand how different data characteristics can impact the performance of the methods. These findings align with those presented in Section 3.6, with the distinction that we assess performance using the integrated Brier score.

Number of samples

The first experiment consists on evaluating the performance of the models as the number of samples increases from 50 to 2000.



(a) Integrated Brier score comparison of the increasing sample size simulation with Coxed library.

(b) Integrated Brier score comparison of the increasing sample size simulation with Simsurv library.

FIGURE 3.14

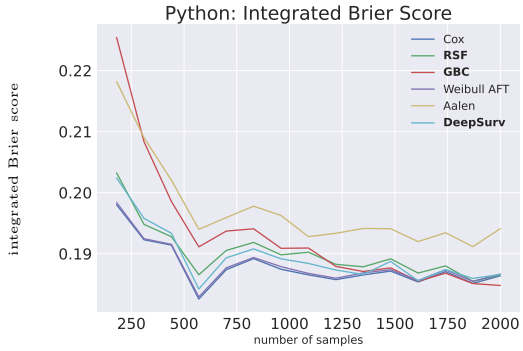
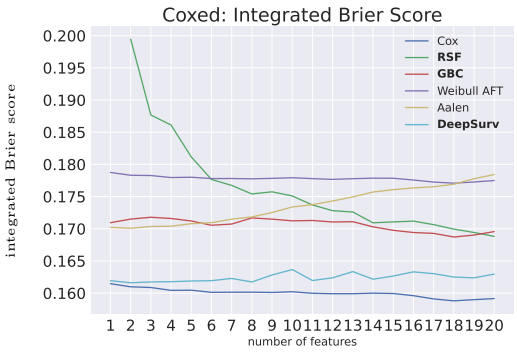


FIGURE 3.15 – Integrated Brier score comparison of the increasing sample size simulation with Python.

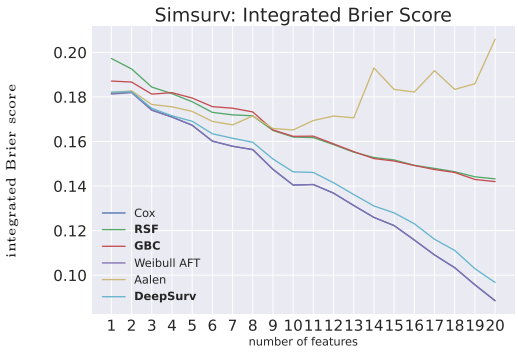
We observe in Figures 3.14a, 3.14b, and 3.15, as discussed in Section 3.6.2, that the performance improves as the number of samples increases. Furthermore, it is noteworthy that the hierarchy of the models remains relatively stable as the number of samples increases. Specifically, Cox proportional hazards outperforms the other methods, with DeepSurv as the second-best performer. This reaffirms the conclusion made in Section 3.6.2 that the models' performance order is independent of the sample size but instead depends on the matching between the underlying assumptions and the dataset real distribution shape.

Number of features

The second experiment consists on evaluating the performance of the models as the number of features decreases from 20 to 1.



(a) Integrated Brier score comparison of the decreasing number of features simulation with Coxed library.



(b) Integrated Brier score comparison of the decreasing number of features simulation with Simsurv library.

FIGURE 3.16

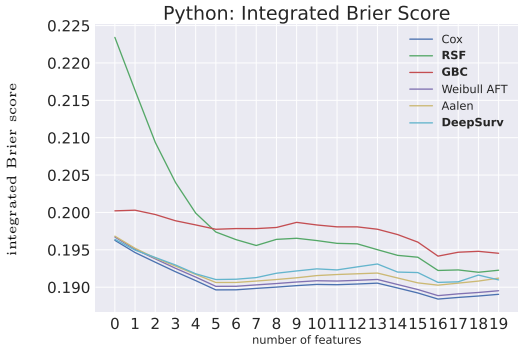


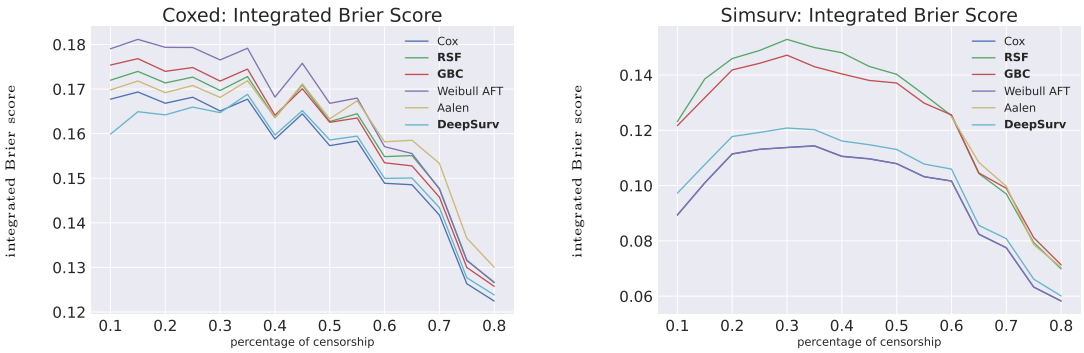
FIGURE 3.17 – Integrated Brier score comparison of the decreasing number of features simulation with Python.

We observe in Figures 3.16a, 3.16b and 3.17 a slight increase in performance as the number of features increases. The most significant change occurs with Random Survival Forest, which exhibits very poor performance compared to the other methods when the number of features is small but becomes competitive as the number of features increases. This is because random survival forest relies on the diversity and richness of features to make accurate predictions. In addition, in Figures 3.16a and 3.16b, we observe that the Aalen additive model does not align with the trend of the other methods, as its performance worsens with an increasing number of features. This could be due to the challenge posed by the additive linearity assumption in capturing the true underlying relationship between covariates and survival outcomes. Finally, we observe that the performance of the Weibull AFT model improves relatively in Figures 3.16b and 3.17 compared to Figure 3.16a. This phenomenon occurs because both simulations, those conducted by the simsurv library and our method implemented in python, are based on the Weibull distribution.

Percentage of censorship

The third experiment, as presented in Section 3.6, consists on increasing the percentage of censorship from 10% to 80%.

We observe in Figures 3.18b and 3.19 an irregular increase in IBS up to 30% and 60% of censorship, respectively. This corresponds to the intuitive expectation that higher censorship rates should lead to poorer performance. However, when the percentage of censorship is high, as can also be seen across the entire curve in Figure 3.18a, we observe an improvement in performance. This phenomenon might be attributed to the distribution of censorship. Censored individuals contribute to the Brier score only until their observed time. Therefore, if their observed time occurs at the beginning of the observation period, their contribution to the score is minimal. Consequently, if there is a significant percentage of censorship, the Brier score risks being small. Additionally, we observe that there is no significant variation in the hierarchy of the models. In Figure 3.18a, Cox PH and DeepSurv consistently maintain the lead throughout the entire experiment, while in Figures 3.18b and 3.19, Cox PH and Weibull AFT remain at the forefront. This corroborates the conclusion drawn in the previous sections, where we found that the main factor determining the ranking of performance is the underlying distribution of the data, rather than the size of the dataset or the percentage of censorship.



(a) Integrated Brier score comparison of the increasing percentage of censorship simulation with Coxed library.

(b) Integrated Brier score comparison of the increasing percentage of censorship simulation with Simsurv library.

FIGURE 3.18

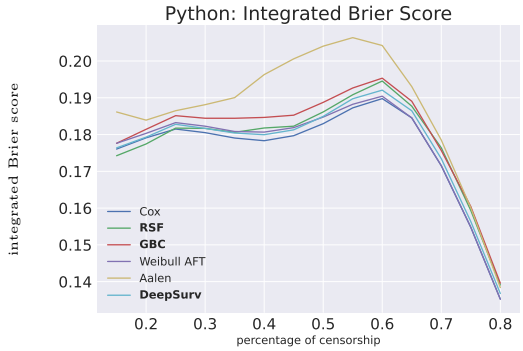


FIGURE 3.19 – Integrated Brier score comparison of the increasing percentage of censorship simulation with Python.

3.7 Conclusions

This paper presents an extensive analysis of different survival methods applied to three datasets and compared by two scoring rules. The study shows how diverse a single method's performance is when changing the measure of comparison and when it is applied to datasets of different distributions, sizes and percentages of censorship. We propose a straightforward aggregation of methods of different natures, parametric, semi-parametric and machine learning, that assume diverse shapes of the hazard function allowing the ensemble model to gain in robustness with respect to each single predictor. This can be observed in Figure 3.7a by the outperformance of the assemblage measured by an overall score that is independent of the dataset. Finally, we present simulation experiments with the objective of studying which dataset characteristics have the most significant influence on the performance of the models. This analysis leads us to the conclusion that the proximity of the model assumptions to the real event distribution is a determining factor in performance. Further research could go in the direction of complexifying the

combination algorithm by considering time-varying weightings and more sophisticated optimization procedures. Another direction could be to find theoretical guarantees for the integrated Brier score of the ensemble method and possibly in a stochastic setting.

Chapitre 4

Predicting Employee Attrition with Survival Analysis

Abstract

Predicting employee attrition presents a significant challenge for companies. By understanding the complex interactions among factors that influence an employee's decision to leave, companies can mitigate future turnover and retain a valuable workforce. We propose using survival analysis techniques to predict employee attrition. This approach is well-suited to address the high level of right censoring, where a significant portion of employees have not left the company by the end of the data collection period, leading classical tabular data algorithms to overestimate or underestimate attrition times. We evaluate the performance of both parametric and machine learning methods, comparing them using various metrics. Furthermore, we investigate the impact of employee characteristics on attrition time distributions through two distinct approaches.

4.1 Introduction

Employee attrition is a big challenge to business continuity and growth. The ability to predict when an employee is likely to leave can offer invaluable insights for human resource strategies, allowing proactive measures to improve retention and maintain organizational stability. Addressing this issue involves understanding the dynamics and causes of employee turnover, assessing the impact attrition has on the organization, and identifying effective strategies to mitigate its effects [101]. Our focus is on the initial step : comprehending the dynamics of employee attrition, including predicting when employees are more likely to leave and identifying the most important factors in this prediction, in order to implement preventative actions promptly.

Many approaches have been developed to study employee turnover. Bennett et al. [9] explored the variables linked to employee dropout using hierarchical multiple regression analysis. Alao D. and Adeyemo A.B. [5] introduced classification decision trees for predicting employee attrition. Further contributions by Ajit P. [4] and Frye B. et al. [45] incorporated machine learning techniques such as PCA, k-NN, Random Forest, and Logistic Regression to refine attrition prediction

models. Most recently, Guerranti F. and Dimitri G.M. [55] integrated an interpretability analysis, offering deeper insights into the predictive models.

Leveraging these varied approaches, survival analysis stands out as a branch of statistics that aims to find the time until a certain critical event occurs, such as employee departure. An important characteristic of survival analysis is that it addresses the problem of censoring. In our work, we specifically focus on the issue of right censoring. Throughout this paper, we will refer to it as **censoring**, however, it should be understood that we are always referring to right censoring.

Censoring occurs when a portion of individuals has not experienced the event by the end of the data collection period. This scenario is particularly relevant in some enterprises, where a large number of employees have not left at the end of the data collection period, highlighting the importance of selecting models that take censoring into account for reliable predictions.

In 1993, Morita, J. et al. [100] pioneered the use of survival analysis methods to investigate employee turnover, setting the stage for numerous subsequent studies in this field. Notably, Frierson, J. and Si, D. [44] proposed to use a Kaplan Meier estimator [76] to evaluate the risk of attrition across different department groups. Furthermore, they used Cox proportional hazard [24] to identify the individual departure probability. Later, Jin, Z. et al. [73] integrated random forest [15] with random survival forest [72] to forecast employee attrition, demonstrating the significant advantages of survival analysis in enhancing predictive performance.

We study the employee attrition by using diverse survival analysis techniques, from traditional models like Cox proportional hazards and Weibull AFT [89] to advanced machine learning methods including Gradient boosting [42], Gradient boosting cumulative incidence function [54], Random Survival Forest, and DeepSurv [77]. We evaluate the performance of these models through specific survival analysis scores, aiming to identify the most effective approach for predicting employee turnover in a specific dataset.

4.2 Preliminaries

4.2.1 Notation

We consider a set of employees $i \in \{1, \dots, N\}$, each one associated to a characteristic vector $x_i \in \mathbb{R}^d$. We define t_i the attrition time of employee i, and c_i its censored time, two non-negative random variables. The time we observe is $u_i = \min\{t_i, c_i\}$, which will correspond to the attrition time t_i if the employee has left the company, and to c_i if the employee is still with the company. Additionally, we define the event indicator $\delta_i = \mathbb{1}\{t_i \leqslant c_i\}$. The objective is to estimate the survival probability function :

$$S(t|x_i) = \mathbb{P}(t_i \geqslant t|x_i) \quad t \geqslant 0\,.$$

This is commonly achieved by assuming a specific shape for the hazard function :

$$H(t|x_i) = -\frac{\partial}{\partial t}\log(S(t|x_i)) \quad t \geqslant 0\,.$$

Further details on the estimation of event times distribution can be found in the work of Cox, D.R. and Oakes, D. [25]

4.2.2 Dataset

The dataset used in this study is confidential, and consequently, we will provide approximate information. The data consists of the order of $N = 10000$ employees, of whom only around 3% had left the company by the end of the study, resulting in a very high percentage of censoring (97%). Each individual has 53 characteristics, including gender, age, country, salary range, among others. These characteristics have been modified from the original data for privacy reasons, and we will refer to them as variable 1, variable 2, etc., to maintain confidentiality. Our goal is to identify the probability of each individual leaving, enabling timely actions to retain valuable employees. The dataset has been anonymised for confidentiality reasons.

4.2.3 Metrics

Concordance index : We first consider the concordance index [60], which indicates how well the model predicts the ordering of event times. To calculate the concordance index we first take every pair in the test set such that the earlier observed time is not censored. Then we consider only pairs (i, j) such that $i < j$ and we also eliminate the pairs for which the times are tied unless at least one of them has an event indicator value of 1. Next, we compute for each pair (i, j) a score $C_{i,j}$ which for $u_i \neq u_j$ is 1 if the subject with earlier time (between i and j) has higher predicted risk (between i and j), is 0.5 if the risks are tied and 0 otherwise. For $u_i = u_j$ and $\delta_i = \delta_j = 1$ we set $C_{i,j} = 1$ if the risks are tied and 0.5 otherwise. If only one of δ_i or δ_j is 1 we set $C_{i,j} = 1$ if the predicted risk is higher for the subject with $\delta = 1$ and 0.5 otherwise. Finally, we compute the concordance index as follows

$$\frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} C_{i,j},$$

where $\mathcal{P}$ represents the set of eligible pairs (i, j).

Concordance index IPCW : The concordance index becomes inaccurate when the percentage of censoring is high, as in our case. Therefore, it is natural to consider the concordance index inverse probability of censoring weighting (IPCW), a weighted version of the concordance index, proposed by Uno, H. et al. [130], which attempts to correct the bias introduced by censoring. This score weights the contributions to the concordance index based on the estimated probability of being uncensored. Let us consider $\hat{S}(t|x_i)$, an estimation of the survival function, the concordance index IPCW is computed as follows :

$$CIPCW = \frac{\sum_{i:\delta_i=1} \sum_{j:t_j>t_i} w_i w_j \mathbb{1}\{\hat{S}(t_i|x_i) < \hat{S}(t_j|x_j)\}}{\sum_{i:\delta_i=1} \sum_{j:t_j>t_i} w_i w_j},$$

where w_i and w_j are the inverse probability of censoring weights for individuals i and j, that we estimate using Kaplan-Meier.

Integrated Brier score (IBS) : Ultimately, recognizing that ranking scores have their limitations, we also consider a calibration score, the integrated Brier score. This represents an integrated version of the Brier score [16]. Let us consider $\hat{S}(t|x_i)$ an estimation of the survival function $S(t|x_i)$, we define $S_C(t|x_i) = \mathbb{P}(c_i \geqslant t|x_i)$ and for a given horizon time $\tau > 0$, the

integrated brier score will be :

$$IBS(\hat{S}) = \frac{1}{N}\sum_{i=1}^{N}\int_0^{\tau} W_i(t)(\mathbb{1}\{u_i > t) - \hat{S}(t|x_i))^2\,,$$

where,

$$W_i(t) = \frac{\delta_i \mathbb{1}\{u_i \leqslant t\}}{\hat{S}_C(u_i|x_i)} + \frac{\mathbb{1}\{u_i > t\}}{\hat{S}_C(t|x_i)}\,.$$

It was proved by Gerds, T.A. and Schumacher, M. [47] that the IBS is a consistent estimator of the mean square error.

We present the metrics precision, recall, and ROC curve in Appendix D1.

4.3 Score Comparison

We estimate the survival curves by splitting the dataset 25 times, allocating 75% of the data for training and 25% for evaluating the performance metrics. As it was mentioned before, taking censoring into account in the evaluation metrics is crucial to accurately reflect the underlying distribution of survival times. To this end, we first consider the concordance index.

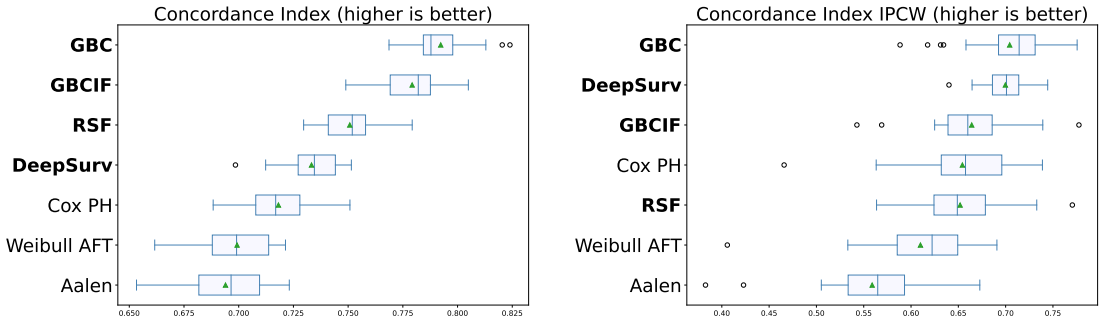


FIGURE 4.1 – Box plot comparison across multiple dataset splits using the concordance index [left] and concordance index IPCW [right] on the attrition dataset.

We observe in Figure 4.1 [left] the boxplot for the concordance index scores across the 25 splits, positioning GBC, GBCIF and RSF as the best models. Moreover, in Figure 4.1 [right] we observe the weighted version of concordance index, designed specifically to avoid the bias of the high censored cases. We notice a general drop of performance comparing to concordance index values, and the increasing on the ranking of DeepSurv.

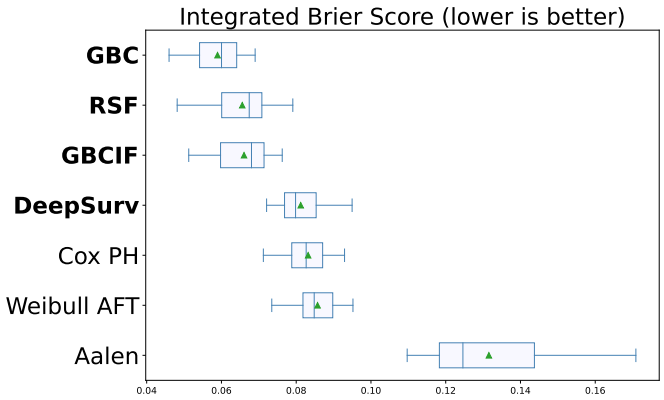


FIGURE 4.2 – Box plot comparison across multiple dataset splits using the integrated Brier score on the attrition dataset.

Figure 4.2 shows the performance using integrated Brier score (IBS) incrementing the sensitivity to probability calibration. However, the outcomes align with those observed in the previous figures. Primarily, there is a notable superiority of machine learning methods (GBC, GBCIF, and RSF), and secondly, the ranking of the methods in relation to the concordance index is maintained.

4.4 Features importance

In this section we aim to identify the features that most influence employee attrition. Understanding which factors contribute most to attrition allows organizations to design targeted interventions. By addressing the root causes of turnover, companies can implement specific policies or programs to improve employee retention and allocate resources more efficiently focusing on areas with the highest impact on employee turnover. Initially, we address this challenge by applying permutation feature importance to identify key features impacting GBC model predictions, our best performer. Subsequently, we select the most crucial features and refine the model by conducting a randomized search to find the optimal hyperparameters. Finally, we assess feature influence on predictions using Shapley values [92].

4.4.1 Permutation feature importance

Permutation importance is a method built to assess the impact of each feature on the performance of a trained model and on a given tabular dataset. It is especially valuable for models that are non-linear or complex, and it consists of randomly shuffling the values of a single feature to observe the effect on the score. This process disrupts the association between the feature and the outcome, revealing the extent to which the model depends on that feature. We conducted this study using the gradient boosting Cox model because it consistently demonstrated superior performance across all evaluated metrics in Section 4.3. Additional information on Cox proportional hazards and random survival forest is available in Appendix D2. The score chosen for this section is the concordance index and we use the permutation feature importance implementation from scikit-learn [104].

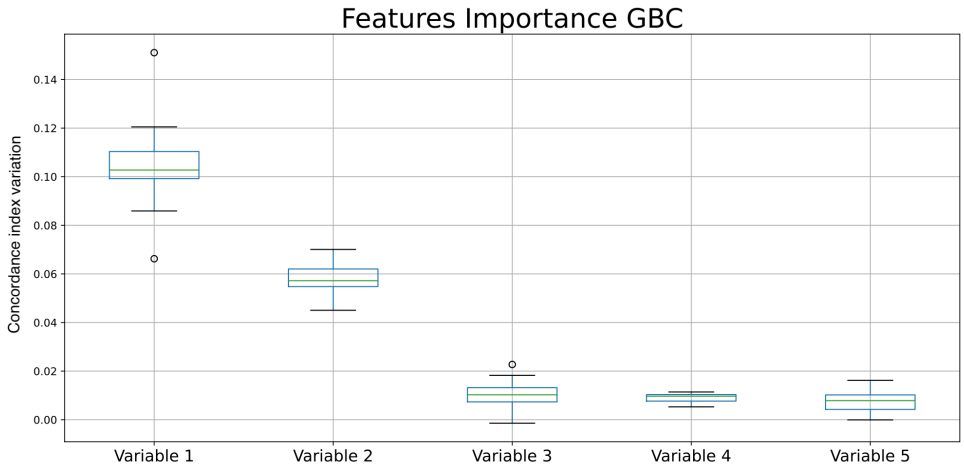


FIGURE 4.3 – Permutation feature importance results of GBC model on the attrition dataset.

Figure 4.3 shows the boxplot of score variations resulting from 15 shufflings. We note that variable 1 is the most significant feature, followed by variable 2 and variable 3. We define the weight of a certain feature as its average concordance index variation and we display the ranked weights and we observe that beyond a certain point, specifically the 14th feature in the weight ranking, the features have null weight. Reducing the number of features used to train the model can significantly decrease model dimensionality and complexity. We examine the variation in the concordance index across three subsets of features to choose the minimal number of features that do not compromise the accuracy of the model.

TABLEAU 4.1 – Concordance index comparison of GBC model when selecting different subsets of features.

	53 features	11 features	7 features	4 features
Concordance index	0.763	0.763	0.761	0.724

In tableau 4.1 we present the concordance index results when training GBC with varying numbers of features. We highlight the significance of feature selection, revealing that retaining just 11 features, merely 21% of the original set, preserves the reliability while significantly reducing the model dimensionality. This identifies the optimal subset of features for efficiently training the GBC model.

4.4.2 Hyperparameters

In this section, we explore the combined impact of hyperparameter optimization and feature selection on the model performance. The hyperparameters we consider include the learning rate, which reduces the contribution of each tree, the maximum depth, limiting the number of nodes in each tree, and the minimum sample leaf, specifying the minimum number of samples required for a node to become a leaf.

We conducted a randomized search, a technique that randomly selects points from a predefined set of hyperparameters for testing through cross-validation. In the end, we chose the

parameters that performed best according to the concordance index.

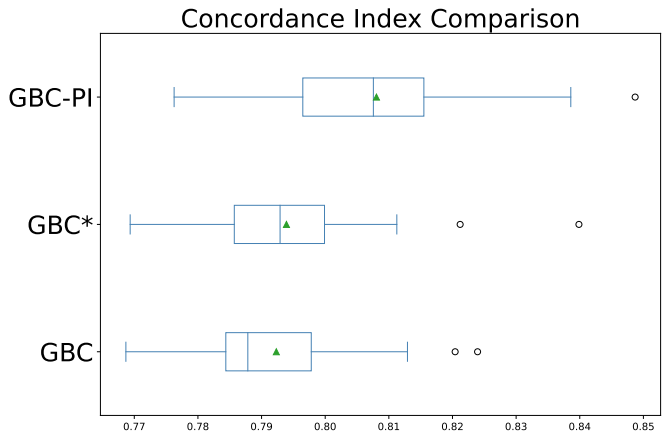


FIGURE 4.4 – Box plot comparison across multiple dataset splits of the concordance index for the GBC model on the attrition dataset. We evaluate the impact of feature selection and hyperparameter optimization.

Figure 4.4 shows the boxplot evaluating the concordance index across 25 dataset splits. GBC indicates the results from Section 4.3, where no hyperparameter tuning was conducted, and the model was trained with all features. GBC* denotes the model fitted after conducting a randomized search for hyperparameters using all 53 features. Lastly, GBC-PI refers to the model for which a randomized search of hyperparameters was performed using only 11 features. By reducing the number of features, the dimensionality of the model decreases, simplifying the optimization of hyperparameters. We observe in Figure 4.4 that conducting a randomized search slightly improves the performance, and combining this with feature selection amplifies the performance increase.

4.4.3 Shapley values

In this section, we study the effect of features on the predictions of the Gradient Boosting Cox model by using Shapley values. Lloyd Shapley introduced Shapley values within cooperative game theory [119], aiming to allocate rewards fairly based on individual contributions to collective success. In the context of model interpretation, this involves assessing the effect of adding or removing a feature on the model prediction across all possible feature combinations or coalitions. the Shapley value of a feature represents its average contribution over these combinations, enhancing model transparency and explainability.

We consider a set of features $\mathcal{X} \subseteq \mathbb{R}^d$ and the output of the model $\nu : 2^{\mathcal{X}} \to \mathbb{R}$, which in this context, represents the predicted risk of the model. The Shapley value of feature i is given by

$$\phi_i(\nu) = \sum_{Y \subseteq \mathcal{X} \setminus i} \frac{|Y|!(|\mathcal{X}| - |Y| - 1)!}{|\mathcal{X}|!} (\nu(Y \cup \{i\}) - \nu(Y)) \,,$$

where Y denotes a subset of features excluding feature i, $\frac{|Y|!(|\mathcal{X}|-|Y|-1)!}{|\mathcal{X}|!}$ represents the number of permutations for which the features in Y come before feature i, and $(\nu(Y \cup \{i\}) - \nu(Y))$ measures

the marginal contribution of i to the subset Y. We use the SHAP library [92] to compute Shapley values.

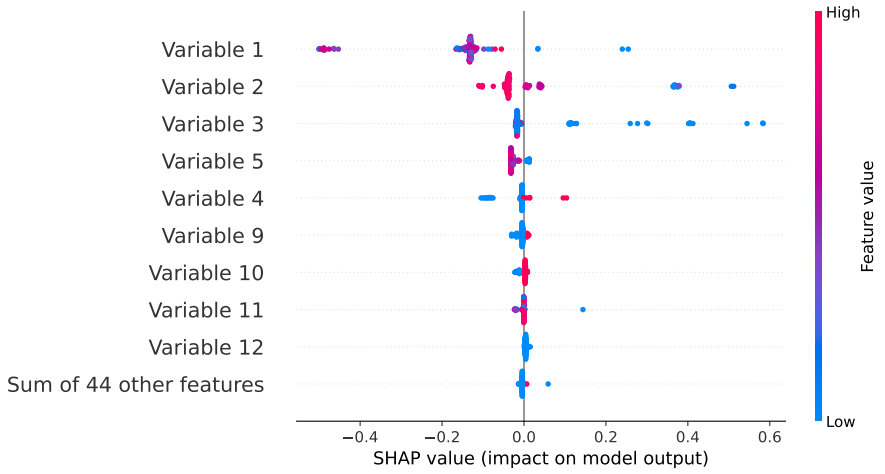


FIGURE 4.5 – Feature importance beeswarm evaluated using Shapley values of GBC model on the attrition dataset.

Figure 4.5 presents a beeswarm plot illustrating the features importance. As observed in Figure 4.3, variable 1 emerges as the most significant feature for GBC predictions, followed by variable 2 and variable 3. Each point on the plot represents a Shapley value for a feature relative to a single prediction, with red points denoting higher feature values and blue points indicating lower values. The horizontal position of each point reflects the impact of the feature on the Shapley value. This indicates that higher values of variable 1 negatively affects the GBC output, implying that individuals with higher values of variable 1 are less likely to leave the company. Similarly, a low value of variable 2 and variable 3 positively affects the GBC predicted risk, meaning that it increases the risk of leaving the company.

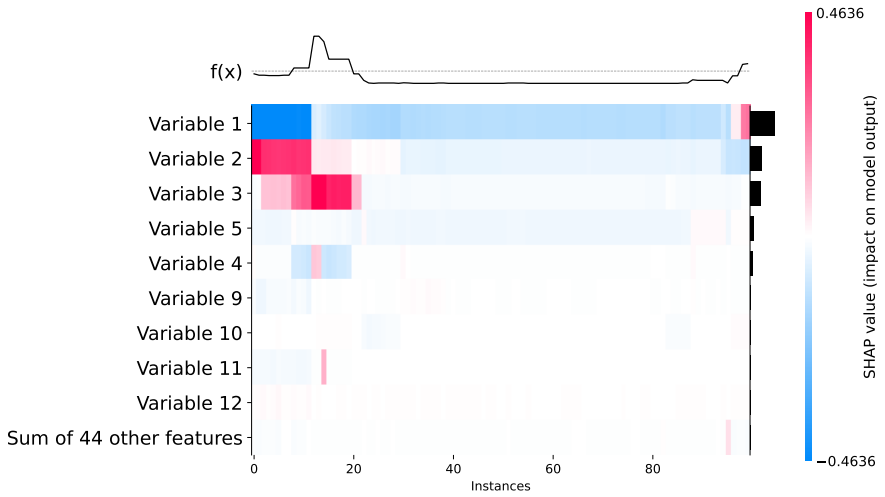


FIGURE 4.6 – Feature importance heatmap evaluated using Shapley values of GBC model on the attrition dataset.

The heatmap 4.6 presents features in rows and instances in columns, with color intensity reflecting the impact of each feature on model predictions. The color scale illustrates the range of impact, allowing identification of key influencing features across the dataset. Figure 4.6 complements the results shown in Figure 4.5, highlighting that lower values of variable 1 have significant negative impact on the model predictions, meaning individuals with lower variable 1 have a lower probability to stay in the company. Conversely, for variable 2 and variable 3, low values have a significant positive impact, suggesting that low values of these variable decrease the probability of staying with the company. This approach allows us to observe how each feature influences the model predictions. We focus on the most significant features, but further analysis of the remaining features remains a valuable endeavor.

4.5 Ensemble methods

In [37], we propose combining multiple methods for survival analysis to enhance robustness and accuracy. In this section, we consider the Cox proportional hazards, the Aalen additive hazards model, gradient boosting Cox, random survival forest, and Weibull AFT. We propose formulating the ensemble prediction as the convex combination of the predictions from each method. Specifically, for each predictor $j = 1, \ldots, K = 5$, we consider its estimation of the survival probability $\hat{S}_j : \mathbb{R}^+ \to [0, 1]$, and the ensemble estimation will be :

$$\hat{S}(t|x_i) = \sum_{j=1}^{K} \lambda_j \hat{S}_j(t|x_i) \quad \text{such that} \quad \sum_{j=1}^{K} \lambda_j = 1\,.$$

where the weights λ_j in the ensemble prediction are determined by minimizing the integrated Brier score using exponential gradient descent with 5000 iterations.

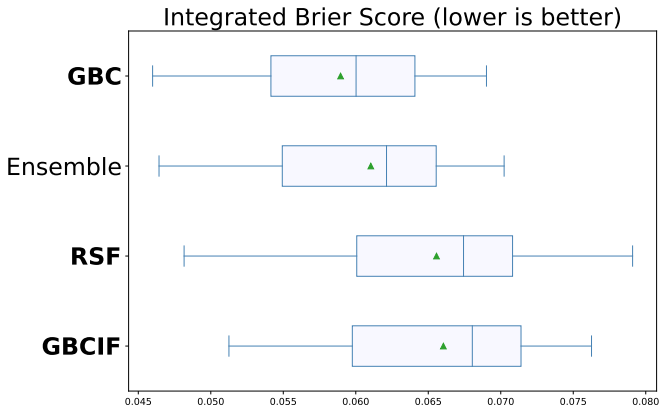


FIGURE 4.7 – Box plot comparison of the ensemble method using the integrated Brier score across multiple dataset splits on the attrition dataset.

Figure 4.7 shows the value of the integrated Brier score of the ensemble method evaluated across 25 dataset splits. We observe that the aggregation does not outperform the best method (GBC), but it remains performant compared to all the other methods, highlighting the advantages of considering multiple predictive models to enhance accuracy and reliability.

4.6 Online learning approach

In [38], we proposed to model the hazard function as an exponential using a parametric approach. Given a parametric family $\Theta \subseteq \mathbb{R}^d$, we assume that there exist a vector $\theta \in \Theta$ such that :

$$H(t|x_i) = \exp(\theta^\top x_i)\mathbb{1}\{t \geqslant \tau_i\} \quad t \geqslant 0\,,$$

where τ_i denotes the arrival time of individual i. The parameters are determined by optimizing the negative log-likelihood of the model, specifically through online convex optimization algorithms. This online setting allows for real-time model updates, accommodates large datasets by processing data in batches, and adapts quickly to new information, possibly enhancing predictive accuracy.

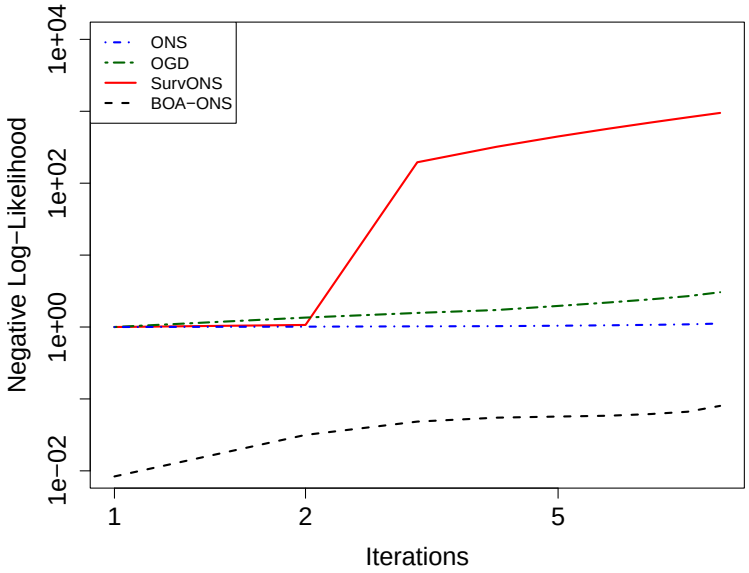


FIGURE 4.8 – Cumulative negative log-likelihood comparison of online methods on the attrition dataset.

Regrettably, the data collection occurred monthly, and we have only nine monthly observations, which limits the scope of online convex optimization. These methods require a larger number of iterations to ensure accurate parameter estimation. In Figure 4.8, we compare the negative log-likelihood estimation of Online Newton Step [65] (ONS), Online Gradient Descent [145] (OGD), Online Bernstein Aggregation [138] (BOA-ONS), and Survival ONS [38] (SurvONS). Figure 4.8 illustrates the absolute value of the cumulative negative log-likelihood. We observe that BOA-ONS minimizes the negative log-likelihood more effectively than the other models, thus suggesting better performance. Conversely, the negative log-likelihood associated with our method, SurvONS, significantly increases. This result indicates that our model is not optimal in the current context, and it reveals that the online setting does not offer advantages, at least with the present configuration. Further investigation is required to understand the possible benefits of an online setting and to compare this approach with the methods implemented in Section 4.3, including a comprehensive evaluation of performance through multiple data splits and the implementation of a survival score metric.

4.7 Conclusions

We observe consistent performance across the different scores, without noting any significant differences between classical and survival scores. The ranking of the models is mostly maintained, reinforcing GBC as the most effective method for estimating employee attrition in this dataset. Additionally, we evaluated the importance of features using two methods : permutation feature importance and Shapley values, both revealing variable 1 as the most significant factor influencing GBC predictions. Shapley values allow us to reinforce the study of feature importance

by explicitly showing how each feature affects the model prediction. We also conducted feature selection to choose a subset of features for training the method, thereby simplifying the optimization of hyperparameters and enhancing the method performance. Finally, we briefly present the application of our results from [37] and [38] to the attrition dataset.

Conclusions

The central theme of this thesis was to explore different aspects of survival analysis. In a first approach, we provided a detailed mathematical framework that enables the adaptation of online convex optimization methods to censored data. We proposed a sequential model for estimating the survival function using online convex optimization tools, which have not previously been explored. In this context, we introduced a stochastic setting that facilitates regret analysis, ensuring logarithmic bounds for the Online Newton Step. Additionally, we developed a new algorithm that adaptively selects the learning rate to compensate for potential increases in regret. We demonstrated that the regret of this algorithm is bounded and discussed grid selection using simulation experiments. We concluded that aggregation methods enhance robustness in hyperparameter selection, but achieving and maintaining fast rates remain a non-trivial task. Secondly, we conducted an extensive analysis of various survival models, examining how method performance varies with the scoring rule and dataset. We proposed a straightforward aggregation method that enhances robustness compared to individual methods, achieving superior performance across an overall score that is independent of the dataset. Moreover, we conducted simulation experiments aimed at understanding which data characteristics most significantly influence method performance. We concluded that the alignment between the model assumptions about the hazard function and the actual data distribution plays the most fundamental role. Finally, we applied the multiple survival analysis approaches to a real industrial case.

In conclusion, we observed that online methods require numerous iterations to accurately estimate parameters, which is not always feasible with real data. Additionally, the parametric approach can be challenging to fit due to issues with the convexity of the loss functions, which, at certain iterations, might be too flat due to the absence of individuals at risk. On the other hand, machine learning models provide a complex non-parametric approach that better accommodates the varied shapes of datasets, although their results are harder to interpret. Estimating survival curves is complex, necessitating consideration of both the temporal aspects of data and censorship. Finally, we observed that selecting an appropriate scoring rule is not straightforward.

There are several ways to expand our work on online learning algorithms : we can study and compare our approach with the state-space models of Fahrmeir [34] ; and we can examine the use of continuous ranked probability scores [131] in the context of online survival data. Additionally, we can explore the use of hybrid methods that perform many batch iterations during the first part of the observed period, followed by online iterations. This approach is designed to address the issue of not having enough observations to implement an online procedure effectively. Concerning the ensemble method of Chapter 3, we can extend the analysis by complexifying the aggregation procedure, considering online weights and diverse optimization methods. In a more general framework, we can explore the adaptation of some machine learning methods for tabular data to censored data ([108],[6]), and the combination of online learning with deep learning techniques.

Annexe A

Appendix A

This appendix is dedicated to presenting the code, in Python and R, for the different sections of Chapter 1.

A1 Illustrating covariate effects with clinical data

The objective is to use the PBC dataset to illustrate the covariate effects in the estimation of survival curves. We fit Kaplan-Meier and Cox proportional hazards and we compare the covariate effects in both models. The code of this section was developed with Python.

```
1
2  # import libraries
3  import statsmodels.api as sm
4  import numpy as np
5  import pandas as pd
6  import matplotlib.pyplot as plt
7
8  # import the dataset with no nan values
9  df_raw = sm.datasets.get_rdataset('pbc', 'survival').data
10 df_no_nans = df_raw.dropna()
11 df_no_nans = df_no_nans.drop('id', axis=1)
12 df_trans = df_no_nans['status']
13 df_no_nans['status']=df_no_nans['status'].replace(1,False)
14 df_no_nans['status']=df_no_nans['status'].replace(2,True)
15 df_no_nans['status']=df_no_nans['status'].replace(0,False)
16
17 # identify covariates X and target y
18 from sksurv.datasets import get_x_y
19 X, y = get_x_y(df_no_nans, attr_labels=['status','time'], pos_label=True)
20
21 # plot the Kaplan-Meier estimation
22 from lifelines import KaplanMeierFitter
23
24 kmf = KaplanMeierFitter()
```

```
kmf.fit(y['time'], y['status'])
kmf.plot()
plt.title('Kaplan-Meier Estimator of the Survival Function', fontsize =
    15)
plt.xlabel('time', fontsize = 12)
plt.ylabel('probability', fontsize = 12)
plt.show()

# identify the treatment groups and plot the KM curves
idx1 = np.where(X['trt']==1.0)[0]
idx2 = np.where(X['trt']==2.0)[0]

kmf_trt1 = KaplanMeierFitter()
kmf_trt2 = KaplanMeierFitter()
kmf_trt1.fit(y['time'][idx1], y['status'][idx1])
kmf_trt2.fit(y['time'][idx2], y['status'][idx2])
kmf_trt1.plot(ci_show = False)
kmf_trt2.plot(ci_show= False)

plt.title('Kaplan-Meier Estimator of the Survival Function', fontsize =
    15)
plt.legend(('treatment 1','treatment 2'))
plt.xlabel('time', fontsize = 12)
plt.ylabel('probability', fontsize = 12)
plt.show()

# identify the age groups and plot the KM curves
idx1 = np.where(X['age']>=49.0)[0]
idx2 = np.where(X['age']<49.0)[0]

kmf_trt1 = KaplanMeierFitter()
kmf_trt2 = KaplanMeierFitter()
kmf_trt1.fit(y['time'][idx1], y['status'][idx1])
kmf_trt2.fit(y['time'][idx2], y['status'][idx2])
kmf_trt1.plot(ci_show = False)
kmf_trt2.plot(ci_show= False)

plt.title('Kaplan-Meier Estimator of the Survival Function', fontsize =
    15)
plt.legend(('age>49','age<49'))
plt.xlabel('time', fontsize = 12)
plt.ylabel('probability', fontsize = 12)
plt.show()

# preprocess the data for the cox model
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OrdinalEncoder
from sklearn.preprocessing import StandardScaler

scaling_cols=['age','bili','chol','albumin','copper','alk.phos','ast','
    trig','platelet','protime']
cat_cols=['trt','edema','sex','ascites','hepato','spiders','stage']

```

```
preprocessor = ColumnTransformer(
    [('cat-preprocessor', OrdinalEncoder(), cat_cols),
    ('standard-scaler', StandardScaler(), scaling_cols)],
    remainder='passthrough', sparse_threshold=0)

# fit the cox PH model
from sksurv.linear_model import CoxPHSurvivalAnalysis

Xprep = preprocessor.fit_transform(X)
cox = CoxPHSurvivalAnalysis()
cox.fit(Xprep,y)
survfunc = cox.predict_survival_function(Xprep)

# baseline hazard
baseline_time = cox.baseline_survival_.x
baseline_probability = cox.baseline_survival_.y

# plot cox PH curves for random different individuals
import random

times = survfunc[0].x
ind = random.sample(range(276),5)
for i in range(5):
    plt.plot(times, survfunc[ind[i]].y)

plt.plot(baseline_time, baseline_probability, linewidth=2.5, color = '
    brown')
plt.legend(('id 1','id 2','id 3','id 4','id 5','baseline'))
plt.xlabel('time',fontsize =12)
plt.ylabel('probability',fontsize = 12)
plt.title('Cox PH Estimator of the Survival Function', fontsize = 15)
plt.show()

# plot cox PH according to the age group
from matplotlib.lines import Line2D

idx1 = np.where(X['age']>=49.0)[0]
idx2 = np.where(X['age']<49.0)[0]

custom_lines = [Line2D([0], [0], color='tab:blue', lw=2),
                Line2D([0], [0], color='tab:orange', lw=2),
                Line2D([0], [0], color='brown', lw=2)]

for i in range(5):
    plt.plot(times, survfunc[idx1[i]].y, color = 'tab:blue')
    plt.plot(times, survfunc[idx2[i]].y, color = 'tab:orange')

plt.plot(baseline_time, baseline_probability, color = 'brown',linewidth
    =2.5 )
plt.legend(custom_lines , ('age>49','age<49','baseline'))
plt.xlabel('time',fontsize =12)
plt.ylabel('probability',fontsize = 12)
plt.title('Cox PH Estimator of the Survival Function', fontsize = 15)
```

```
plt.show()
```

A2 Scoring rules

The objective is to illustrate the effect of censorship in the accuracy estimation of the Cox proportional hazards model. This section was coded in Python. We first define all the required functions.

```

from lifelines import KaplanMeierFitter
from scipy.integrate import trapz

# Kaplan Meier for Brier score
def Kaplan_Meier(y_trn, times):
    kmf = KaplanMeierFitter()
    kmf.fit(y_trn['time'], ~y_trn['status'])
    return 1-kmf.cumulative_density_at_times(times).values

# Brier score
def brier_score (y_trn, y_val, survfunc, times, eps):
    km_survfunc = Kaplan_Meier(y_trn, times)
    y_time = y_val['time']
    y_status = y_val['status']
    brier_vector = []
    N_val = len(y_time)
    for t in range(len(times)):
        km_survfunc[t] = np.maximum(km_survfunc[t], float(eps))
        mean = 0
        for i in range(N_val):
            idx = next((t for t, time in enumerate(times) if time > y_time[i]), -1)
            idx = idx-1 if idx>0 else idx
            km_survfunc[idx] = np.maximum(km_survfunc[idx], float(eps))
            mean = mean + (((y_time[i]<= times[t])*y_status[i]*(0 - survfunc[i][t])**2)/km_survfunc[idx]\
                           + ((y_time[i]>times[t])*(1-survfunc[i][t])**2)/km_survfunc[t])
        brier_vector.append(mean/N_val)
    return brier_vector

# Brier score with no censorship weights
def brier_score_no (y_val,survfunc,times):
    y_time = y_val['time']
    brier_vector = []
    N_val = len(y_time)
    for t in range(len(times)):
        mean = 0
        for i in range(N_val):
            idx = next((t for t, time in enumerate(times) if time > y_time[i]), -1)
            idx = idx-1 if idx>0 else idx
```

```
            mean = mean + (((y_time[i]<= times[t])*(0 - survfunc[i][t])**2)\
                           + ((y_time[i]>times[t])*(1-survfunc[i][t])**2))
        brier_vector.append(mean/N_val)
    return brier_vector

# integrated Brier score
def integrated_brier_score (y_trn, y_val, survfunc, times, esp, no=False):
    if no:
        brier_vector = brier_score_no(y_val, survfunc, times)
        integrated = trapz(brier_vector, times)
    else:
        brier_vector = brier_score(y_trn, y_val, survfunc, times, esp)
        integrated = trapz(brier_vector, times)
    return integrated/(times[-1]-times[0])

# parameters of the data simulation
N = 10
d = 3
mean = [1,-2]
cov = [[1,0],[0,3]]
beta_real = np.array([[-0.5, -0.8, 0.3]]).reshape(d, 1)

# function to create the data
from scipy.stats import multivariate_normal, expon

def create_dataset(N, seed):
    rng = np.random.default_rng(seed)
    Z = rng.multivariate_normal(mean=mean, cov=cov, size=N)
    Z = np.c_[Z, np.ones(Z.shape[0])]
    beta_real = np.array([[-0.5, -0.8, 0.3]]).reshape(d, 1)

    T = [float(expon(scale=1/np.exp(np.dot(beta_real.T, Z[i, :])))).rvs(random_state= rng)) for i in range(N)]
    C = rng.uniform(low=0, high=0.35, size=N)
    hat_T = np.array([float(min(T[i],C[i])) for i in range(N)])
    delta = T<C

    pd_Z = pd.DataFrame(Z)

    data = pd.DataFrame({'feature 1': pd_Z[0], 'feature 2': pd_Z[1], 'feature 3': pd_Z[2],'status': delta, 'time': hat_T, 'real_time': T})

    #print(f'percentage of censorship: {(N-sum(delta))*100/N}')

    return data
```

To continue, we define a function that, given a certain seed, creates a dataset, trains a Cox PH model, and provides the values of the concordance index and integrated Brier score, both considering and not considering censorship.

```

```

```
from sksurv.datasets import get_x_y
from sksurv.linear_model import CoxPHSurvivalAnalysis
from sksurv.metrics import concordance_index_censored

# concordance index and integrated Brier score computation
def multiple_seeds(seed):
    data_trn = create_dataset(1500, seed)
    data_val = create_dataset(500, int(100-seed))

    X_trn, y_trn = get_x_y(data_trn.drop('real_time', axis = 1),
        attr_labels=['status','time'], pos_label=True)
    X_val, y_val = get_x_y(data_val.drop('real_time', axis = 1),
        attr_labels=['status','time'], pos_label=True)

    data_val = data_val.drop('time',axis = 1)
    data_val = data_val.rename(columns = {'real_time':'time'})
    X_true, y_true = get_x_y(data_val, attr_labels=['status','time'],
        pos_label=True)
    y_true['status'] = [True for i in range(len(y_true))]

    cox = CoxPHSurvivalAnalysis(alpha=0.001)
    cox.fit(X_trn,y_trn)

    ci_val = concordance_index_censored(y_val['status'],y_val['time'],
        cox.predict(X_val))[0]
    ci_true = concordance_index_censored(y_true['status'],y_true['time'],
        cox.predict(X_val))[0]

    survfunc_val = cox.predict_survival_function(X_val)
    times_val = survfunc_val[0].x
    cox_preds_val = np.asarray([[fn(t) for t in times_val]
        for fn in survfunc_val])

    ibs_val = integrated_brier_score(y_trn,y_val,cox_preds_val,times_val,
        0.0001)
    ibs_true = integrated_brier_score(y_trn,y_true,cox_preds_val,
        times_val,0.0001, no = True)

    return ci_val, ci_true, ibs_val, ibs_true
```

We apply this procedure with multiple seeds in parallel and we get the results.

```
# parallel seeds
from joblib import Parallel, delayed
x = Parallel(n_jobs=25)(delayed(multiple_seeds)(seed)
    for seed in range(50))

# get the results
ci_val = [x[i][0] for i in range(50)]
ci_true = [x[i][1] for i in range(50)]
ibs_val = [x[i][2] for i in range(50)]
ibs_true = [x[i][3] for i in range(50)]

# plot concordance index
```

```
plt.plot(ci_val)
plt.plot(ci_true)
plt.title('Concordance Index', fontsize = 15)
plt.legend(('censored','non-censored'))
plt.xlabel('seeds', fontsize =10)
plt.show()

# plot integrated Brier score
plt.plot(ibs_val)
plt.plot(ibs_true)
plt.title('Integrated Brier Score', fontsize = 15)
plt.legend(('censored','non-censored'))
plt.xlabel('seeds', fontsize =10)
plt.show()
```

Now, it is remaining to compute the negative log-likelihood.

```
# partial likelihood function
def partial_likelihood(beta,Z,delta,T):
    risk_scores = np.dot(Z, beta)
    log_partial_likelihood = 0
    for i, (ti, di) in enumerate(zip(T, delta)):
        if di:  # Event occurred
            risk_set = (T >= ti)
            log_risk_set_sum = np.log(np.sum(np.exp(risk_scores[risk_set
    ])))
            log_partial_likelihood += risk_scores[i] - log_risk_set_sum

    return -log_partial_likelihood

# compute the partial negative log-likelihood for a given seed
def multiple_seeds(seed):
    data_trn = create_dataset(1500, seed)
    data_val = create_dataset(500, int(100-seed))

    X_trn, y_trn = get_x_y(data_trn.drop('real_time', axis = 1),
    attr_labels=['status','time'], pos_label=True)
    X_val, y_val = get_x_y(data_val.drop('real_time', axis = 1),
    attr_labels=['status','time'], pos_label=True)

    hat_T = data_val['time']
    T = data_val['real_time']
    delta = data_val['status']
    Z = np.transpose([data_val['feature 1'],data_val['feature 2'],
    data_val['feature 3']])

    delta_true = [True for i in range(len(delta))]

    cox = CoxPHSurvivalAnalysis(alpha=0.001)
    cox.fit(X_trn,y_trn)

    beta = cox.coef_

    like_val = partial_likelihood(beta,Z,delta,hat_T)
```

```
    like_true = partial_likelihood(beta,Z,delta_true,T)


    return like_val, like_true

# run many seeds in parallel
from joblib import Parallel, delayed
x = Parallel(n_jobs=50)(delayed(multiple_seeds)(seed) for seed in range
    (50))

# obtain the results
like_val = [x[i][0] for i in range(50)]
like_true = [x[i][1] for i in range(50)]

# plot the likelihood
plt.plot(like_val)
plt.plot(like_true)
plt.title('Partial Negative Log-Likelihood', fontsize = 15)
plt.legend(('censored','non-censored'))
plt.xlabel('seeds', fontsize =10)
plt.show()
```

A3 Online convex optimization

The objective is to illustrate the influence of the learning rate selection in the cumulative loss of Online Newton Step (ONS) algorithm. This section was coded in R.

```
# algorithm parameters
D <- 1.13
d <- 3
N <- 2*10^3
n_it <- 500
beta_real <- matrix(c(-0.5, -0.8, 0.3), d, 1)

gamma1 <- 10^{-3/2}
epsilon1 <- 1/(gamma1*D)^2

gamma2<- 10
epsilon2 <- 1/(gamma2*D)^2

# Monte Carlo simulations
M = 100

ons_beta1 <- array(0, dim = c(n_it,d,M))
ons_beta2 <- array(0, dim = c(n_it,d,M))
ons_like1 <- array(0, dim = c(n_it,M))
ons_like2 <- array(0, dim = c(n_it,M))
real_like <- array(0, dim = c(n_it,M))
gamma_arr1 <- array(0, dim = c(n_it,M))
gamma_arr2 <- array(0, dim = c(n_it,M))

```

```
for (j in 1:M) {
  print('iteration')
  print(j)
  X1 <- mvtnorm::rmvnorm(N, matrix(c(1, -2), 2, 1), diag(c(1,3)))
  X1 <- cbind(X1,1)
  arrival_time = runif(N, min=0, max = n_it)
  Time_indiv <- arrival_time + sapply(1:N, function(i) rexp(1,rate=exp(
    crossprod(beta_real, X1[i,])[1])))
  Censor_indiv <- arrival_time + runif(N, min=0, max = .35)
  hat_T <- sapply(1:N, function(i) {min(Time_indiv[i], Censor_indiv[i])})
  delta <- (Time_indiv < Censor_indiv)

  R <- list()
  for (t in 1:n_it){
    R[[t]] <- c(1)
  }
  for (i in 2:N) {
    t1 <- max(1,floor(arrival_time[i])-1)
    t2 <- min(n_it,floor(hat_T[i])+1)
    for (t in t1:t2)
      R[[t]] <- c(R[[t]], i)
  }

  #ONS
  ons_mu1 <- ons(arrival_time,hat_T,delta,X1,D,gamma1,n_it,epsilon1,R)
  ons_mu2 <- ons(arrival_time,hat_T,delta,X1,D,gamma2,n_it,epsilon2,R)

  ons_beta1[,,j] <- ons_mu1$beta_arr
  ons_beta2[,,j] <- ons_mu2$beta_arr
  gamma_arr1[,j] <- ons_mu1$gamma_temp
  gamma_arr2[,j] <- ons_mu2$gamma_temp

  for (t in 1:n_it){
    ons_like1[t,j] <- instgrad(t, arrival_time, hat_T, delta, X1,
    ons_mu1$beta_arr[t,], R[[t]])$lik
    ons_like2[t,j] <- instgrad(t, arrival_time, hat_T, delta, X1,
    ons_mu2$beta_arr[t,], R[[t]])$lik
    real_like[t,j] <- instgrad(t, arrival_time, hat_T, delta, X1,
    beta_real[,1], R[[t]])$lik
  }
}

# average estimations
ons_mean1 <- array(0, dim = c(n_it,d))
ons_mean2 <- array(0, dim = c(n_it,d))

for (idx in 1:d){
  for (t in 1:n_it){
    ons_mean1[t,idx]<- mean(ons_beta1[t,idx,])
    ons_mean2[t,idx] <- mean(ons_beta2[t,idx,])
  }
}

```

```
74 beta_real_arr = t(matrix(rep(as.numeric(beta_real),n_it), nrow = 3))
75
76 # error estimation plot
77 plot(1:n_it,apply((ons_mean1 - beta_real_arr)^2, 1, sum), type ='l', col
       = 'blue', lwd = 2, lty =4, xlab = 'iterations',ylab = 'error', log = '
       xy', ylim = c(0.0025,0.98) )
78 lines(1:n_it,apply((ons_mean2 - beta_real_arr)^2, 1, sum),col = '
       darkgreen',lwd = 2 ,lty = 3)
79 title('Estimation Error')
80 legend('bottomleft', legend=c("ONS 1",'ONS 2'), col=c('blue','darkgreen')
       , lty = c(4,3),lwd = 2, cex=0.8)
81
82 # negative log-likelihood estimation
83 like_ons_mean1 <- matrix(0,n_it,1)
84 like_ons_mean2 <- matrix(0,n_it,1)
85 like_real_mean <- matrix(0,n_it,1)
86
87 for (t in 1:n_it){
88   like_real_mean[t] <- mean(real_like[t,])
89   like_ons_mean1[t]<- mean(ons_like1[t,])
90   like_ons_mean2[t] <- mean(ons_like2[t,])
91 }
92
93 plot(1:n_it, cumsum(abs(like_ons_mean1-like_real_mean)), type = 'l', col
       = 'blue', lwd = 2, lty=4, xlab = 'iterations',ylab = 'likelihood
       difference')
94 lines(1:n_it,cumsum(abs(like_ons_mean2-like_real_mean)), col ='darkgreen'
       ,lwd =2, lty =4)
95 title('Cumulative Negative Log-Likelihood')
96 legend('topleft', legend=c("ONS 1",'ONS 2'), col=c('blue','darkgreen'),
       lty = c(4,3),lwd = 2, cex=0.8)
```

Annexe B

Appendix B

B1 Background on parametric inference

B1.1 Proof of Proposition 1

Démonstration. We define the equivalent of the survival probability for the censored distribution

$$G(t|x_i, \tau_i) = \mathbb{P}(c_i \geqslant t|x_i, \tau_i)\,.$$

Given $\theta \in \Theta$, we write the density of u_i distinguishing two cases :

$$\mathbb{P}(u_i \in [t, t+h), \delta_i = 1|\theta, x_i, \tau_i) = \mathbb{P}(t_i \in [t, t+h), c_i \geqslant t|x_i, \tau_i, \theta),$$

and

$$\mathbb{P}(u_i \in [t, t+h), \delta_i = 0|\theta, x_i, \tau_i) = \mathbb{P}(c_i \in [t, t+h), t_i \geqslant t|\theta, x_i, \tau_i).$$

By conditional independence we obtain

$$\begin{aligned}\mathbb{P}(u_i \in [t, t+h), \delta_i = 1|\theta, x_i, \tau_i) &= \mathbb{P}(t_i \in [t, t+h)|\theta, x_i, \tau_i)\mathbb{P}(c_i \geqslant t+h|\theta, x_i, \tau_i),\\ \mathbb{P}(u_i \in [t, t+h), \delta_i = 0|\theta, x_i, \tau_i) &= \mathbb{P}(c_i \in [t, t+h)|\theta, x_i, \tau_i)S(t+h|\theta, x_i, \tau_i).\end{aligned}$$

When h goes to zero, it tends respectively to

$$G(t|\theta, x_i, \tau_i)f(t|\theta, x_i, \tau_i),$$

and

$$g(t|\theta, x_i, \tau_i)\mathbb{P}(t_i \geqslant t|\theta, x_i, \tau_i).$$

Therefore, by the independence of the random variables (t_i, c_i) among the events $i \in \{1, \dots, N\}$ we obtain the density

$$f_{(u_i,\delta_i)_{1\leq i\leq N}}((u_i,\delta_i)_{1\leq i\leq N} \mid \theta, x_i, \tau_i) = \prod_{i=1}^{N} g(u_i|\theta, x_i, \tau_i)^{1-\delta_i} G(u_i|\theta, x_i, \tau_i)^{\delta_i} \prod_{i=1}^{N} f(u_i|\theta, x_i, \tau_i)^{\delta_i} S(u_i|\theta, x_i, \tau_i)^{1-\delta_i} .$$

Here we use the assumption of non-informative censoring (see Kalbfleisch et al. [74]), which means that the censored distribution does not involve the parameter θ. Then we obtain a simplified version of the likelihood, up to a multiplicative constant

$$\ell(\theta) \propto \prod_{i=1}^{N} f(u_i|\theta, x_i, \tau_i)^{\delta_i} S(u_i|\theta, x_i, \tau_i)^{1-\delta_i} .$$

Omitting an additional constant, we can equivalently write the log-likelihood to be

$$\log(\ell(\theta)) = \sum_{i=1}^{N} \delta_i \log(f(u_i|\theta, x_i, \tau_i)) + (1-\delta_i) \log(S(u_i|\theta, x_i, \tau_i)).$$

Let us remark that $f(t|\theta, x_i, \tau_i) = H(t|x_i, \tau_i)S(t|x_i, \tau_i)$ and from the definition of $H(t|x_i, \tau_i)$ we can write the log-likelihood as

$$\log(\ell(\theta)) = \sum_{i=1}^{N} \delta_i \log(H(u_i|x_i, \tau_i)) - \int_{\tau_i}^{u_i} H(s|x_i, \tau_i)ds.$$

Following the exponential model of Definition 1 we replace $H(t|x_i, \tau_i)$ in the previous equation to get

$$\log(\ell(\theta)) = \sum_{i=1}^{N} \delta_i \theta^T x_i(u_i) - \int_{\tau_i}^{u_i} \exp(\theta^T x_i(s))ds \,,$$

We write the negative log-likelihood :

$$\ell(\theta) = -\log(\ell(\theta)) = \sum_{i=1}^{N} -\delta_i \theta^T x_i(u_i) + \int_{\tau_i}^{u_i} \exp(\theta^T x_i(s))ds.$$

□

B2 Online Convex Optimization

B2.1 Proof of Lemma 1

Démonstration. Only the second assertion needs to be proven, the first one being Lemma 4.2.1 from Hazan [63] is already showed. To prove the second assertion we first see that Equation (2.5) means that for all $\theta \in \Theta$

$$\nabla^2 \ell(\theta) \succcurlyeq \mu \nabla \ell(\theta) \nabla \ell(\theta)^\top,$$

which implies that for all vector $\nu \in \mathbb{R}^d$

$$\nu^\top \nabla^2 \ell(\theta) \nu \geqslant \mu \, \nu^\top \nabla \ell(\theta) \nabla \ell(\theta)^\top \nu.$$

Since $\nabla \ell(\theta) \nabla \ell(\theta)^\top$ is a rank one matrix and $\nu = \nabla \ell(\theta)$ is an eigenvector associated to the unique non-null eigenvalue, we can replace ν in the previous equation to get

$$\nabla \ell(\theta)^\top \nabla^2 \ell(\theta) \nabla \ell(\theta) \geqslant \mu \nabla \ell(\theta)^\top \nabla \ell(\theta) \nabla \ell(\theta)^\top \nabla \ell(\theta).$$

When $\nabla \ell(\theta) \neq 0$ we can write

$$\mu \leqslant \frac{\nabla \ell(\theta)^\top \nabla^2 \ell(\theta) \nabla \ell(\theta)}{||\nabla \ell(\theta)||^4},$$

and as this is true for every $\theta \in \Theta$ we have

$$\mu \leqslant \min_{\theta \in \Theta} \frac{\nabla \ell(\theta)^\top \nabla^2 \ell(\theta) \nabla \ell(\theta)}{||\nabla \ell(\theta)||^4}.$$

□

B2.2 Proof of Lemma 2

Démonstration. The proof starts similarly than the one of Lemma 4.2.2 of Hazan [63]. We consider the concave function $p(\theta) = \exp(-\mu \ell(\theta))$. We derive that for $\theta_1, \theta_2 \in \Theta$:

$$\begin{aligned}
\ell(\theta_2) &\geq \ell(\theta_1) - \frac{1}{\mu} \log(1 - \mu(\nabla \ell(\theta_1)^T (\theta_2 - \theta_1))) \\
&\geq \ell(\theta_1) + \nabla \ell(\theta_1)^T (\theta_2 - \theta_1) \\
&\qquad - \left(\frac{1}{\mu} \log(1 - \mu(\nabla \ell(\theta_1)^T (\theta_2 - \theta_1))) + \nabla \ell(\theta_1)^T (\theta_2 - \theta_1) \right).
\end{aligned}$$

Using the Cauchy-Schwarz inequality we upper bound $|\nabla \ell(\theta_1)^T (\theta_2 - \theta_1)| \leq \|\nabla \ell(\theta_1)\| D$ for any $\theta_2 \in \Theta$. Combined with the monotonicity of the function $\mu^{-1} \log(1 - \mu z) + z$ which is decreasing for any $-\|\nabla \ell(\theta_1)\| D \leq z \leq \|\nabla \ell(\theta_1)\| D$ we obtain :

$$\ell(\theta_2) \geq \ell(\theta_1) + \nabla \ell(\theta_1)^T (\theta_2 - \theta_1) - \frac{1}{\mu} \log(1 + \mu \|\nabla \ell(\theta_1)\| D) + \|\nabla \ell(\theta_1)\| D \,.$$

By definition of the directional derivative constant, we thus can estimate :

$$\begin{aligned}
\gamma &\leq \min_{\theta_1, \theta_2 \in \Theta} 2 \frac{-\frac{1}{\mu} \log(1 + \mu \|\nabla \ell(\theta_1)\| D) + \|\nabla \ell(\theta_1)\| D}{(\nabla \ell(\theta_1)(\theta_2 - \theta_1))^2}, \\
&\leq \min_{\theta_1 \in \Theta} 2 \frac{-\frac{1}{\mu} \log(1 + \mu \|\nabla \ell(\theta_1)\| D) + \|\nabla \ell(\theta_1)\| D}{(\|\nabla \ell(\theta_1)\| D)^2}
\end{aligned}$$

by another application of the Cauchy-Schwarz inequality. □

B2.3 The Online Newton Step algorithm

Algorithm 3 Online Newton Step [65]

Input : $(\ell_t)_{t=1,2,\dots}, \gamma > 0, n \geqslant 1, \epsilon > 0$
Initialization : $\theta_0 \in \Theta,\ A_0^{-1} = (1/\epsilon)\mathbb{1}_d$
for iteration $t = 1, \dots, n$ **do**
Observe : $\nabla \ell_t(\theta_t)$
Recursion :

$$A_t^{-1} = A_{t-1}^{-1} - \frac{A_{t-1}^{-1}\nabla\ell_t(\theta_t)\nabla\ell_t(\theta_t)^T A_{t-1}^{-1}}{1 + \nabla\ell_t(\theta_t)A_{t-1}^{-1}\nabla\ell_t(\theta_t)^T}$$

$$\theta_{t+1} = \mathrm{Proj}_t\Big(\theta_t - \frac{1}{\gamma}A_t^{-1}\nabla\ell_t(\theta_t)\Big)$$

where $\mathrm{Proj}_t(\theta^*) \in \arg\min_{\theta\in\Theta}(\theta - \theta^*)^T A_t(\theta - \theta^*)$.
end for
return θ_n

B3 Stochastic Setting

In this section we prove Theorem 2, and for this we need to recall the hypothesis of Theorem 7 from [139].

(H1) The diameter of Θ is D and the loss functions ℓ_t are continuously differentiable over Θ a.s. with integrable gradients.

(H2) The random loss functions $(\ell_t)_{t=1,2,\dots}$ are stochastically exp-concave 4 for some $\gamma \geqslant 0$.

(H3) The gradients $(\nabla\ell_t(\theta_t))_{t=1,2,\dots}$, satisfy for $G_1, G_2 > 0$ and all $k \geqslant 1, t = 1, 2, \dots$, and $\theta \in \Theta$:

$$\begin{aligned}
\mathbb{E}_{t-1}[(\nabla\ell_t(\theta_t)^\top(\theta_t - \theta))^{2k}] &\leq k!(G_1 D)^{2(k-1)}\mathbb{E}_{t-1}[(\nabla\ell_t(\theta_t)^\top(\theta_t - \theta))^2] \qquad a.s.,\\
\mathbb{E}_{t-1}[||\nabla\ell_t(\theta)||^{2k}] &\leq k! G_1^{2(k-1)}\mathbb{E}_{t-1}[||\ell_t(\theta_t)||^2] \qquad a.s.,\\
\mathbb{E}_{t-1}[||\nabla\ell_t(\theta)||^2] &\leq G_2^2 \qquad\qquad a.s.
\end{aligned}$$

Let us notice that condition **(H3)** is satisfied in the bounded cases $||\nabla\ell_t(\theta_t)||^2 \leqslant G^2,\ t = 1, 2, \dots$ with $G_1 := G_2 := G$. Condition **(H3)** is independent on the risk $L_t(\theta_t) = \mathbb{E}_{t-1}[\ell_t(\theta_t)],\ t = 1, 2, \dots$, and thus, it does not interfere with condition **(H2)**. Additionally, we notice that in our setting where we consider the stochastic losses ℓ_t defined in (2.6), the hypothesis **(H1)** is already satisfied. Now, we recall the stochastic regret bound theorem.

Theorem 4 (Wintenberger [139]). Under **(H1)**, **(H2)** with constant γ and **(H3)**, for $\varrho > 0$ and $n \geqslant 1$ the ONS algorithm 3 with learning rate $\gamma/3$ satisfies with probability $1 - 3\varrho$ the stochastic

regret bound :

$$Risk_n \leqslant \frac{3}{2\gamma}\left(1 + d\log\left(1 + \frac{2(\gamma D)^2\left(nG_2^2 + G_1^2\log(\varrho^{-1})\right)}{9}\right)\right) + \left(\frac{4\gamma(G_1 D)^2}{9} + \frac{18}{\gamma}\right)\log(\varrho^{-1})\,.$$

In order to simplify notation we refer to the right-hand-side bound as $\mathcal{B}(n)$. In addition we need a proposition presented in [139] that gives us a constant γ that assures the stochastic exp-concavity of the losses.

Proposition 2 (Wintenberger [139]). If L_t is μ-strongly convex and there exists $G > 0$ such that

$$G^2 I_d \succcurlyeq \mathbb{E}_{t-1}[\nabla\ell_t(\theta)\nabla\ell_t(\theta)^T], \qquad \forall\theta \in \Theta, a.s., t = 1, 2, \dots\,,$$

then Definition 4 holds with $\gamma := \mu/G^2$.

In the ideal case, we would like to prove that ℓ_t satisfies the conditions **(H2)** and **(H3)**. To prove **(H2)** we can use Proposition 2 if we find a constant such that the loss is strongly convex and a constant that bounds the expectation of the gradients. Unfortunately, we are not able to find this last constant a.s. but, proving a weaker version of **(H3)** we can define an auxiliary loss function that satisfies all the hypothesis and allows us to prove Theorem 2.

First, we prove that with high probability there is a constant G that upper bounds the norm of the gradients of $(\ell_t)_{t=1,2,\dots}$, this corresponds to the weaker **(H3)**. Secondly, we prove that the conditional risks $(L_t)_{t=1,2,\dots}$ are strongly convex for some constant μ, which consists of finding a lower bound of $\nabla^2 L_t(\theta)$ that does not depend on θ and t. This corresponds to only one of the conditions of Proposition 2, necessary to prove **(H2)**. Finally, we show how to use weak **(H3)** and half of **(H2)** to prove Theorem 2.

B3.1 Upper bound (H3)

We want to find an upper bound for $||\nabla\ell_t(\theta)||^2$ and for this we first define for all $t = 1, 2, \dots$

$$R_t = \sum_{i=1}^{N_t} r_{it} \qquad \text{where} \qquad r_{it} = \mathbb{1}\{\tau_i \leqslant t, u_i > t-1\}\,,$$

and where N_t is the count function of the Poisson process defined in Section 2.4. Following, we prove that for all $t = 1, 2, \dots$, R_t is upper bounded with high probability.

Lemma 4. Let $\varrho > 0$. Then, with probability at least $1 - \varrho$, for all $t = 1, 2, \dots$, we have

$$R_t \leqslant 32e^{Dx_\infty}(4\lambda + 1 + \log(2/\varrho))\,.$$

Démonstration. Since $u_i = \min\{c_i, t_i\} \leqslant t_i$ we can upper bound $R_t \leqslant \sum_{i=1}^{\infty} \mathbb{1}\{t_i \geqslant t-1\}\mathbb{1}\{\tau_i \leqslant t\}$. Then, we define $A_t = \{i : \tau_i \leqslant t\}$ and $Z_t = \sum_{i\in A_t} \mathbb{1}\{t_i \geqslant t-1\}$ and therefore, it will be enough to

find a bound to Z_t to conclude. Given a constant $z > 0$ and $m \geqslant 1$, we fix $N_t \leqslant m$ and we first upper bound the conditional probability

$$\mathbb{P}(Z_t \geqslant z | N_t = m) = \mathbb{P}\left(\sum_{i \in A_t} \mathbb{1}\{t_i \geqslant t-1\} \geqslant z \Big| N_t = m \right).$$

Let us notice that $N_t = |A_t|$. We would like to apply the concentration inequality of Chernoff-Hoeffding to the sum of Bernouilli random variables $\mathbb{1}\{t_i \geqslant t-1\}$ (see Hoeffding [69]), and for this we need to upper bound

$$\begin{aligned}
&\mathbb{P}(t_i \geqslant t-1 | i \in A_t, x_i) \\
&\qquad = \sum_{s=1}^{t} \mathbb{P}(t_i \geqslant t-1 | s-1 \leqslant \tau_i \leqslant s, x_i) \mathbb{P}(s-1 \leqslant \tau_i \leqslant s | 0 \leqslant \tau_i \leqslant t) \\
&\qquad = \frac{1}{t} \sum_{s=1}^{t} \mathbb{P}(t_i \geqslant t-1 | s-1 \leqslant \tau_i \leqslant s, x_i),
\end{aligned}$$

where we use the uniform distribution of the Poisson process points given an interval (for more details on Poisson processes, see Daley and Vere-Jones [28]). Then, by the definition of the survival function (see Section 2.2) we get

$$\begin{aligned}
\mathbb{P}(t_i \geqslant t-1 | i \in A_t, x_i) &\leqslant \frac{1}{t} \sum_{s=1}^{t} S(t-1|s, x_i) \wedge 1 \\
&= \frac{1}{t} \sum_{s=1}^{t} \exp\left(-(t-1-s) e^{\theta^{*\top} x_i} \right) \wedge 1 \\
&\leqslant \frac{1}{t} \sum_{s=1}^{t} \exp\left(-(t-1-s) e^{-D x_\infty} \right) \wedge 1 \\
&\leqslant \frac{1}{t} \sum_{s=-1}^{\infty} \exp\left(-s e^{-D x_\infty} \right) \wedge 1 \\
&= \frac{2 - \exp(-e^{-D x_\infty})}{t\left(1 - \exp(-e^{-D x_\infty})\right)} \\
&\leqslant \frac{4\, e^{D x_\infty}}{t}.
\end{aligned}$$

In the last line we use that $1 - \exp(-x) \geq x/2$ for $0 \leq x \leq 1$.

The Chernoff-Hoeffding's inequality gives us for any sequence $X_1, \ldots, X_m$ with $\mathbb{E}[X_i] \leqslant p$ and any $\varepsilon > 0$

$$\mathbb{P}\left(\sum_{i=1}^{m} X_i \geqslant pm + \varepsilon \right) \leqslant \exp\left(-\frac{\varepsilon^2}{2mp(1-p)} \right) \leqslant \exp\left(-\frac{\varepsilon^2}{2mp} \right).$$

Applying this to the sum of the $\mathbb{1}\{t_i \geqslant t-1\}$ with $\mathbb{E}[\mathbb{1}\{t_i \geqslant t-1\} | i \in A_t] \leqslant \frac{e^{2+D x_\infty}}{t}$ given

$|A_t| = m$ and using the conditional independence of the Poisson process points, we obtain

$$\mathbb{P}\left(Z_t \geqslant \frac{m}{t} 4e^{Dx_\infty} + \varepsilon \middle| |A_t| = m\right) \leqslant \exp\left(-\frac{\varepsilon^2 t}{8me^{Dx_\infty}}\right).$$

Therefore, replacing $|A_t|$ by N_t

$$\mathbb{P}\left(Z_t \geqslant \frac{m}{t} 4e^{Dx_\infty} + \varepsilon \middle| N_t = m\right) \leqslant \exp\left(-\frac{\varepsilon^2 t}{8me^{Dx_\infty}}\right).$$

We set $z = \frac{m}{t} 4e^{Dx_\infty} + \varepsilon$ with which we get $\varepsilon = z - \frac{m}{t} 4e^{Dx_\infty}$ and

$$\mathbb{P}\left(Z_t \geqslant z \middle| N_t = m\right) \leqslant \exp\left(-\frac{\left(z - \frac{m}{t} 4e^{Dx_\infty}\right)^2 t}{8me^{Dx_\infty}}\right).$$

If we suppose $\frac{n}{t} 4e^{Dx_\infty} \leqslant \frac{z}{2}$ we obtain

$$\mathbb{P}\left(Z_t \geqslant z \middle| N_t = m\right) \leqslant \exp\left(-\frac{z^2 t}{32me^{Dx_\infty}}\right).$$

With this we found a bound for the conditional probability of Z_t being bigger than a certain constant. The next step is to bound $\mathbb{P}(Z_t \geqslant z)$, and for this we need to upper bound the probability of N_t being large. Let $M > 0$, since N_t follows a Poisson distribution of intensity λt, we can apply a Chernoff bound argument (more details in Mitzenmacher and Upfal [98]) obtaining

$$\mathbb{P}(N_t > M) \leqslant \left(\frac{e\lambda t}{M}\right)^M e^{-\lambda t} \qquad \text{for} \qquad M > \lambda t\,,$$

and

$$\mathbb{P}(N_t > M) \leqslant e^{-M-\lambda t} \qquad \text{when} \qquad M > e^2 \lambda t\,.$$

Now, we compute

$$\begin{aligned}
\mathbb{P}(Z_t \geqslant z) &= \sum_{m=1}^{M} \mathbb{P}(Z_t \geqslant z | N_t = m)\mathbb{P}(N_t = m) + \mathbb{P}(Z_t \geqslant z | N_t > M)\mathbb{P}(N_t > M) \\
&\leq \sum_{m=1}^{M} \exp\left(-\frac{z^2 t}{8me^{2+Dx_\infty}}\right) \mathbb{P}(N_t = m) \\
&\qquad\qquad + \mathbb{P}(Z_t \geqslant z | N_t > M)\mathbb{P}(N_t > M) \\
&\leqslant \exp\left(-\frac{z^2 t}{8Me^{2+Dx_\infty}}\right) + \exp(-M - \lambda t)\,,
\end{aligned}$$

where we use the bounds we previously found for $\mathbb{P}(Z_t \geqslant z | N_t = m)$ and $\mathbb{P}(N_t > M)$. Finally, we need to choose z and M such that $\mathbb{P}(Z_t \geqslant z) \leqslant \varrho/t^2$. Reminding the constrain $M > e^2 \lambda t$, we want

$$\exp(-M - \lambda t) \leqslant \frac{\varrho}{2t^2}\,,$$

which is true if

$$M \geqslant \log\left(\frac{2t^2}{\varrho}\right) - \lambda t$$

and then we can choose $M = e^2\lambda t + \log(2t^2/\varrho)$ that satisfies both conditions. Similarly, we want

$$\exp\left(-\frac{z^2 t}{32Me^{Dx_\infty}}\right) \leqslant \frac{\varrho}{2t^2}\,,$$

which is true if

$$z \geqslant \sqrt{\frac{32Me^{Dx_\infty}}{t}\log(2t^2/\varrho)}\,,$$

and reminding the constrain $z \geqslant \frac{8M}{t}e^{Dx_\infty}$ we choose z such that

$$z \geqslant \frac{8M}{t}e^{Dx_\infty} + 2\sqrt{\frac{8Me^{Dx_\infty}}{t}\log(2t^2/\varrho)}\,.$$

Due to Young's inequality $a + 2\sqrt{ab} \leqslant 2a + b$, we can also choose

$$z \geqslant \frac{16M}{t}e^{Dx_\infty} + \log(2t^2/\varrho)\,,$$

which replacing M becomes

$$\begin{aligned}
&\frac{16}{t}e^{Dx_\infty}(e^2\lambda t + \log(2t^2/\varrho)) + \log(2t^2/\varrho) \\
&\qquad = 16\lambda e^{Dx_\infty} + \left(1 + \frac{16}{t}e^{Dx_\infty}\right)\log(2t^2/\varrho) \\
&\qquad \leqslant 32e^{Dx_\infty}(4\lambda + 1 + \log(1/\varrho))\,.
\end{aligned}$$

In conclusion, we choose $z = 32e^{Dx_\infty}(4\lambda + 1 + \log(1/\varrho))$ and we get

$$\begin{aligned}
\mathbb{P}(Z_t \geqslant z) &= \mathbb{P}(Z_t \geqslant 32e^{Dx_\infty}(4\lambda + 1 + \log(1/\varrho))) \\
&\leqslant \exp\left(-\frac{z^2 t}{32Me^{Dx_\infty}}\right) + \exp(-M - \lambda t) \\
&\leqslant \frac{\varrho}{2t^2} + \frac{\varrho}{2t^2} \\
&= \frac{\varrho}{t^2}\,.
\end{aligned}$$

Using an upper-bound over t

$$\mathbb{P}(\forall t = 1, 2, \ldots \qquad Z_t \geqslant 32e^{Dx_\infty}(4\lambda + 1 + \log(1/\varrho))) \leqslant \sum_{t=1,2,\ldots}\frac{\varrho}{t^2} = \varrho\frac{\pi^2}{6} \leq 2\varrho\,,$$

which concludes the proof. □

Finally, we are now ready to give the desired upper bound for $||\nabla\ell_t(\theta)||^2$ in the following proposition

Proposition 3. Let $\varrho > 0$. Then, with probability $1 - \varrho$ we have

$$||\nabla \ell_t(\theta)||^2 \leqslant G^2 \qquad \forall \theta \in \Theta, t = 1, 2, \dots ,$$

with $G := 32e^{Dx_\infty}(4\lambda + 1 + \log(2/\varrho))\big(1 + e^{Dx_\infty}\big)x_\infty$.

Démonstration. Let us notice that $\nabla \ell_t(\theta) \in \mathbb{R}^d$ and recall

$$\nabla \ell_t(\theta) = \sum_{i=1}^{N_t} -y_{it}x_i + r_{it}\exp(\theta^\top x_i)x_i \left(u_i \wedge t - \tau_i \vee 0\right)_+ .$$

Then, we have

$$||\nabla \ell_t(\theta)|| \leqslant \sum_{i=1}^{N_t} ||y_{it}x_i|| + ||r_{it}\exp(\theta^\top x_i)x_i \left(u_i \wedge t - \tau_i \vee (t-1)\right)_+ || ,$$

noticing that $y_{it} \leqslant r_{it}$, $x_i \leqslant x_\infty$, $\exp(\theta^\top x_i) \leqslant \exp(Dx_\infty)$
and $\left(u_i \wedge t - \tau_i \vee (t-1)\right)_+ \leqslant 1$,

$$\begin{aligned} ||\nabla \ell_t(\theta)|| &\leqslant \sum_{i=1}^{N_t} ||r_{it}x_\infty|| + ||r_{it}\exp(Dx_\infty)x_\infty|| \\ &\leqslant \sum_{i=1}^{N_t} r_{it} \cdot \Big(1 + \exp(Dx_\infty)\Big)x_\infty \\ &\leqslant R_t\Big(1 + \exp(Dx_\infty)\Big)x_\infty , \end{aligned}$$

by definition of $R_t = \sum_{i=1}^{N_t} r_{it}$. In consequence,

$$||\nabla \ell_t(\theta)||^2 \leqslant \big(32e^{Dx_\infty}(4\lambda + 1 + \log(2/\varrho))\big(1 + \exp(Dx_\infty)\big)x_\infty\big)^2 ,$$

with probability $1 - \varrho$ and where the last inequality is due to Lemma 4. This conclude the proof. □

B3.2 Strong convexity (H2)

Before showing the strong convexity let us remark that we can write $S(t|x_i, \tau_i) = \exp\left(-\int_{\tau_i}^{t} H(s|x_i, \tau_i)ds\right)$ and because $f(t|x_i, \tau_i) = H(t|x_i, \tau_i)S(t|x_i, \tau_i)$ the density of t_i is given by

$$f(t|x_i, \tau_i) = H(t|x_i, \tau_i)\exp\left(-\int_{\tau_i}^{t} H(s|x_i, \tau_i)ds\right).$$

Given $\theta^* \in \Theta$, the real parameter and replacing by our parametric model $h(t|x_i, \theta^*, \tau_i) = \exp(\theta^{*T}x_i)$ we have

$$f(t|x_i, \tau_i) := \exp(\theta^{*T}x_i)\exp\left(-(t - \tau_i)\exp(\theta^{*T}x_i)\right)\mathbb{1}\{t \geqslant \tau_i\}. \tag{B.1}$$

We also denote

$$\ell_t(\theta; s, c, x, \tau) := \Big(- \mathbb{1}\{t-1 < s \leqslant t \wedge c\}\theta^T x + \exp(\theta^T x)\big((c \wedge s \wedge t) - (\tau \vee (t-1))\big)_+\Big)$$

and recalling that $y_{it} = \mathbb{1}\{t-1 < t_i \leqslant t \wedge c_i\}$, $u_i = t_i \wedge c_i$ and $r_{it} = \mathbb{1}\{\tau_i < t, u_i \geqslant t-1\}$, we have

$$\begin{aligned}\ell_t(\theta) &= \sum_{i=1}^{N_t} -y_{it}\theta^T x_i + r_{it}\exp(\theta^T x_i)\big((u_i \wedge t) - (\tau_i \vee (t-1))\big) \\ &= \sum_{i=1}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\,.\end{aligned}$$

In addition, as $(t_i)_{i\geqslant 1}$ and $(c_i)_{i\geqslant 1}$ are i.i.d. we name T and C random variables that are distributed as t_i and c_i, respectively. We first prove the following Lemma that gives us an explicit expression of the risk function $L_t(\theta) := \mathbb{E}_{t-1}[\ell_t(\theta)]$, $\theta \in \Theta$, $t = 1, 2, \ldots$.

Lemma 5. For every $t = 1, 2, \ldots$ and every $\theta \in \Theta$ the risk function is given by

$$\begin{aligned}L_t(\theta) &= \lambda\mathbb{E}\Big[(e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{1}\{T \leqslant C\}(1-T)_+ \mid \tau = 0\Big] \\ &+ \sum_{\substack{i:\{u_i > t-1\} \\ i:\{\tau_i \leqslant t-1\}}} \Big(e^{(\theta-\theta^*)^\top x_i} - \theta^\top x_i\Big)\,\mathbb{P}(\mathbb{1}\{t-1+\tau_i < T \leqslant \tau_i + t \wedge C\}|x_i, \tau_i, \tau = 0)\,.\end{aligned}$$

Démonstration. The expected value is

$$\mathbb{E}_{t-1}[\ell_t(\theta)] = \mathbb{E}_{t-1}\left[\sum_{i=1}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\right],$$

which we separate in two terms

$$\mathbb{E}_{t-1}[\ell_t(\theta)] = \mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\right] + \sum_{i=1}^{N_{t-1}} \mathbb{E}_{t-1}\left[\ell_t(\theta; t_i, c_i, x_i, \tau_i)|x_i, \tau_i\right]. \tag{B.2}$$

Now, recalling that g and f respectively denote the conditional densities of t_i and c_i given (τ_i, x_i) and, because c_i and t_i are independent given (τ_i, x_i), we first calculate the first term

$$\begin{aligned}&\mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\right] \\ &\qquad = \mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \mathbb{E}\left[\ell_t(\theta; t_i, c_i, x_i, \tau_i)|x_i, \tau_i\right]\right] \\ &\qquad = \mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \ell_t(\theta; s, c, x_i, \tau_i) g(c|x_i, \tau_i) f(s|x_i, \tau_i) ds dc\right].\end{aligned}$$

Now because x_i are i.i.d. and independent from τ_i and c_i, denoting by x a random variable

with the same distribution we have

$$\mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\right] = \mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \ell_t(\theta; s, c, x, \tau_i) g(c|x, \tau_i) f(s|x, \tau_i) ds dc\right]$$

which can be written as the stochastic integral

$$\begin{aligned}
&\mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\right] \\
&= \mathbb{E}\left[\int_{t-1}^{t}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \ell_t(\theta; s, c, x, v) g(c|x, v) f(s|x, v) ds dc dN(v)\right] \\
&= \lambda\mathbb{E}\left[\int_{t-1}^{t}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \ell_t(\theta; s, c, x, v) g(c|x, v) f(s|x, v) ds\, dc\, dv\right] \\
&= \lambda\mathbb{E}\left[\int_{t-1}^{t}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \ell_t(\theta; s+v, c+v, x, v) g(c+v|x, v) f(s+v|x, v) ds\, dc\, dv\right] \\
&= \lambda\mathbb{E}\left[\int_{t-1}^{t}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \ell_t(\theta; s+v, c+v, x, v) g(c|x, 0) f(s|x, 0) ds\, dc\, dv\right] \\
&= \lambda\mathbb{E}\left[\int_{-\infty}^{\infty}\int_{t-1}^{t}\int_{-\infty}^{\infty} \ell_t(\theta; s+v, c+v, x, v)\, f(s|x, 0)\, ds\, dv\, g(c|x, 0) dc\right].
\end{aligned}$$

We do not know $g(c|x, v)$ but we know that $g(c|x, v) = g(c - \epsilon|x, v - \epsilon)$ for all $\epsilon \in \mathbb{R}$. For instance, $g(c|x, v) = g(c - v|x, 0)$ and, the same is satisfied for f. Then, we change the variable $v \in [0, t]$ in $w = v - (t - 1)$:

$$\int_{t-1}^{t} \ell_t(\theta; s+v, c+v, x, v)\, dv = \int_{0}^{1} \ell_1(\theta; s+w, c+w, x, w)\, dw\,.$$

We obtain

$$\mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\right] = \lambda\mathbb{E}\left[\int_{-\infty}^{\infty}\int_{0}^{1}\int_{-\infty}^{\infty} \ell_1(\theta; s+w, c+w, x, w) f(s|x, 0) ds\, dw\, g(c|x, 0) dc\right].$$

Considering the integral on the time s

$$\int_{-\infty}^{\infty} \ell_1(\theta; s+w, c+w, x, w) f(s|x, 0) ds = \int_{0}^{\infty} \ell_1(\theta; s+w, c+w, x, w) e^{\theta^{*T} x} \exp(-s e^{\theta^{*T} x}) ds\,,$$

we obtain

$$-\int_{(-w)_+}^{(1-w)\wedge c} \theta^T x e^{\theta^{*T} x} \exp(-s e^{\theta^{*T} x}) ds \\ = -\theta^T x \mathbb{P}((-w)_+ < T \leq (1-w)\wedge c | \tau = 0)\,,$$

and

$$\int_{(1-w)\wedge c}^{\infty} ((c+w)\wedge(s+w)\wedge 1 - w_+)_+ e^{\theta^T x} e^{\theta^{*T} x} \exp(-s e^{\theta^{*T} x}) ds \\ = e^{\theta^T x}((c+w)\wedge 1 - w_+)_+ \mathbb{P}(T \geq (1-w)\wedge c | \tau = 0)\,,$$

and

$$\int_{(-w)_+}^{(1-w)\wedge c} ((s+w) - w_+) e^{\theta^T x} e^{\theta^{*T} x} \exp(-s e^{\theta^{*T} x}) ds \\ = e^{\theta^T x}\Big(- ((s+w) - w_+)\exp(-s e^{\theta^{*T} x}) \,|_{(-w)_+}^{(1-w)\wedge c} + \int_{(-w)_+}^{(1-w)\wedge c} \exp(-s e^{\theta^{*T} x}) ds\Big) \\ = -e^{\theta^T x}((c+w)\wedge 1 - w_+)_+ \mathbb{P}(T \geq (1-w)\wedge c | \tau = 0) \\ + \exp((\theta - \theta^*)^T x)\mathbb{P}((-w)_+ < T \leq (1-w)\wedge c | \tau = 0)\,.$$

All in all we obtain

$$\int_0^{\infty} \ell_1(\theta; s+w, c+w, x, w) e^{\theta^{*T} x} \exp(-s e^{\theta^{*T} x}) ds \\ = (e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{P}((-w)_+ < T \leq (1-w)\wedge c | \tau = 0)\,,$$

thus we have

$$\mathbb{E}\left[\sum_{i=N_{t-1}}^{N_t} \ell_t(\theta; t_i, c_i, x_i, \tau_i)\right] \\ = \lambda \mathbb{E}\left[(e^{(\theta-\theta^*)^T x} - \theta^T x)\int_{-\infty}^{\infty}\int_0^1 \mathbb{P}((-w)_+ < T \leq (1-w)\wedge c | \tau = 0) dw\, g(c|x,0) dc\right] \\ = \lambda \mathbb{E}\Big[(e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{E}\Big[\int_0^1 \mathbb{1}\{(-w)_+ < T \leq (1-w)\wedge C\} dw \Big| x, \tau = 0\Big]\Big] \\ = \lambda \mathbb{E}\Big[(e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{E}\Big[\int_{0\vee -T}^{1-T} \mathbb{1}\{T \leqslant C\} dw \Big| x, \tau = 0\Big]\Big] \\ = \lambda \mathbb{E}\Big[(e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{E}\Big[\mathbb{1}\{T \leqslant C\}((1-T) - 0 \vee -T)_+ \Big| x, \tau = 0\Big]\Big] \\ = \lambda \mathbb{E}\Big[(e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{E}\Big[\mathbb{1}\{T \leqslant C\}(1-T)_+ \Big| x, \tau = 0\Big]\Big].$$

Replacing in Equation (B.2) we have

$$\mathbb{E}_{t-1}[\ell_t(\theta)] = \lambda\mathbb{E}\Big[(e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{E}\Big[\mathbb{1}\{T \leqslant C\}(1-T)_+\Big|x, \tau = 0\Big]\Big] + \sum_{i=1}^{N_{t-1}} \mathbb{E}_{t-1}\left[\ell_t(\theta; t_i, c_i, x_i, \tau_i)|x_i, \tau_i\right] .$$

To calculate the second term, we note that we know u_i if $u_i \leqslant t-1$ and in this case $\ell_t(\theta; t_i, c_i, x_i, \tau_i) = 0$, therefore, we consider only the individuals i such that $u_i > t-1$. The sum becomes

$$\sum_{i=1}^{N_{t-1}} \mathbb{E}_{t-1}\left[\ell_t(\theta; t_i, c_i, x_i, \tau_i)|x_i, \tau_i\right] = \sum_{\substack{i:\{u_i > t-1\}\\ i:\{\tau_i \leqslant t-1\}}} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \ell_t(\theta; s, c, x_i, \tau_i) g(c|x_i, \tau_i) f(s|x_i, \tau_i) ds dc\,,$$

which, following the calculations of the first term we obtain

$$\sum_{i=1}^{N_{t-1}} \mathbb{E}_{t-1}\left[\ell_t(\theta; t_i, c_i, x_i, \tau_i)|x_i, \tau_i\right] = \sum_{\substack{i:\{u_i > t-1\}\\ i:\{\tau_i \leqslant t-1\}}} \left(e^{(\theta-\theta^*)^\top x_i} - \theta^\top x_i\right) \mathbb{P}(t-1+\tau_i < T \leqslant \tau_i + t \wedge C | x = x_i, \tau_i, \tau = 0)\,,$$

where (T, C, x, τ) is independent of (t_i, c_i, x_i, τ_i) for every $i \geq 1$. Let us notice that τ_i and x_i, which we suppose are observed at the same time as τ_i, are known at time $t-1$. Replacing this term in Equation (B.2) leads to

$$\begin{aligned}\mathbb{E}_{t-1}[\ell_t(\theta)] &= \lambda\mathbb{E}\Big[(e^{(\theta-\theta^*)^T x} - \theta^T x)\mathbb{1}\{T \leqslant C\}(1-T)_+ \mid \tau = 0\Big] \\ &\quad + \sum_{\substack{i:\{u_i > t-1\}\\ i:\{\tau_i \leqslant t-1\}}} \left(e^{(\theta-\theta^*)^\top x_i} - \theta^\top x_i\right) \mathbb{P}(t-1+\tau_i < T \leqslant \tau_i + t \wedge C | x_i, \tau_i, \tau = 0),\end{aligned}$$

that finalizes the proof. □

We define

$$J(\theta) := \lambda\mathbb{E}\Big[(e^{(\theta-\theta^*)^\top x} - \theta^T x)\mathbb{1}\{T \leqslant C\}(1-T)_+\Big|x, \tau = 0\Big]\,, \tag{B.3}$$

and we are ready to show the strong convexity of the risk function that we give in the following proposition.

Proposition 4. The risk function satisfies

$$\nabla^2 L_t(\theta) \succcurlyeq \lambda e^{-Dx_\infty}\mathbb{E}[xx^\top \mathbb{1}\{T \leqslant C\}(1-T)_+|\tau = 0], \qquad \forall\theta \in \Theta, t = 1, \ldots, n\,.$$

Therefore, under Assumption 2 L_t is μ-strongly convex for $\mu = \lambda e^{-Dx_\infty} A$.

Démonstration. Lemma 5 gives us an expression of the risk $L_t =: J + R_t$ with R_t some random

convex function. By convexity $\nabla^2 L_t(\theta) \succcurlyeq \nabla^2 J(\theta)$, $\theta \in \Theta$, and therefore, it is enough to bound the hessian of the first term J. We calculate

$$\nabla J(\theta) = \lambda \mathbb{E}\Big[(e^{(\theta-\theta^*)^\top x} - 1)x \mathbb{1}\{T \leqslant C\}(1-T)_+ \big| x, \tau = 0\Big]$$

and

$$\nabla^2 J(\theta) = \lambda \mathbb{E}\Big[e^{(\theta-\theta^*)^\top x} x x^\top \mathbb{1}\{T \leqslant C\}(1-T)_+ \big| x, \tau = 0\Big] .$$

Let us notice that $e^{(\theta-\theta^*)^\top x} \geqslant e^{-D\|x\|_\infty}$ and then

$$\nabla^2 J(\theta) \succcurlyeq \lambda e^{-D\|x\|_\infty} \mathbb{E}[x x^\top \mathbb{1}\{T \leqslant C\}(1-T)_+ | x, \tau = 0] ,$$

which due to Assumption 2 concludes the proof. □

B3.3 Proof of Theorem 2

Démonstration. First of all, we remind that Proposition 3 implies **(H3)** with $G = G_1 = G_2$, but this bound for the gradients is satisfied with probability $1 - \varrho$ instead of almost surely and therefore we cannot claim that **(H3)** is always fulfilled. But there is a problem with this definition because **(H3)** considers all $t = 1, 2, \ldots$ and, in order to have a $\mathcal{F}_t$-measurable function we need to define a time dependent $\textbf{(H3)}_t$:

$\textbf{(H3)}_t$ For $t + 1 \geqslant s \geqslant 1$ the gradients $\nabla \ell_s(\theta_s)$, satisfy for $G > 0$ and all $k \geqslant 1$, and $\theta \in \Theta$:

$$\begin{aligned}
\mathbb{E}_{s-1}[(\nabla \ell_s(\theta_s)^\top (\theta_s - \theta))^{2k}] &\leq k!(GD)^{2(k-1)} \mathbb{E}_{s-1}[(\nabla \ell_s(\theta_s)^\top (\theta_s - \theta))^2] , \\
\mathbb{E}_{s-1}[||\nabla \ell_s(\theta)||^{2k}] &\leq k! G^{2(k-1)} \mathbb{E}_{s-1}[||\ell_s(\theta_s)||^2] , \\
\mathbb{E}_{s-1}[||\nabla \ell_t(\theta)||^2] &\leq G^2 .
\end{aligned}$$

We define $\Omega_t = \{(y_{is}, x_i, \tau_i, u_i)_{s \leqslant t}$ such that $\textbf{(H3)}_t$ is satisfied$\}$ for all $t = 1, 2, \ldots$ sand we check that Ω_t is $\mathcal{F}_t$-measurable. Next, for all $t = 1, 2, \ldots$ we define the auxiliary loss function

$$\hat{\ell}_t(\theta) = \ell_t(\theta) \mathbb{1}\{\Omega_{t-1}\} ,$$

which is $\mathcal{F}_t$-measurable. Let us notice that we need to define Ω_t using $\textbf{(H3)}_t$ instead of the inequality of Proposition 3 to preserve the past dependency and the measurability. We prove that the function $\hat{\ell}_t$ satisfies the conditions **(H1)**, **(H2)** and **(H3)**.

First of all, **(H1)** is still verified because the indicator function does not depend on θ. Secondly, if $\textbf{(H3)}_t$ is not realized then the function $\hat{\ell}_t$ is zero and all the bounds hold. Thirdly, if $\textbf{(H3)}_t$ is realized, ℓ_t satisfies the inequalities of **(H3)** and $\hat{\ell}_t = \ell_t$ by definition. Then the bounds in **(H3)** are also true for $\hat{\ell}_t$, concluding that $\hat{\ell}_t$ satisfies the inequalities of **(H3)** for all $t = 1, 2, \ldots$. Finally, it remains to prove **(H2)**.

By $\mathcal{F}_{t-1}$-measurability of Ω_{t-1} we calculate for $\theta \in \Theta$:

$$\mathbb{E}_{t-1}[\nabla \hat{\ell}_t(\theta) \nabla \hat{\ell}_t(\theta)^\top] = \mathbb{1}\{\Omega_{t-1}\} \mathbb{E}_{t-1}[\nabla \ell_t(\theta) \nabla \ell_t(\theta)^\top] .$$

If $\mathbf{(H3)}_t$ is not realized, $\mathbb{1}\{\Omega_{t-1}\} = 0$ and so (4) is true for any constant $\gamma \geqslant 0$. If $\mathbf{(H3)}_t$ is realized, $\mathbb{1}\{\Omega_{t-1}\} = 1$ and there exist $G > 0$ such that :

$$\mathbb{E}_{t-1}[\nabla\hat{\ell}_t(\theta)\nabla\hat{\ell}_t(\theta)^\top] = \mathbb{E}_{t-1}[\nabla\ell_t(\theta)\nabla\ell_t(\theta)^\top] \preccurlyeq G^2\mathcal{I}_d\,.$$

This, together with the strong convexity of Proposition 4 give us the hypothesis of Proposition 2 assuring the stochastic exp-concavity for $\gamma = \lambda e^{-Dx_\infty}A/G^2$ and concluding **(H2)**. Now, we have that $\hat{\ell}_t$ satisfies all the conditions of Theorem 4 assuring the logarithmic stochastic regret bound of ONS.

To study the stochastic regret bound we need also to define for all $t = 1, 2, \ldots$ the risk function $\hat{L}_t(\theta) = \mathbb{E}_{t-1}[\hat{\ell}_t(\theta)]$ and we notice that as $\mathbb{1}\{\Omega_{t-1}\}$ is $\mathcal{F}_{t-1}$-measurable :

$$\hat{L}_t(\theta) = \mathbb{1}\{\Omega_{t-1}\}\mathbb{E}_{t-1}[\ell_t(\theta)] = \mathbb{1}\{\Omega_{t-1}\}L_t(\theta)\,, \qquad \theta \in \Theta.$$

Now, it remains to prove that ONS has logarithmic stochastic regret also for L_t and therefore, we calculate for every $n \geq 1, t = 1, \ldots, n$,
$\theta^* \in \arg\min_{\theta\in\Theta}\sum_{t=1}^n L_t(\theta)$ and θ_t the prediction of ONS at time t :

$$\begin{aligned}
\mathbb{P}\left[\sum_{t=1}^n L_t(\theta_t) - L_t(\theta) > \mathcal{B}(n)\right] &= \mathbb{P}\left[\sum_{t=1}^n L_t(\theta_t) - L_t(\theta^*) > \mathcal{B}(n), \bigcap_{t\geqslant 2}\Omega_{t-1}\right] \\
&\quad + \mathbb{P}\left[\sum_{t=1}^n L_t(\theta_t) - L_t(\theta^*) > \mathcal{B}(n), \left(\bigcap_{t\geqslant 2}\Omega_{t-1}\right)^c\right] \\
&\leqslant \mathbb{P}\left[\sum_{t=1}^n \left(L_t(\theta_t) - L_t(\theta^*)\right)\mathbb{1}\{\Omega_{t-1}\} > \mathcal{B}(n)\right] \\
&\quad + \mathbb{P}\left[\left(\bigcap_{t\geqslant 2}\Omega_{t-1}\right)^c\right]\,,
\end{aligned}$$

where $\mathcal{B}(n)$ is the stochastic regret bound for $\hat{L}(\theta)$ of Theorem 4 which we remind :

$$\begin{aligned}
\mathcal{B}(n) = \frac{3}{2\gamma}\left(1 + d\log\left(1 + \frac{2(\gamma D)^2G^2\left(n + \log(\varrho^{-1})\right)}{9}\right)\right) \\
+ \left(\frac{4\gamma(GD)^2}{9} + \frac{18}{\gamma}\right)\log(\varrho^{-1})\,.
\end{aligned}$$

Plugging in $\mathcal{B}(n)$ the specific values of γ, G and μ found in Propositions 2, 3, and 4, respectively,

$$\gamma = \frac{\mu}{G^2} = \frac{\lambda e^{-Dx_\infty}A}{(32e^{Dx_\infty}(4\lambda + 1 + \log(2/\varrho))(1 + e^{Dx_\infty})x_\infty)^2}\,,$$

we obtain the regret bound

$$Risk_n \leqslant \frac{3G^2 e^{Dx_\infty}}{2\lambda A}\left(1 + d\log\left(1 + \frac{2(\lambda AD)^2\left(n + \log(\varrho^{-1})\right)}{9G^2 e^{2Dx_\infty}}\right)\right) + \left(\frac{4\lambda AD^2}{9e^{Dx_\infty}} + \frac{18Ge^{Dx_\infty}}{\lambda A}\right)\log(\varrho^{-1})\,. \tag{B.4}$$

Then, because of Theorem 4, this bound holds with probability 3ϱ and as

$$\mathbb{P}\left[\left(\bigcap_{t\geqslant 2}\Omega_{t-1}\right)^c\right] \leqslant \varrho\,,$$

we have :

$$\mathbb{P}\left[\sum_{t=1}^{n} L_t(\theta_t) - L_t(\theta) > \mathcal{O}(\log(n/\varrho))\right] \leqslant 4\varrho\,,$$

and thus, with probability $1 - 4\varrho$, ONS algorithm has logarithmic stochastic regret. □

B3.4 Proof of Corollary 1

Démonstration. Due to the μ-strong convexity of $L_t(\theta)$ proved in Proposition 4 we have for all $t = 1, 2, \ldots$:

$$\mu||\theta_t - \theta^*||^2 \leqslant \nabla L_t(\theta^*)^\top(\theta_t - \theta^*) + \mu||\theta_t - \theta^*||^2 \leqslant L_t(\theta_t) - L_t(\theta^*)\,,$$

where the first inequality is true because $\nabla L_t(\theta^*)^\top(\theta_t - \theta^*) \geqslant 0$. Then, because of Theorem 2 :

$$\sum_{t=1}^{n}||\theta_t - \theta^*||^2 \leqslant \frac{1}{\mu}\sum_{t=1}^{n} L_t(\theta_t) - L_t(\theta^*) \leqslant \frac{1}{\mu}\mathcal{B}(n)\,,$$

and remembering that $\mu = \lambda e^{-Dx_\infty}A$, the bound is :

$$\frac{1}{\mu}\mathcal{B}(n) = \frac{3G^2 e^{2Dx_\infty}}{2\lambda^2 A^2}\left(1 + d\log\left(1 + \frac{2(\lambda AD)^2\left(n + \log(\varrho^{-1})\right)}{9G^2 e^{2Dx_\infty}}\right)\right) + \left(\frac{4\lambda AD^2}{9e^{Dx_\infty}} + \frac{18Ge^{Dx_\infty}}{\lambda A}\right)\log(\varrho^{-1})\,, \tag{B.5}$$

which is $\mathcal{O}(\log(n/\varrho))$. We conclude the convergency of θ_t to θ^* and then :

$$||\bar{\theta}_n - \theta^*||^2 \leqslant \frac{1}{n}\sum_{t=1}^{n}||\theta_t - \theta^*||^2 - \frac{1}{n}\sum_{t=1}^{n}||\theta_t - \bar{\theta}_n||^2 \leqslant \frac{1}{\mu}\mathcal{O}(\log(n/\varrho)/n)\,,$$

concluding the convergency of $\bar{\theta}_n$ to θ^*. □

B4 Survival ONS

B4.1 Proof of Lemma 3

Démonstration. We first compute

$$\begin{aligned}\nabla\hat{\ell}_{t,\gamma}(\theta_1) &= \nabla\ell_t(\hat{\theta}) + \gamma\left(\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta})\right)\nabla\ell_t(\hat{\theta}) \\ &= \left(1+\gamma\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta})\right)\nabla\ell_t(\hat{\theta})\,.\end{aligned}$$

We need to show that there exists $\hat{\gamma} > 0$ such that

$$\hat{\ell}_{t,\gamma}(\theta_2) \geqslant \hat{\ell}_{t,\gamma}(\theta_1) + \nabla\hat{\ell}_{t,\gamma}(\theta_1)(\theta_2-\theta_1) + \frac{\hat{\gamma}}{2}\left(\nabla\hat{\ell}_{t,\gamma}(\theta_1)(\theta_2-\theta_1)\right)^2$$

and if we replace $\hat{\ell}_{t,\gamma}$ this inequality is equivalent to

$$\begin{aligned}&\ell_t(\hat{\theta}) + \nabla\ell_t(\hat{\theta})(\theta_2-\hat{\theta}) + \frac{\gamma}{2}\left(\nabla\ell_t(\hat{\theta})(\theta_2-\hat{\theta})\right)^2 \\ &\geqslant \ell_t(\hat{\theta}) + \nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta}) + \frac{\gamma}{2}\left(\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta})\right)^2 \\ &\quad + \left(1+\gamma\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta})\right)\nabla\ell_t(\hat{\theta})(\theta_2-\theta_1) \\ &\quad + \frac{\hat{\gamma}}{2}\left((1+\gamma\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta}))\nabla\ell_t(\hat{\theta})(\theta_2-\theta_1)\right)^2.\end{aligned}$$

Grouping, this requirement becomes

$$\begin{aligned}\frac{\gamma}{2}\left(\nabla\ell_t(\hat{\theta})(\theta_2-\hat{\theta})\right)^2 \geqslant \frac{\gamma}{2}&\left(\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta})\right)^2 \\ &+ \gamma\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta})\nabla\ell_t(\hat{\theta})(\theta_2-\theta_1) \\ &+ \frac{\hat{\gamma}}{2}\left((1+\gamma\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta}))\nabla\ell_t(\hat{\theta})(\theta_2-\theta_1)\right)^2,\end{aligned}$$

which is

$$0 \geqslant \left(\frac{\hat{\gamma}}{2}(1+\gamma\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta}))^2 + \frac{\gamma}{2}\right)\left(\nabla\ell_t(\hat{\theta})(\theta_2-\theta_1)\right)^2.$$

To satisfy this inequality we need

$$\hat{\gamma} \leqslant \frac{\gamma}{(1+\gamma\nabla\ell_t(\hat{\theta})(\theta_1-\hat{\theta}))^2},$$

which is true for the choice $\hat{\gamma} = \frac{\gamma}{(1+\gamma D||\nabla\ell_t(\hat{\theta})||)^2}$ and this concludes the proof. □

B4.2 Proof of Theorem 3

Démonstration. At each iteration t we consider $\hat{\theta}_t$ the prediction of SurvONS and $\theta_t(\gamma)$ the prediction of ONS with $\gamma \in \Gamma$. We define the directional derivative lower bound function as in

Equation (2.7)

$$\hat{\ell}_{t,\gamma_t}(\theta) = \ell_t(\hat{\theta}_t) + \nabla\ell_t(\hat{\theta}_t)(\theta - \hat{\theta}_t) + \frac{\gamma_t}{2}\left(\nabla\ell_t(\hat{\theta}_t)(\theta - \hat{\theta}_t)\right)^2.$$

Let us notice that $\gamma \leqslant \frac{1}{4GD}$ and $\hat{\ell}_{t,\gamma_t}(\theta) \leqslant \ell_t(\theta)$ for all θ.

We take $\theta^* \in \arg\min_{\theta\in\Theta} \sum_{t=1}^n \ell_t(\theta)$ and we can upper-bound the regret for any $\gamma \in \Gamma$

$$\begin{aligned}
Regret_n &= \sum_{t=1}^n \ell_t(\hat{\theta}_t) - \ell_t(\theta^*) \\
&\leqslant \sum_{t=1}^n \hat{\ell}_{t,\gamma_t}(\hat{\theta}_t) - \hat{\ell}_{t,\gamma_t}(\theta^*) \\
&= \sum_{t=1}^n \hat{\ell}_{t,\gamma_t}(\hat{\theta}_t) - \hat{\ell}_{t,\gamma_t}(\theta_t(\gamma)) + \hat{\ell}_{t,\gamma_t}(\theta_t(\gamma)) - \hat{\ell}_{t,\gamma_t}(\theta^*) \\
&= \sum_{t=1}^n \nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma)) - \sum_{t=1}^n \frac{\gamma_t}{2}\left(\nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma))\right)^2 \\
&\qquad + \sum_{t=1}^n \hat{\ell}_{t,\gamma_t}(\theta_t(\gamma)) - \hat{\ell}_{t,\gamma_t}(\theta^*)\,.
\end{aligned}$$

We upper-bound the first term using the regret-bound of BOA [138] which works for $\gamma \leqslant \frac{1}{4GD}$

$$\sum_{t=1}^n \nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma)) \leqslant \frac{\log(K)}{\gamma} + 2\gamma\sum_{t=1}^n \left(\nabla\ell(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma))\right)^2.$$

Therefore, the regret is bounded by

$$\begin{aligned}
Regret_n \leqslant \frac{\log(K)}{\gamma} + \sum_{t=1}^n \left(\frac{4\gamma - \gamma_t}{2}\right)\left(\nabla\ell(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma))\right)^2 \\
+ \sum_{t=1}^n \hat{\ell}_{t,\gamma_t}(\theta_t(\gamma)) - \hat{\ell}_{t,\gamma_t}(\theta^*).
\end{aligned}$$

We consider the surrogate losses $\hat{\ell}_{t,\hat{\gamma}_t}$ for $t = 1, 2, \ldots$ and $\hat{\gamma}_t = 4\max\{\gamma, \gamma_t/4\}$

$$\hat{\ell}_{t,\hat{\gamma}_t}(\theta) = \ell_t(\hat{\theta}_t) + \nabla\ell_t(\hat{\theta}_t)(\theta - \hat{\theta}_t) + 2\max\{\gamma, \frac{\gamma_t}{4}\}\left(\nabla\ell_t(\hat{\theta}_t)(\theta - \hat{\theta}_t)\right)^2,$$

and we write the last term of the regret bound

$$\begin{aligned}
&\sum_{t=1}^{n} \hat{\ell}_{t,\gamma_t}(\theta_t(\gamma)) - \hat{\ell}_{t,\gamma_t}(\theta^*) \\
&\quad= \sum_{t=1}^{n} \left(\hat{\ell}_t(\theta_t(\gamma);\gamma) - \hat{\ell}_t(\theta^*;\gamma)\right) + \sum_{t=1}^{n} \left(\hat{\ell}(\theta_t(\gamma)) - \hat{\ell}_t(\theta_t(\gamma);\gamma)\right) \\
&\qquad\quad - \sum_{t=1}^{n} \left(\hat{\ell}_t(\theta^*) - \hat{\ell}_t(\theta^*;\gamma)\right) \\
&\quad= \sum_{t=1}^{n} \left(\hat{\ell}_t(\theta_t(\gamma);\gamma) - \hat{\ell}_t(\theta^*;\gamma)\right) \\
&\qquad\quad - \sum_{t=1}^{n} \frac{(4\gamma-\gamma_t)_+}{2} \left(\nabla \ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma))\right)^2 \\
&\qquad\qquad\qquad + \sum_{t=1}^{n} \frac{(4\gamma-\gamma_t)_+}{2} \left(\nabla \ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta^*)\right)^2 .
\end{aligned}$$

We substitute this expression in the regret bound

$$\begin{aligned}
Regret_n \leqslant \frac{\log(K)}{\gamma} &+ \sum_{t=1}^{n} \left(\hat{\ell}_t(\theta_t(\gamma);\gamma) - \hat{\ell}_t(\theta^*;\gamma)\right) \\
&- \sum_{t=1}^{n} \frac{(\gamma_t-4\gamma)_+}{2} \left(\nabla \ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma))\right)^2 \\
&+ \sum_{t=1}^{n} \frac{(4\gamma-\gamma_t)_+}{2} \left(\nabla \ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta^*)\right)^2 .
\end{aligned}$$

Now, we note that by Lemma 3 we have

$$\begin{aligned}
\hat{\ell}_t(\theta_t(\gamma);\gamma) - \hat{\ell}_t(\theta^*;\gamma) \leqslant \nabla \hat{\ell}_t(\theta_t(\gamma);\gamma)(\theta_t(\gamma) - \theta^*) \\
- \frac{\hat{\gamma}_t}{2} \left(\nabla \hat{\ell}_t(\theta_t(\gamma);\gamma)(\theta_t(\gamma) - \theta^*)\right)^2 ,
\end{aligned}$$

where we can write $\max\{\gamma, \gamma_t/4\} = \gamma + (\gamma_t/4 - \gamma)_+$ and get

$$\hat{\gamma}_t = \frac{4(\gamma + (\gamma_t/4-\gamma)_+)}{(1 + 4(\gamma + (\gamma_t/4-\gamma)_+)(\nabla \ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \hat{\theta}_t)))^2} \geqslant \gamma .$$

Therefore, we can apply the regret bound of ONS which yields to

$$\begin{aligned}
\sum_{t=1}^{n} \hat{\ell}_t(\theta_t(\gamma);\gamma) - \hat{\ell}_t(\theta^*;\gamma) \leqslant \frac{5d\log(n)}{\gamma} &+ \frac{\gamma}{2} \sum_{t=1}^{n} \left(\nabla \hat{\ell}_t(\theta_t(\gamma);\gamma)(\theta_t(\gamma) - \theta^*)\right)^2 \\
&- \sum_{t=1}^{n} \frac{\hat{\gamma}_t}{2} \left(\nabla \hat{\ell}_t(\theta_t(\gamma);\gamma)(\theta_t(\gamma) - \theta^*)\right)^2 . \qquad \text{(B.6)}
\end{aligned}$$

But, since

$$\nabla\hat{\ell}_t(\theta_t(\gamma);\gamma) = \left(1 + 4(\gamma + (\gamma_t/4 - \gamma)_+)\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \hat{\theta}_t)\right)\nabla\ell_t(\hat{\theta}_t)\,,$$

we can write

$$\begin{aligned}&\left(\nabla\hat{\ell}_t(\theta_t(\gamma);\gamma)(\theta_t(\gamma) - \theta^*)\right)^2 \\ &\quad= \left(1 + 4(\gamma + (\gamma_t/4 - \gamma)_+)\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \hat{\theta}_t)\right)^2 \left(\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*)\right)^2,\end{aligned}$$

which yields to

$$\hat{\gamma}_t \left(\nabla\hat{\ell}_t(\theta_t(\gamma);\gamma)(\theta_t(\gamma) - \theta^*)\right)^2 = 4(\gamma + (\gamma_t/4 - \gamma)_+) \left(\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*)\right)^2. \tag{B.7}$$

Using the assumption $4(\gamma + (\gamma_t/4 - \gamma)_+) \leqslant 1/GD$ we can also get

$$\left(\nabla\hat{\ell}_t(\theta_t(\gamma);\gamma)(\theta_t(\gamma) - \theta^*)\right)^2 \leqslant 4\left(\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*)\right)^2. \tag{B.8}$$

Therefore, plugging (B.7) and (B.8) in (B.6) we get

$$\begin{aligned}\sum_{t=1}^{n} \hat{\ell}_t(\theta_t(\gamma);\gamma) - \hat{\ell}_t(\theta^*;\gamma) &\leqslant \frac{5d\log(n)}{\gamma} + \frac{4\gamma}{2}\sum_{t=1}^{n}\left(\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*)\right)^2 \\ &\quad - 2\sum_{t=1}^{n}(\gamma + (\gamma_t/4 - \gamma)_+)\left(\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*)\right)^2 \\ &= \frac{5d\log(n)}{\gamma} \\ &\quad - 2\sum_{t=1}^{n}(\gamma_t/4 - \gamma)_+\left(\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*)\right)^2.\end{aligned}$$

Thus, the regret bound becomes

$$\begin{aligned}&Regret_n \leqslant \frac{2\log(K) + 5d\log(n)}{\gamma} \\ &- 2\sum_{t=1}^{n}(\gamma_t/4 - \gamma)_+\left((\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*))^2 + (\nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma)))^2\right) \\ &+ \sum_{t=1}^{n}\frac{(4\gamma - \gamma_t)_+}{2}(\nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta^*))^2,\end{aligned}$$

and as $(4\gamma - \gamma_t)_+ = 4\gamma + 4(\gamma_t/4 - \gamma)_+ - \gamma_t$, we can regroup and get

$$\begin{aligned}
Regret_n \leqslant & \frac{2\log(K) + 5d\log(n)}{\gamma} \\
& + 2\sum_{t=1}^{n}(\gamma_t/4 - \gamma)_+\Big((\nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta^*))^2 - (\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*))^2 \\
& \qquad - (\nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta_t(\gamma)))^2\Big) \\
& + \sum_{t=1}^{n}\frac{4\gamma - \gamma_t}{2}(\nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta^*))^2 \\
= & \frac{2\log(K) + 5d\log(n)}{\gamma} \\
& - 4\sum_{t=1}^{n}(\gamma_t/4 - \gamma)_+\Big((\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \theta^*))(\nabla\ell_t(\hat{\theta}_t)(\theta_t(\gamma) - \hat{\theta}_t))\Big) \\
& + \sum_{t=1}^{n}\frac{4\gamma - \gamma_t}{2}(\nabla\ell_t(\hat{\theta}_t)(\hat{\theta}_t - \theta^*))^2
\end{aligned}$$

□

Annexe C

Appendix C

C1 Scoring Rules

C1.1 Concordance Index

The concordance index was introduced by Harrell et al. [60] and it is the most widely used performance metric for time-to-event analysis [123]. It measures the fraction of pairs of subjects that are correctly ordered within all the possible pairs that can be ordered. The highest (and best) value that can be obtained is 1, which means that there is a complete agreement between the order of the observed and predicted times. The lowest value that can be obtained is 0, which means that all the prediction pairs are ordered backward with respect to the observed times, while a value of 0.5 denotes a random model.

First, we take every pair in the test set such that the earlier observed time is not censored. Then, we consider only pairs (i, j) such that $i < j$ and we also eliminate the pairs for which the times are tied. Next, we define a score $C_{i,j}$ for each pair (i, j) such as $y_i \neq y_j$, equal to 1 if the subject with earlier time (between i and j) has higher predicted risk, equal to 0.5 if the risks are tied, or equal to 0 otherwise.

Finally, given a subset of the data $\mathcal{D}$ of size n, we compute the concordance index as follows :

$$CI(\hat{S}, \mathcal{D}) = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} C_{i,j},$$

where,

$$C_{i,j} = \begin{cases} 1 & \text{if } y_i < y_j \text{ and } \hat{R}(x_i) > \hat{R}(x_j) \\ 0.5 & \text{if } \hat{R}(x_i) = \hat{R}(x_j) \\ 0 & \text{otherwise,} \end{cases}$$

and $\mathcal{P} = \{(i, j) \in \mathcal{D} \times \mathcal{D} : i < j,\ y_i \neq y_j,\ \text{if } y_i < y_j, \text{ then } \delta_i = 1\}$ is the set of all eligible pairs. To calculate the concordance index, we use the version of scikit-survival library [106] in Python.

C1.2 Integrated Brier score

We consider an approach based on the estimates of the probability functions that will be used as predictions of the event status $\mathbb{1}\{T_i > t\}$. In this case, $\mathbb{1}\{T_i > t\}$ has to be compared with $\hat{S}(t|X_i)$, leading to the mean squared error (MSE) at time t :

$$MSE(\hat{S}, t) = \mathbb{E}[(\mathbb{1}\{T_i > t\} - \hat{S}(t|X_i))^2].$$

The Brier score, introduced initially to measure the inaccuracy of probabilistic weather forecast by Brier [16], is an estimator of the MSE. It is important to remark that the MSE cannot be directly computed from the dataset since we do not know the underlying distribution of T_i but only the realizations of Y_i. Let us define $S_C(t|X_i) = \mathbb{P}(C_i > t|X_i)$ the survival censoring distribution and the Brier score :

$$BS(\hat{S}, t, \mathcal{D}) = \frac{1}{n}\sum_{i=1}^{n} W_i(t)(\mathbb{1}\{y_i > t) - \hat{S}(t|x_i))^2,$$

where (x_i, y_i, δ_i) for $0 < i \leqslant n$ are points from $\mathcal{D}$, and W_i is defined for all t as :

$$W_i(t) = \frac{\delta_i \mathbb{1}\{y_i \leqslant t\}}{\hat{S}_C(y_i|x_i)} + \frac{\mathbb{1}\{y_i > t\}}{\hat{S}_C(t|x_i)}.$$

Gerds and Schumacher [47] proved that the Brier score is a consistent estimator for the mean square error when the estimation $\hat{S}_C$ of S_C is well specified. Let us notice that in our implementation of the score, we use a Kaplan-Meier [76] estimator for the survival censoring function $\hat{S}_C$, which does not depend on the covariates. This assumption is not always the case for the real censoring function S_C, and it can lead to misspecifications of the model (wrong hypothesis on the probability shape) and, thus, to an estimation bias [52].

Finally, we consider over this paper the integrated Brier score :

$$IBS(\hat{S}, \mathcal{D}) = \frac{1}{\tau}\int_0^{\tau} BS(\hat{S}, t, \mathcal{D})dt,$$

where τ is a user-specified time horizon. There exist diverse scoring rules for survival models based on $L1$-loss, logarithmic loss and 1-calibration in between others (see [56] and [52] for more details). Other approaches of the estimation of prediction errors and model misspecification can be found in [86]. We chose the concordance index and integrated Brier score because they measure different aspects of the models, ranking and calibration, allowing us to have a good understanding of the performance of the methods.

C2 Implemented Methods

C2.1 Cox Proportional Hazard (Cox PH)

Cox proportional hazard is a semi-parametric method proposed by Cox [24] with the objective of measuring the impact of each covariate/feature in the estimation of the survival probability

function. It models the hazard function as a general linear regression of the covariates and a non-parametric baseline function $\lambda_0(t)$ that depends only on time. Given a subject with a covariate vector $x = \{x^1, \ldots, x^d\}$, the hazard function is as follows :

$$h(t|x) = \lambda_0(t) \exp\left(\beta^T x\right),$$

where the parameter $\beta = (\beta_1, ..., \beta_d)$ is estimated by maximizing the likelihood. This model is semi-parametric in the sense that the baseline function $\lambda_0(t)$ does not need to be specified and it can be chosen differently for each unique time. Cox proportional hazard is one of the most often used methods in time-to-event analysis and has a wide range of applications [83], [88], [115]. We use the implementation from scikit-survival library [106], where a regularization parameter α for ridge regression penalty is used and whose default is equal to 0. The mortality risk prediction will be determined by the log hazard ratio $R(x) = \beta^T x$.

C2.2 Gradient Boosting Cox (GBC)

Gradient boosting Cox is a machine learning method that was first proposed by Breiman [14], developed by Friedman [42] and adapted to survival models by Ridgeway [112]. The main idea is to combine a series of base learners in an additive manner to obtain a strong overall model. The base learners implemented in this case will be regression trees fitted at each stage on the negative gradient of the loss function. This is an additive method in the sense that it is constructed sequentially in a step-by-step greedy way. We can define the overall function f as follows :

$$f(x) = \sum_{k=1}^{K} \rho_k \cdot g_k(x, \theta),$$

where g_k is used to denote the base learners and K is the number of learners. Therefore, the objective is to maximize the log-likelihood function of Cox's proportional hazard model by replacing the linear regression $\beta^T x$ with the additive function $f(x)$ such that we have the following expression for the hazard function :

$$h(t|x) = \lambda_0(t) \exp\left(f(x)\right).$$

We use the implementation of scikit-survival [106] where we find three parameters of our interest, the learning rate that shrinks the contribution of each tree and it is set as default by 0.1, the maximum depth that specifies the depth to which each tree will be built and that is set equal to 3 by default, and the minimum samples leaf that determines the number of samples required to be at a leaf node and its default is equal to 1. Similar as Cox proportional hazard the mortality risk prediction can be interpreted as the log hazard ratio $f(x)$.

C2.3 Random Survival Forest (RSF)

Random survival forest was proposed by Ishwaran et al. [72] as an adaptation for censored data of the random forest method introduced by Breiman et al. [15]. It is an ensemble of tree-based learners where each tree is built from a bootstrap sampling of the training set in order to reduce the correlation between the trees. Also, for each node, it only evaluates the split criterion for a random subset of features and thresholds. The quality of a split is measured by the log-rank splitting rule [13] and then predictions are formed by aggregating predictions of the individual trees.

We implemented random survival forest from scikit-survival [106] and we will consider three of its parameters. The first one is the maximum depth which is set as infinity, which means that the nodes are expanded until no further partitioning is possible. The second one is the maximum features number which indicates the maximum number of features to consider when looking for the best split ; this parameter is set as the number of data features. The last parameter is the minimum samples leaf which in this case the default value is 3. Here, the mortality risk is defined by the ensemble mortality (see [72] for details) which corresponds to the sum of the cumulative Hazard functions estimated by the forest.

C2.4 Weibull Accelerated Failure Time (Weibull AFT)

Weibull AFT is a parametric model that was named after Waloddi Weibull, who was the first to promote its usefulness, particularly in the domain of strength of materials [136]. Accelerated failure time models also assume that the effect of a covariate is to accelerate or decelerate the life course. Given the parameters ρ and λ, the survival function of the Weibull distribution can be given as :

$$S(t|x) = \exp\left(-\left(\frac{t}{\rho(x)}\right)^{\lambda}\right),$$

where we consider the scale parameter $\rho(x) = \exp\left(\beta_0 \cdot (\beta^T x)\right)$ and λ is the parameter that controls the concavity of the cumulative hazard, indicating acceleration or deceleration hazards. In this case, we implement Weibull AFT from lifelines library [29] and we consider a penalizer parameter and a $\ell 1$-ratio to adjust how much of the penalizer should be attributed to an $\ell 1$ penalty. Both of them are initially set as zero by default. Here, the mortality risk is defined by $\mathbb{E}[T_i|x_i]$.

C2.5 Aalen's Additive Fitter (Aalen)

Aalen's additive is a parametric method proposed by Aalen [1]. This model responds to the fact that not all the covariates effects must be proportional, which is different from the assumption of Cox proportional hazard, but some of them can have additive effects. Besides, Aalen's additive model allows the effects of the covariates to vary over time which is not always the case with the other methods. The hazard function, in this case, is given as follows :

$$h(t|x) = \beta_0(t) + \beta^T(t)x,$$

where $\beta(t)$ is an unknown parameter of dimension d that are estimated by a linear regression (see [2]). We consider only the penalizer coefficient, which attaches an $\ell 2$ penalizer to the size of the parameters during regression that improves the stability of the estimations and controls the high correlation of the features. This penalizer is set to zero by default. Similarly as Weibull AFT, the mortality risk is defined by $\mathbb{E}[T_i|x_i]$.

C2.6 DeepSurv

DeepSurv is a nonlinear version of the Cox proportional hazard method proposed by Katzman et al. [77]. DeepSurv allows the use of neural networks within the original design of Cox's and aims to offer more flexibility in terms of the structure of the model than Cox proportional hazard.

DeepSurv is a multi-layer perceptron that predicts the risk of failure. The output of the network $\hat{r}_\theta(x)$ is a single node that estimates the risk function. The loss function to minimize is the negative log-partial likelihood of the Cox proportional hazard method :

$$\ell(\theta) = -\frac{1}{L}\sum_{i:\delta_i=1}\left(\hat{r}_\theta(x_i) - \log\left(\sum_{j\in\mathcal{R}(T_i)} e^{\hat{r}_\theta(x_j)}\right)\right) + \lambda||\theta||^2,$$

where λ is a ℓ_2 regularization parameter and L is the number of uncensored subjects. The network weights that minimize the loss function can be estimated by a gradient descent algorithm [114]. We use the implementation from Pysurvival [41], where we can choose the structure of the multilayer perceptron by choosing the number of hidden units per layer. We will consider two fully connected hidden layers and, consequently, two parameters to be set. The default number of units is 60 for the first layer and 10 for the second. Let us note that the Pysurvival library is now outdated. We have forked the library directly from the GitHub repository [1] (`https://github.com/square/pysurvival`). Other implementations of DeepSurv are available in the PyCox library [82] and the R package survivalmodels [121].

1. Our forked version can be found at : `https://github.com/camferna/pysurvival-wsklearn`

Annexe D

Appendix D

D1 Score Comparison

D1.1 Metrics

Precision and recall : We study the relation between precision and recall, commonly used to evaluate the performance of classification models. We estimate the probability of leaving within a given time horizon $t = 9$ months using various survival methods. Consequently, for a range of probability thresholds, we determine whether individual i leaves in 9 months or not transforming the estimation of the survival curves into a classification model. We compare the outputs of the classification predictions with δ_i, which allows us to define :

$$precision = \frac{true\ positives}{true\ positives + false\ positives},$$

$$recall = \frac{true\ positives}{true\ positives + false\ negatives}.$$

We study how precision and recall evolve across the grid of thresholds.

ROC curve : Following the previous setting, we also use the receiver operating characteristic curve, which is a graphical plot that illustrates the variation of the recall (also known as true positive rate) while increasing the probability of a false prediction (also known as false positive rate). We define the false positive rate as :

$$FPR = \frac{false\ positives}{false\ positives + true\ negatives}.$$

To this end, we consider a grid of thresholds and we calculate the FPR and the recall for each point on the grid.

In order to be able to use precision and recall and ROC curves to measure the performance of the models, we first need to transform the predictions into a classification setting. However, this transformation may not always align with our analytical needs and can discard important information of the survival curves.

D1.2 Comparison

In this section, we compare the performance of the models using precision and recall, as well as ROC curves. This approach requires us to frame it as a classification task, where an individual either leaves within 9 months or stays. This setting is limiting as it does not allow for the simultaneous evaluation of other time horizons. Diverse approaches have been proposed in order to use ROC to evaluate the performance of survival models [67, 75] but they were out of the scope of our study. We consider this approach in order to align with industrial applications and to increase communication with other teams.

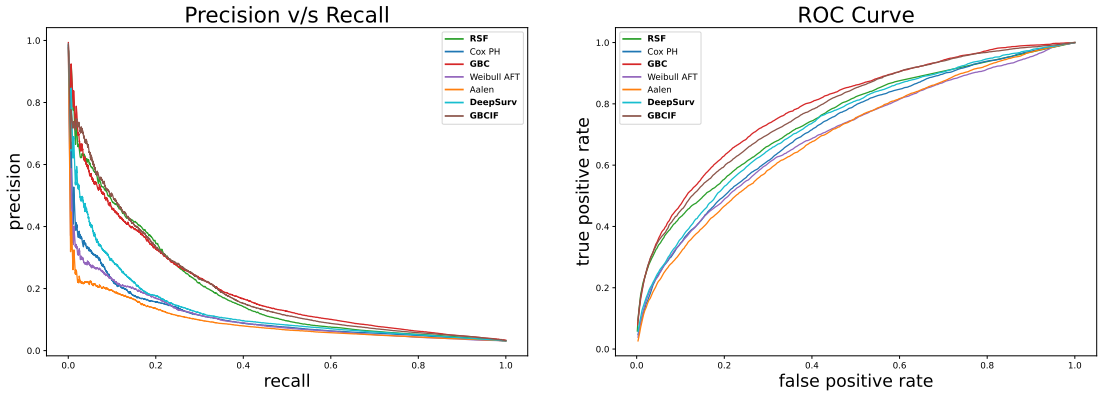


FIGURE D.1 – Precision v/s recall [left] and ROC curve [right] comparison on the attrition dataset.

Figure D.1 shows the average precision versus recall curves on the left, and the average ROC curves on the right. In the precision v/s recall graph we observe a consistent outperformance of Gradient Boosting (GBC), Gradient Boosting Cumulative Incidence Function (GBCIF), and Random Survival Forest (RSF), highlighting the superiority of machine learning approaches. The figure on the right supports this conclusion, and further, we notice the consistent outperformance of GBC. This result supports the conclusions of Section 4.3.

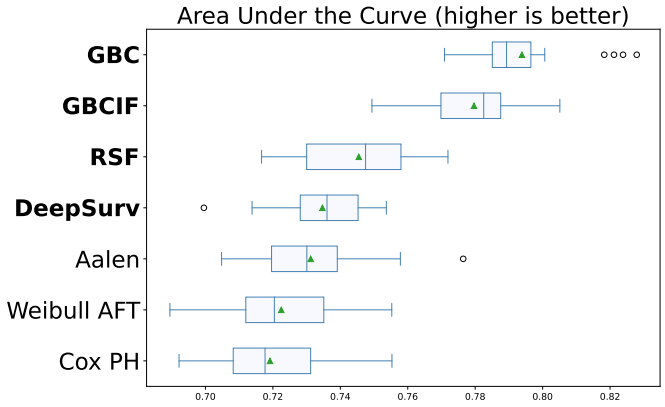


FIGURE D.2 – Box plot comparison across multiple dataset splits using the AUC value on the attrition dataset.

In Figure D.2 we compare performance using the AUC value (area under the ROC curve), where a higher AUC indicates better predictive accuracy. Machine learning methods consistently outperform others, with GBC showing the best performance across all metrics.

D2 Features Importance

D2.1 Cox proportional hazards

Following the study of Section 4.4, we present the result of permutation feature importance for Cox proportional hazards model.

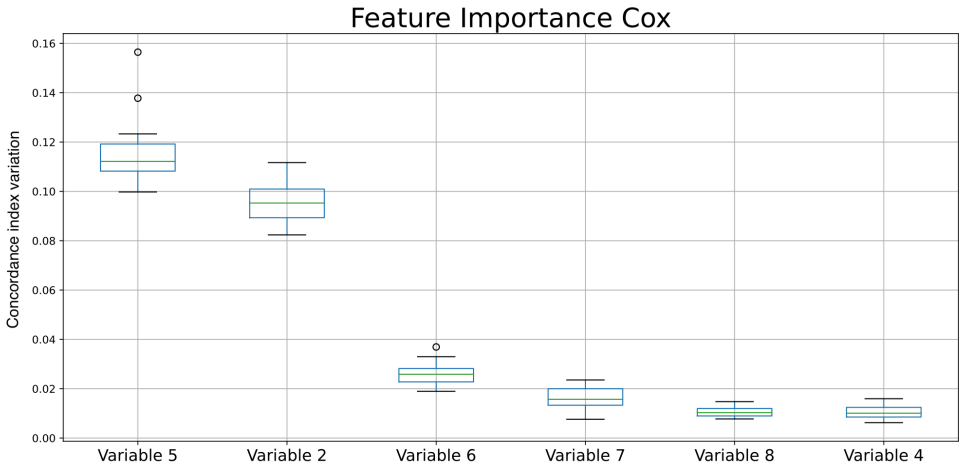


FIGURE D.3 – Permutation feature importance results of Cox PH model on the attrition dataset.

We observe in Figure D.3 that the most important feature is variable 5, followed by job variable 2 and variable 6. This outcome differs from the results presented in Section 4.4.1. Such differences can be attributed to the varying complexities with which each modeling method captures feature interactions, as well as their underlying assumptions about data distribution. We selected various thresholds to choose the features and compared the variance of the concordance index when training the model with different subsets of features

TABLEAU D.1 – Concordance index comparison of Cox PH model when selecting different subsets of features.

	53 features	27 features	14 features	10 features
Concordance index	0.782	0.782	0.784	0.773

In Tableau D.1, we observe that selecting the 14 most important features slightly increases the concordance index. This improvement may result from avoiding overfitting by excluding a large number of non-relevant features, reducing potential noise, and simplifying the model, which makes it easier to train and optimize. Additionally, this simplification allows for an enhanced focus on the relevant features.

In Figure D.4, we observe the impact of feature selection and hyperparameter optimization. For the Cox proportional hazards model, we consider only one hyperparameter, which is used for the ridge regression penalty regularization. Similar to the gradient boosting Cox model, we note a performance advantage when selecting features and optimizing the hyperparameters.

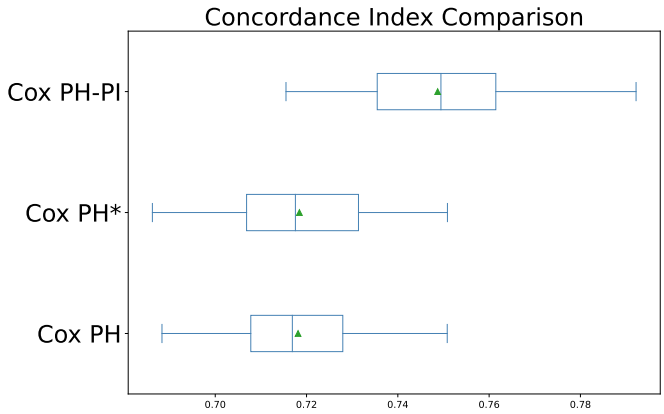


FIGURE D.4 – Box plot comparison across multiple dataset splits of the concordance index for the Cox PH model on the attrition dataset. We evaluate the impact of feature selection and hyperparameter optimization.

In Figure D.5, we present the beeswarm plot of the Cox proportional hazards model, which shows the Shapley values of different features. Similar to Figure D.3, variable 5 emerges as the most important feature for Cox PH model predictions, followed by variable 2. However, the third place is now occupied by variable 7, likely due to the differences in the methods used to calculate feature importance. Shapley values evaluate how each feature contributes by considering all possible combinations, reflecting the marginal effect of each feature within the context of others. Conversely, permutation feature importance, especially when measured using the concordance index, evaluates feature significance based on the impact on prediction performance when the feature values are shuffled, potentially overlooking the complex interactions between features.

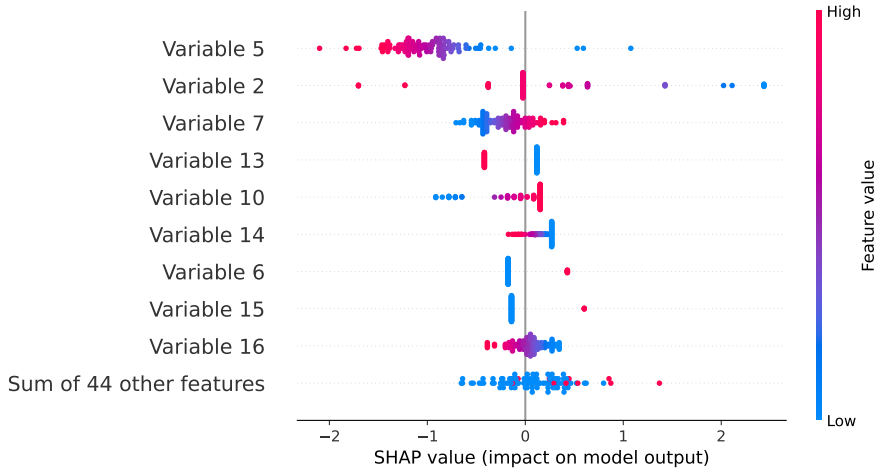


FIGURE D.5 – Feature importance beeswarm evaluated using Shapley values of Cox PH model on the attrition dataset.

Additionally, we observe that a high value of variable 5 negatively impacts model predictions, thereby decreasing the risk, which implies that employees who have higher values are more likely to stay. Similarly to the GBC model, a high value of variable 2 reduces the risk of leaving the company. Finally, a smaller value of variable 7 increases the probability of employees staying with the company.

D2.2 Random survival forest

In this section, we replicate the previous study using a random survival forest. Figure D.6 displays the results of the permutation feature importance, where we identify variable 5 as the most significant feature, followed by variable 2 and variable 1. The first two features align with the most relevant features identified by the Cox proportional hazards model. Consequently, similar to the previous section, we selected various subsets of features to examine the impact of feature selection on the concordance index.

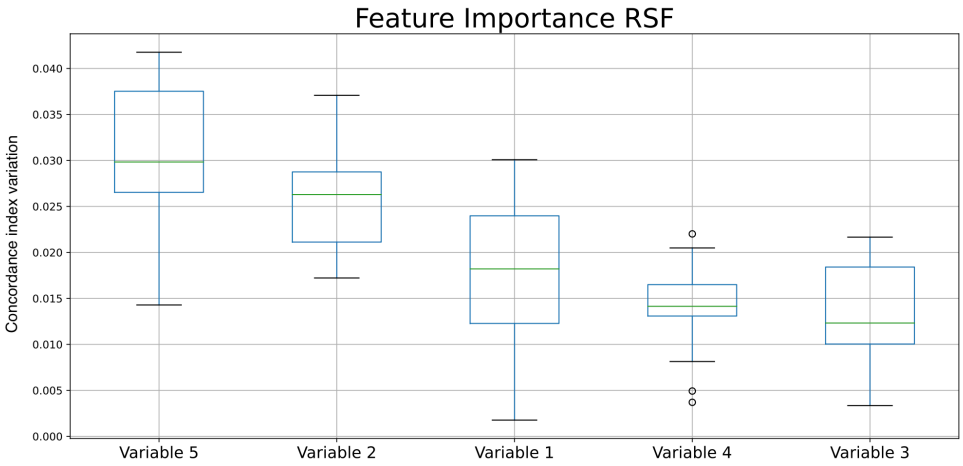


FIGURE D.6 – Permutation feature importance results of RSF model on the attrition dataset.

In Tableau D.2, we note that selecting the 27 most important features slightly improves the concordance index compared to training with the entire feature set.

TABLEAU D.2 – Concordance index comparison of RSF model when selecting different subsets of features.

	53 features	27 features	21 features	11 features
Concordance index	0.794	0.781	0.771	0.752

Finally, we evaluate the performance of the RSF model by optimizing the hyperparameters through randomized search and selecting the 27 most important features. In Figure D.7, similar to previous experiments, we observe that feature selection favors the optimization of hyperparameters and, consequently, the performance of the method.

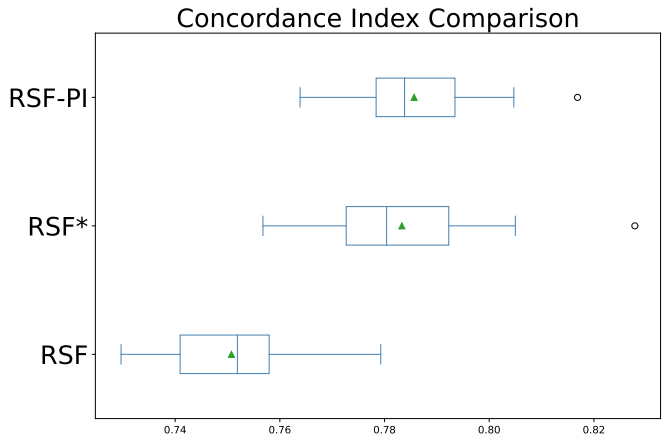


FIGURE D.7 – Box plot comparison across multiple dataset splits of the concordance index for the RSF model on the attrition dataset. We evaluate the impact of feature selection and hyperparameter optimization.

In Figure D.8, we present the beeswarm plot of the Random Survival Forest, which illustrates the Shapley values of the different features. We note that, consistent with Figure D.6, the three most significant features, variable 2, variable 5, and variable 1, maintain their importance lead, though their ranking has shifted.

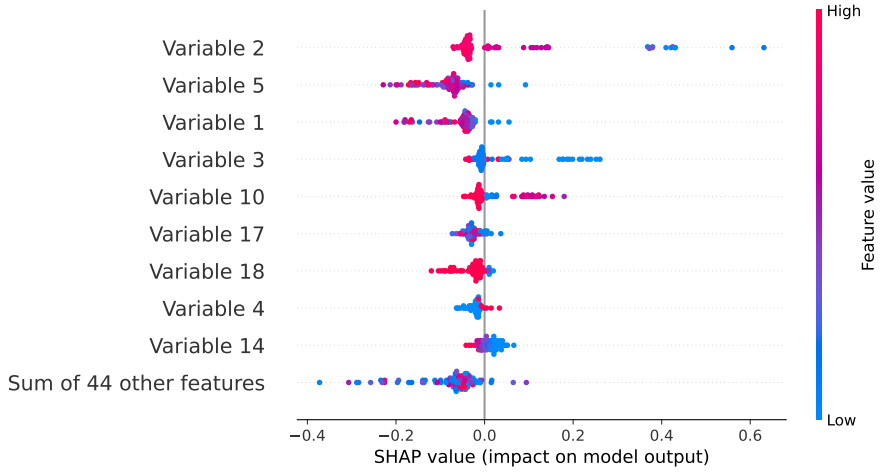


FIGURE D.8 – Feature importance beeswarm evaluated using Shapley values of RSF model on the attrition dataset.

Following the trends observed in Figure 4.5 and Figure D.5, we notice that a high value of variable 2 decreases the risk of leaving the company. Additionally, employees who have higher values of variable 5 are less likely to leave, similar to employees with higher values of variable 1 who are also less likely to depart.

In conclusion, important features vary across models, yet feature selection consistently im-

proves hyperparameter optimization and model performance. Additionally, the use of Shapley values reinforces the study of feature importance, allowing us to explicitly understand how each feature affects the model predictions.

Bibliographie

[1] O. O. Aalen. « A linear regression model for the analysis of life times ». In : *Statistics in Medicine* 8.8 (1989), p. 907-925.

[2] O. O. Aalen et T. H. Scheike. « Aalen's additive regression model ». In : *Encyclopedia of Biostatistics* 1 (2005).

[3] A. Agresti. *Categorical Data Analysis*. John Wiley & Sons, 2003.

[4] P. Ajit. « Prediction of employee turnover in organizations using machine learning algorithms ». In : *Algorithms* 4.5 (2016), p. C5.

[5] D. Alao et A. Adeyemo. « Analyzing employee attrition using decision tree algorithms ». In : *Computing, Information Systems, Development Informatics and Allied Research Journal* 4.1 (2013), p. 17-28.

[6] S. O. Arik et T. Pfister. « Tabnet : Attentive interpretable tabular learning. arXiv 2019 ». In : *arXiv preprint arXiv :1908.07442* (1908).

[7] S. O. Arik et T. Pfister. « Tabnet : Attentive interpretable tabular learning ». In : *arXiv* (2020).

[8] E. Arjas et P. Haara. « A logistic regression model for hazard : asymptotic results ». In : *Scandinavian Journal of Statistics* (1987), p. 1-18.

[9] N. Bennett, T. C. Blum, R. G. Long et P. M. Roman. « A firm-level analysis of employee attrition ». In : *Group & Organization Management* 18.4 (1993), p. 482-499.

[10] J. Berrisch et F. Ziel. « CRPS learning ». In : *Journal of Econometrics* (2021).

[11] H. Binder, A. Allignol, M. Schumacher et J. Beyersmann. « Boosting for high-dimensional time-to-event data with competing risks ». In : *Bioinformatics* 25.7 (2009), p. 890-896.

[12] H. Binder et M. Schumacher. « Allowing for mandatory covariates in boosting estimation of sparse high-dimensional survival models ». In : *BMC Bioinformatics* 9.1 (2008), p. 1-10.

[13] J. M. Bland et D. G. Altman. « The logrank test ». In : *British Medical Journal (BMJ)* 328.7447 (2004), p. 1073.

[14] L. Breiman. « Pasting bites together for prediction in large data sets and on-line ». In : *Univ. of Calif., Berkeley, Dept. of Statistics Technical Report* (1997).

[15] L. Breiman. « Random forests ». In : *Machine Learning* 45.1 (2001), p. 5-32.

[16] G. W. Brier et al. « Verification of forecasts expressed in terms of probability ». In : *Monthly Weather Review* 78.1 (1950), p. 1-3.

[17] S. L. Brilleman, R. Wolfe, M. Moreno-Betancur et M. J. Crowther. « Simulating survival data using the simsurv R package ». In : *Journal of Statistical Software* 97 (2021), p. 1-27.

[18] N. Cesa-Bianchi et G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[19] C. Chen et al. « Predictive maintenance using cox proportional hazard deep learning ». In : *Advanced Engineering Informatics* 44 (2020), p. 101054.

[20] G. Chen. « Nearest neighbor and kernel survival analysis : Nonasymptotic error bounds and strong consistency rates ». In : *International Conference on Machine Learning*. PMLR. 2019, p. 1001-1010.

[21] T. Chen et C. Guestrin. « Xgboost : A scalable tree boosting system ». In : *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, p. 785-794.

[22] Y. Chen, M. Hollander et N. Langberg. « Small-sample results for the Kaplan-Meier estimator ». In : *Journal of the American Statistical Association* 77.377 (1982), p. 141-144.

[23] B. Christoffersen. « dynamichazard : Dynamic hazard models using state space models ». In : *Journal of Statistical Software* 99 (2021), p. 1-38.

[24] D. R. Cox. « Regression models and life-tables ». In : *Journal of the Royal Statistical Society : Series B (Methodological)* 34.2 (1972), p. 187-202.

[25] D. R. Cox et D. Oakes. *Analysis of Survival Data*. T. 21. CRC press, 1984.

[26] S. J. Cutler et F. Ederer. « Maximum utilization of the life table method in analyzing survival ». In : *Journal of Chronic Diseases* 8.6 (1958), p. 699-712.

[27] A. Cwiling, V. Perduca et O. Bouaziz. « A comprehensive framework for evaluating time to event predictions using the restricted mean survival time ». In : *arXiv preprint arXiv :2306.16075* (2023).

[28] D. J. Daley et D. Vere-Jones. *An Introduction to the Theory of Point Processes. Vol. I*. Springer-Verlag, 2003.

[29] C. Davidson-Pilon. « Lifelines : survival analysis in Python ». In : *Journal of Open Source Software* 4.40 (2019), p. 1317.

[30] T. P. Debray, H. Koffijberg, D. Nieboer, Y. Vergouwe, E. W. Steyerberg et K. G. Moons. « Meta-analysis and aggregation of multiple published prediction models ». In : *Statistics in medicine* 33.14 (2014), p. 2341-2362.

[31] T. G. Dietterich. « Ensemble methods in machine learning ». In : *International Workshop on Multiple Classifier Systems*. Springer. 2000, p. 1-15.

[32] J. Duchi, E. Hazan et Y. Singer. « Adaptive subgradient methods for online learning and stochastic optimization. » In : *Journal of Machine Learning Research* 12.7 (2011).

[33] B. Efron. « The two sample problem with censored data ». In : *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. T. 4. 1967, p. 831-853.

[34] L. Fahrmeir. « Dynamic modelling and penalized likelihood estimation for discrete time survival data ». In : *Biometrika* 81.2 (1994), p. 317-330.

[35] L. Fahrmeir, W. Brachinger, A. Hamerle et G. Tutz. *Multivariate statistische verfahren*. Walter de Gruyter, 1996.

[36] D. Faraggi et R. Simon. « A neural network model for survival data ». In : *Statistics in Medicine* 14.1 (1995), p. 73-82.

[37] C. Fernandez, C. S. Chen, P. Gaillard et A. Silva. « Experimental Comparison of Ensemble Methods and Time-to-Event Analysis Models Through Integrated Brier Score and Concordance Index ». In : *arXiv preprint* (2024). eprint : arXiv:2403.07460.

[38] C. Fernandez, P. Gaillard, J. de Vilmarest et O. Wintenberger. « Online learning approach for survival analysis ». In : *arXiv preprint* (2024). eprint : arXiv:2402.05145.

[39] R. A. Fisher. « Two new properties of mathematical likelihood ». In : *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 144.852 (1934), p. 285-307.

[40] T. R. Fleming et D. P. Harrington. *Counting Processes and Survival Analysis*. T. 625. John Wiley & Sons, 2013.

[41] S. Fotso et al. *PySurvival : Open source package for Survival Analysis modeling*. 2019–.

[42] J. H. Friedman. « Greedy function approximation : a gradient boosting machine ». In : *Annals of Statistics* (2001), p. 1189-1232.

[43] N. Friedman, D. Geiger et M. Goldszmidt. « Bayesian network classifiers ». In : *Machine Learning* 29.2 (1997), p. 131-163.

[44] J. Frierson et D. Si. « Who's next : Evaluating attrition with machine learning algorithms and survival analysis ». In : *Big Data–BigData 2018 : 7th International Congress, Held as Part of the Services Conference Federation, SCF 2018, Seattle, WA, USA, June 25–30, 2018, Proceedings*. Springer. 2018, p. 251-259.

[45] A. Frye, C. Boomhower, M. Smith, L. Vitovsky et S. Fabricant. « Employee attrition : what makes an employee quit ? » In : *SMU Data Science Review* 1.1 (2018), p. 9.

[46] E. A. Gehan. « A generalized Wilcoxon test for comparing arbitrarily singly-censored samples ». In : *Biometrika* 52.1-2 (1965), p. 203-224.

[47] T. A. Gerds et M. Schumacher. « Consistent estimation of the expected Brier score in general survival models with right-censored event times ». In : *Biometrical Journal* 48.6 (2006), p. 1029-1040.

[48] D. Glass. « Graunt's life table ». In : *Journal of the Institute of Actuaries* 76.1 (1950), p. 60-64.

[49] M. K. Goel, P. Khanna et J. Kishore. « Understanding survival analysis : Kaplan-Meier estimate ». In : *International journal of Ayurveda Research* 1.4 (2010), p. 274.

[50] M. K. Golmakani et E. C. Polley. « Super learner for survival data prediction ». In : *The International Journal of Biostatistics* 16.2 (2020), p. 20190065.

[51] B. Gompertz. « XXIV. On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies. In a letter to Francis Baily, Esq. FRS &c ». In : *Philosophical Transactions of the Royal Society of London* (1825), p. 513-583.

[52] E. Graf, C. Schmoor, W. Sauerbrei et M. Schumacher. « Assessment and comparison of prognostic classification schemes for survival data ». In : *Statistics in Medicine* 18.17-18 (1999), p. 2529-2545.

[53] M. Greenwood. « The natural duration of cancer (report on public health and medical subjects no 33) ». In : *London : Stationery Office* (1926).

[54] O. Grisel et V. Maladiere. *Survival analysis benchmark*. `https://github.com/soda-inria/survival-analysis-benchmark/blob/main/notebooks/truck_dataset.ipynb`. 2023.

[55] F. Guerranti et G. M. Dimitri. « A Comparison of Machine Learning Approaches for Predicting Employee Attrition ». In : *Applied Sciences* 13.1 (2022), p. 267.

[56] H. Haider, B. Hoehn, S. Davis et R. Greiner. « Effective Ways to Build and Evaluate Individual Survival Distributions. » In : *Machine Learning Research* 21.85 (2020), p. 1-63.

[57] J. Hannan. « Approximation to Bayes risk in repeated play ». In : *Contributions to the Theory of Games* 3.2 (1957), p. 97-139.

[58] L. K. Hansen et P. Salamon. « Neural network ensembles ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12.10 (1990), p. 993-1001.

[59] J. J. Harden et J. Kropko. « Simulating duration data for the Cox model ». In : *Political Science Research and Methods* 7.4 (2019), p. 921-928.

[60] F. E. Harrell Jr, K. L. Lee et D. B. Mark. « Multivariable prognostic models : issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors ». In : *Statistics in Medicine* 15.4 (1996), p. 361-387.

[61] T. Hastie et C. Loader. « Local regression : Automatic kernel carpentry ». In : *Statistical Science* (1993), p. 120-129.

[62] T. Hastie, R. Tibshirani, J. H. Friedman et J. H. Friedman. *The Elements of Statistical Learning : Data Mining, Inference, and Prediction*. T. 2. Berlin, Germany : Springer, 2009.

[63] E. Hazan. *Introduction to Online Convex Optimization*. MIT Press, 2022.

[64] E. Hazan et al. « Introduction to online convex optimization ». In : *Foundations and Trends® in Optimization* 2.3-4 (2016), p. 157-325.

[65] E. Hazan, A. Agarwal et S. Kale. « Logarithmic regret algorithms for online convex optimization ». In : *Machine Learning* 69.2 (2007), p. 169-192.

[66] E. Hazan, T. Koren et K. Y. Levy. « Logistic regression : Tight bounds for stochastic and online optimization ». In : *Conference on Learning Theory*. PMLR. 2014, p. 197-209.

[67] P. J. Heagerty, T. Lumley et M. S. Pepe. « Time-dependent ROC curves for censored survival data and a diagnostic marker ». In : *Biometrics* 56.2 (2000), p. 337-344.

[68] F. Herzberg. *Motivation to work*. Routledge, 2017.

[69] W. Hoeffding. « Probability inequalities for sums of bounded random variables ». In : *The collected works of Wassily Hoeffding* (1994), p. 409-426.

[70] T. Hothorn, P. Bühlmann, S. Dudoit, A. Molinaro et M. J. Van Der Laan. « Survival ensembles ». In : *Biostatistics* 7.3 (2006), p. 355-373.

[71] IBM. « Kaggle Telco Customer Churn : » in : *IBM Cognos Analytics* ().

[72] H. Ishwaran, U. B. Kogalur, E. H. Blackstone et M. S. Lauer. « Random survival forests ». In : *The Annals of Applied Statistics* 2.3 (2008), p. 841-860.

[73] Z. Jin, J. Shang, Q. Zhu, C. Ling, W. Xie et B. Qiang. « RFRSF : Employee turnover prediction based on random forests and survival analysis ». In : *Web Information Systems Engineering–WISE 2020 : 21st International Conference, Amsterdam, The Netherlands, October 20–24, 2020, Proceedings, Part II 21*. Springer. 2020, p. 503-515.

[74] J. D. Kalbfleisch et R. L. Prentice. *The Statistical Analysis of Failure Time Data*. John Wiley & Sons, 2011.

[75] A. N. Kamarudin, T. Cox et R. Kolamunnage-Dona. « Time-dependent ROC curve analysis in medical research : current methods and applications ». In : *BMC medical research methodology* 17 (2017), p. 1-19.

[76] E. L. Kaplan et P. Meier. « Nonparametric estimation from incomplete observations ». In : *Journal of the American Statistical Association* 53.282 (1958), p. 457-481.

[77] J. L. Katzman, U. Shaham, A. Cloninger, J. Bates, T. Jiang et Y. Kluger. « DeepSurv : personalized treatment recommender system using a Cox proportional hazards deep neural network ». In : *BMC Medical Research Methodology* 18.1 (2018), p. 1-12.

[78] F. M. Khan et V. B. Zubek. « Support vector regression for censored data (SVRc) : a novel tool for survival analysis ». In : *IEEE International Conference on Data Mining*. IEEE. 2008, p. 863-868.

[79] D. P. Kingma et B. J. Adam. « A method for stochastic optimization ». In : *arXiv preprint arXiv :1412.6980* 1412 (2014).

[80] J. F. Kingman. *Poisson Processes*. T. 3. Clarendon Press, 1992.

[81] J. P. Klein, M. L. Moeschberger et al. *Survival Analysis : Techniques for Censored and Truncated Data*. T. 1230. Springer, 2003.

[82] H. Kvamme. *PyCox : PyTorch-based Survival Analysis Library*. `https://github.com/havakv/pycox`. 2019.

[83] W. R. Lane, S. W. Looney et J. W. Wansley. « An application of the Cox proportional hazards model to bank failure ». In : *Journal of Banking & Finance* 10.4 (1986), p. 511-531.

[84] D. Lang, U. Wittig-Berman et A. Rizkalla. « The Influences of Role Stress, Physical Symptoms, and Job Satisfaction on Turnover Intentions ». In : *Journal of Social Behavior and Personality* 7.4 (1992), p. 555.

[85] J. F. Lawless. *Statistical models and methods for lifetime data*. John Wiley & Sons, 2011.

[86] J. F. Lawless et Y. Yuan. « Estimation of prediction error for survival models ». In : *Statistics in Medicine* 29.2 (2010), p. 262-274.

[87] C. Lee, W. Zame, J. Yoon et M. Van Der Schaar. « Deephit : A deep learning approach to survival analysis with competing risks ». In : *Proceedings of the AAAI conference on artificial intelligence*. T. 32. 2018.

[88] K.-Y. Liang, S. G. Self et X. Liu. « The Cox proportional hazards model with change point : An epidemiologic application ». In : *Biometrics* (1990), p. 783-793.

[89] E. Liu, R. Y. Liu et K. Lim. « Using the Weibull accelerated failure time regression model to predict time to health events ». In : *Applied Sciences* 13.24 (2023), p. 13041.

[90] C. Loader. *Local regression and likelihood*. Springer, 1999.

[91] J. Lu. « Predicting customer churn in the telecommunications industry : An application of survival analysis modeling using SAS ». In : *SAS User Group International (SUGI27) Online Proceedings* 114 (2002), p. 27.

[92] S. M. Lundberg et S.-I. Lee. « A Unified Approach to Interpreting Model Predictions ». In : *Advances in neural information processing systems*. 2017, p. 4765-4774.

[93] M. Lunn et D. McNeil. « Applying Cox regression to competing risks ». In : *Biometrics* (1995), p. 524-532.

[94] B. Ma, G. Yan, B. Chai et X. Hou. « XGBLC : an improved survival prediction model based on XGBoost ». In : *Bioinformatics* 38.2 (2022), p. 410-418.

[95] N. Mantel. « Chi-square tests with one degree of freedom ; extensions of the Mantel-Haenszel procedure ». In : *Journal of the American Statistical Association* 58.303 (1963), p. 690-700.

[96] N. Mantel et W. Haenszel. « Statistical aspects of the analysis of data from retrospective studies of disease ». In : *Journal of the National Cancer Institute* 22.4 (1959), p. 719-748.

[97] P. McCullagh et J. A. Nelder. *Generalized Linear Models*. Routledge, 2019.

[98] M. Mitzenmacher et E. Upfal. *Probability and Computing : Randomization and Probabilistic Techniques in Algorithms and Data Analysis*. Cambridge University Press, 2017.

[99] K. K. Mohbey. « Employee's attrition prediction using survival analysis and Cox proportional hazard model ». In : *International Journal of Digital Enterprise Technology* 2.1 (2022), p. 27-37.

[100] J. G. Morita, T. W. Lee et R. T. Mowday. « Introducing survival analysis to organizational researchers : A selected application to turnover research. » In : *Journal of Applied Psychology* 74.2 (1989), p. 280.

[101] H. Ongori. « A review of the literature on employee turnover ». In : *Academic Journals* (2007).

[102] S. Y. Park, J. E. Park, H. Kim et S. H. Park. « Review of statistical methods for evaluating the performance of survival or other time-to-event prediction models (from conventional to deep learning approaches) ». In : *Korean Journal of Radiology* 22.10 (2021), p. 1697.

[103] C. M. Parkes. « Accuracy of predictions of survival in later stages of cancer ». In : *Br Med J* 2.5804 (1972), p. 29-31.

[104] F. Pedregosa et al. « Scikit-learn : Machine Learning in Python ». In : *Journal of Machine Learning Research* 12 (2011), p. 2825-2830.

[105] A. V. Peterson Jr. « Expressing the Kaplan-Meier estimator as a function of empirical subsurvival functions ». In : *Journal of the American Statistical Association* 72.360a (1977), p. 854-858.

[106] S. Pölsterl. « scikit-survival : A Library for Time-to-Event Analysis Built on Top of scikit-learn. » In : *Machine Learning Research* 21.212 (2020), p. 1-6.

[107] S. Pölsterl, N. Navab et A. Katouzian. « Fast training of support vector machines for survival analysis ». In : *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2015, p. 243-259.

[108] S. Popov, S. Morozov et A. Babenko. « Neural oblivious decision ensembles for deep learning on tabular data ». In : *arXiv preprint arXiv :1909.06312* (2019).

[109] H. Putter, M. Fiocco et R. B. Geskus. « Tutorial in biostatistics : competing risks and multi-state models ». In : *Statistics in medicine* 26.11 (2007), p. 2389-2430.

[110] S.-A. Qi et al. « An effective meaningful way to evaluate survival models ». In : *arXiv preprint arXiv :2306.01196* (2023).

[111] R. Ranganath, A. Perotte, N. Elhadad et D. Blei. « Deep survival analysis ». In : *Machine Learning for Healthcare Conference*. PMLR. 2016, p. 101-114.

[112] G. Ridgeway. « The state of boosting ». In : *Computing Science and Statistics* (1999), p. 172-181.

[113] P. Royston. « The lognormal distribution as a model for survival time in cancer, with an emphasis on prognostic factors ». In : *Statistica Neerlandica* 55.1 (2001), p. 89-104.

[114] S. Ruder. « An overview of gradient descent optimization algorithms ». In : *arXiv preprint arXiv :1609.04747* (2016).

[115] W. Sauerbrei et M. Schumacher. « A bootstrap resampling procedure for model building : application to the Cox regression model ». In : *Statistics in Medicine* 11.16 (1992), p. 2093-2109.

[116] R. E. Schapire. « The strength of weak learnability ». In : *Machine Learning* 5.2 (1990), p. 197-227.

[117] M. Schumacher et al. « Randomized 2×2 trial evaluating hormonal treatment and the duration of chemotherapy in node-positive breast cancer patients ». In : *German Breast Cancer Study Group. Journal of Clinical Oncology* 12.10 (1994), p. 2086-2093.

[118] S. Selvin. *Survival Analysis for Epidemiologic and Medical Research*. Cambridge University Press, 2008.

[119] L. S. Shapley et al. *A value for n-person games*. Princeton University Press Princeton, 1953.

[120] J. Sherman et W. J. Morrison. « Adjustment of an inverse matrix corresponding to a change in one element of a given matrix ». In : *The Annals of Mathematical Statistics* 21.1 (1950), p. 124-127.

[121] R. Sonabend et Y. Foucher. *survivalmodels : Models for Survival Analysis*. R package version 0.1.191. CRAN. 2024.

[122] R. E. B. Sonabend. « A theoretical and methodological framework for machine learning in survival analysis : Enabling transparent and accessible predictive modelling on right-censored time-to-event data ». Thèse de doct. UCL (University College London), 2021.

[123] H. Steck, B. Krishnapuram, C. Dehing-Oberije, P. Lambin et V. C. Raykar. « On ranking in survival analysis : Bounds on the concordance index ». In : *Advances in Neural Information Processing Systems*. 2008, p. 1209-1216.

[124] S. M. Stigler. « Gauss and the invention of least squares ». In : *The Annals of Statistics* (1981), p. 465-474.

[125] G. Therneau T Grambsch. « Modeling Survival Data : Extending the Cox Model ». In : *Springer-Verlag* (2000).

[126] T. M. Therneau et P. M. Grambsch. *Modeling Survival Data : Extending the Cox Model*. New York, NY, USA : Springer, 2000.

[127] J. Tobin. « Estimation of relationships for limited dependent variables ». In : *Econometrica : Journal of the Econometric Society* (1958), p. 24-36.

[128] G. Tutz. *Regression for Categorical Data*. T. 34. Cambridge University Press, 2011.

[129] G. Tutz et M. Schmid. *Modeling Discrete Time-to-Event Data*. Springer, 2016.

[130] H. Uno, T. Cai, M. J. Pencina, R. B. D'Agostino et L.-J. Wei. « On the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data ». In : *Statistics in Medicine* 30.10 (2011), p. 1105-1117.

[131] V. V. V'yugin et V. G. Trunov. « Online learning with continuous ranked probability score ». In : *Conformal and Probabilistic Prediction and Applications*. PMLR. 2019, p. 163-177.

[132] M. J. Van der Laan, E. C. Polley et A. E. Hubbard. « Super learner ». In : *Statistical applications in genetics and molecular biology* 6.1 (2007).

[133] T. Van Erven, W. M. Koolen et D. Van der Hoeven. « MetaGrad : Adaptation using multiple learning rates in online learning ». In : *arXiv preprint arXiv :2102.06622* (2021).

[134] W. N. Van Wieringen, D. Kun, R. Hampel et A.-L. Boulesteix. « Survival prediction using gene expression data : a review and comparison ». In : *Computational Statistics & Data Analysis* 53.5 (2009), p. 1590-1603.

[135] P. Wang, Y. Li et C. K. Reddy. « Machine learning for survival analysis : A survey ». In : *ACM Computing Surveys (CSUR)* 51.6 (2019), p. 1-36.

[136] W. Weibull. « A statistical theory of strength of materials ». In : *IVB-Handl.* (1939).

[137] S. Wiegrebe, P. Kopper, R. Sonabend, B. Bischl et A. Bender. « Deep learning for survival analysis : a review ». In : *Artificial Intelligence Review* 57.3 (2024), p. 65.

[138] O. Wintenberger. « Optimal learning with Bernstein online aggregation ». In : *Machine Learning* 106.1 (2017), p. 119-141.

[139] O. Wintenberger. « Stochastic Online Convex Optimization ; Application to probabilistic time series forecasting ». In : *arXiv preprint arXiv :2102.00729* (2021).

[140] T. J. Ypma. « Historical development of the Newton–Raphson method ». In : *SIAM Review* 37.4 (1995), p. 531-551.

[141] C. Zhang et Y. Ma. *Ensemble Machine Learning : Methods and Applications*. Berlin, Germany : Springer, 2012.

[142] L. Zhao et D. Feng. « Deep neural networks for survival analysis using pseudo values ». In : *IEEE journal of biomedical and health informatics* 24.11 (2020), p. 3308-3314.

[143] Z.-H. Zhou. *Ensemble Methods : Foundations and Algorithms*. London, UK : Chapman et Hall/CRC, 2019.

[144] J. Zietz, E. N. Zietz et G. S. Sirmans. « Determinants of house prices : a quantile regression approach ». In : *The Journal of Real Estate Finance and Economics* 37.4 (2008), p. 317-333.

[145] M. Zinkevich. « Online convex programming and generalized infinitesimal gradient ascent ». In : *Proceedings of the 20th international conference on machine learning (ICML-03)*. 2003, p. 928-936.

Contributions and Applications to Survival Analysis

Résumé

L'analyse de survie a suscité l'intérêt de diverses disciplines, allant de la médecine et de la maintenance prédictive à diverses applications industrielles. Sa popularité croissante peut être attribuée aux avancées significatives en matière de puissance de calcul et à la disponibilité accrue des données. Des approches variées ont été développées pour répondre au défi des données censurées, allant des outils statistiques classiques aux techniques contemporaines d'apprentissage automatique. Cependant, il reste encore une marge considérable pour l'amélioration. Cette thèse vise à introduire des approches innovantes qui fournissent des insights plus profonds sur les distributions de survie et à proposer de nouvelles méthodes avec des garanties théoriques qui améliorent la précision des prédictions.

Il est notamment remarquable de constater l'absence de modèles capables de traiter les données séquentielles, une configuration pertinente en raison de sa capacité à s'adapter rapidement à de nouvelles informations et de son efficacité à gérer de grands flux de données sans nécessiter d'importantes ressources mémoire. La première contribution de cette thèse est de proposer un cadre théorique pour la modélisation des données de survie en ligne. Nous modélisons la fonction de risque comme une exponentielle paramétrique qui dépend des covariables, et nous utilisons des algorithmes d'optimisation convexe en ligne pour optimiser la vraisemblance de notre modèle, une approche qui est novatrice dans ce domaine. Nous proposons un nouvel algorithme adaptatif de second ordre, SurvONS, qui assure une robustesse dans la sélection des hyperparamètres tout en maintenant des bornes de regret rapides. De plus, nous introduisons une approche stochastique qui améliore les propriétés de convexité pour atteindre des taux de convergence plus rapides.

La deuxième contribution de cette thèse est de fournir une comparaison détaillée de divers modèles de survie, incluant les modèles semi-paramétriques, paramétriques et ceux basés sur l'apprentissage automatique. Nous étudions les caractéristiques des ensembles de données qui influencent la performance des méthodes, et nous proposons une procédure d'agrégation qui améliore la précision et la robustesse des prédictions. Enfin, nous appliquons les différentes approches discutées tout au long de la thèse à une étude de cas industrielle : la prédiction de l'attrition des employés, un problème fondamental dans le monde des affaires moderne. De plus, nous étudions l'impact des caractéristiques des employés sur les prédictions d'attrition en utilisant l'importance des caractéristiques par permutation et les valeurs de Shapley.

Mots clés : analyse de survie, optimisation convexe en ligne, optimisation stochastique, apprentissage automatique, apprentissage en ligne

---

Laboratoire de Probabilités, Statistique et Modélisation
Sorbonne Université – Campus Pierre et Marie Curie – 4 place Jussieu – 75005 Paris – France

Abstract

Survival analysis has attracted interest from a wide range of disciplines, spanning from medicine and predictive maintenance to various industrial applications. Its growing popularity can be attributed to significant advancements in computational power and the increased availability of data. Diverse approaches have been developed to address the challenge of censored data, from classical statistical tools to contemporary machine learning techniques. However, there is still considerable room for improvement. This thesis aims to introduce innovative approaches that provide deeper insights into survival distributions and to propose new methods with theoretical guarantees that enhance prediction accuracy. Notably, we notice the lack of models able to treat sequential data, a setting that is relevant due to its ability to adapt quickly to new information and its efficiency in handling large data streams without requiring significant memory resources. The first contribution of this thesis is to propose a theoretical framework for modeling online survival data. We model the hazard function as a parametric exponential that depends on the covariates, and we use online convex optimization algorithms to minimize the negative log-likelihood of our model, an approach that is novel in this field. We propose a new adaptive second-order algorithm, SurvONS, which ensures robustness in hyperparameter selection while maintaining fast regret bounds. Additionally, we introduce a stochastic approach that enhances the convexity properties to achieve faster convergence rates.
The second contribution of this thesis is to provide a detailed comparison of diverse survival models, including semi-parametric, parametric, and machine learning models. We study the dataset characteristics that influence the methods performance, and we propose an aggregation procedure that enhances prediction accuracy and robustness. Finally, we apply the different approaches discussed throughout the thesis to an industrial case study: predicting employee attrition, a fundamental issue in modern business. Additionally, we study the impact of employee characteristics on attrition predictions using permutation feature importance and Shapley values.

Keywords: survival analysis, online convex optimization, stochastic optimization, machine learning, online learning

---