Active Learning : an unbiased approach

Carlos Eduardo Ribeiro de Mello

THÈSE

présentée par

Carlos Eduardo Ribeiro de Mello

pour l'obtention du grade de

DOCTEUR de l'ÉCOLE CENTRALE PARIS

Active Learning: An Unbiased Approach

préparée à l'*École Centrale Paris*, Laboratoire de Mathématiques Appliquées aux Systèmes, MAS, en cotutelle avec l'*Universidade Federal do Rio de Janeiro/COPPE*

Soutenue le 4 Juin 2013 devant le jury composé de:

Directeurs de thèse:

Geraldo Zimbrão da Silva – Professeur, Universidade Federal do Rio de Janeiro/COPPE

Carlos Eduardo Pedreira – Professeur, Universidade Federal do Rio de Janeiro/COPPE

Marie-Aude Aufaure – Professeur, École Centrale Paris

Président:

Nelson Maculan Filho – Professeur, Universidade Federal do Rio de Janeiro/COPPE

Rapporteurs:

Antônio de Pádua Braga – Professeur, Universidade Federal de Minas Gerais

Vincent Lemaire – HDR, Chercheur Senior, OrangeLabs

Examinateur:

Antoine Cornuéjols – Professeur, AgroParisTech/INA-PG

2013ECAP0036

To my family.

AGRADECIMENTOS

Agradeço a Deus por tudo!

Agradeço a Adria Lyra, meu amor, minha companheira, por estar sempre ao meu lado ao longo desta jornada, sendo paciente, companheira e me dando todo o apoio, mesmo durante aquelas temporadas em Paris. Certamente sua contribuição para o sucesso desta tese esta muito além da revisão do texto atenciosa a que se dispôs.

O meu muito obrigado à minha mãezinha querida, Isabel Cristina, por me trazer sempre aquela sua paz interior, e ao meu pai, meu orientador na vida e eterno herói, por sempre servir me de seus conselhos sempre valiosos. A minha "rimã", Mariana, meu agradecimento por ser meu exemplo de comprometimento e responsabilidade. À minha avozinha, Irene, meu agradecimento pelo incentivo nos estudos. Meu agradecimento à minha sobrinha, Giovana, por entender as ausências e por não trazer sempre a tia Adria para visita-la.

À minha família pelo amor, carinho e compreensão nos momentos em que fui ausente e por serem meu alicerce.

I thank my friends from Centrale, Rania Soussi, Etienne Cuvelier, Olivier Teboul, Cassio Melo, Nesrine Ben Mustapha, and Abhijeet Gaikwad, for their support and friendship along this work.

Aos meus amigos, Filipe Braida, Pedro Rougemont, Fellipe Duarte, Marden Pasinato e todos os meu colegas e funcionários da COPPE, meu sincero agradecimento.

Aos meus orientadores, Geraldo Zimbrão e Carlos Pedreira, pela orientação enriquecedora, pelo apoio às minhas propostas de trabalho e por sempre depositarem confiança em mim.

To my advisor, Marie-Aude Aufaure, for all support and guidance along this work, always open for my proposals and to trust on me.

Aos meus colegas da Universidade Federal Rural do Rio de Janeiro pelo apoio durante a realização desta tese.

I thank all members of the jury for agreeing to read the manuscript and for proving valuable remarks about this thesis.

Ao CNPq pelo suporte financeiro.

A todos o meu MUITO OBRIGADO!

THANK YOU ALL!

Resumo da Tese apresentada à COPPE/UFRJ e à *École Centrale Paris* como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

APRENDIZADO ATIVO: UMA ABORDAGEM NÃO-VIESADA

Carlos Eduardo Ribeiro de Mello

Junho/2013

Orientadores: Geraldo Zimbrão da Silva
Carlos Eduardo Pedreira
Marie-Aude Aufaure

Programa: Engenharia de Sistemas e Computação

Aprendizado Ativo surge como um importante tópico em diversos cenários de aprendizado supervisionado onde obter dados é barato, mas rotulá-los é custoso. Em geral, este consiste em uma estratégia de consulta, uma heurística gulosa baseada em algum critério de seleção, que busca pelas observações potencialmente mais informativas para serem rotuladas a fim de formar um conjunto de treinamento. Uma estratégia de consulta é portanto um procedimento de amostragem com viés, visto que esta favorece sistematicamente algumas observações, gerando um conjunto de treinamento enviesado, ao invés de realizar sorteios independentes e identicamente distribuídos. A principal hipótese desta tese recai na redução do viés oriundo do critério de seleção. A proposta principal consiste em reduzir o viés através da seleção de um conjunto mínimo de treinamento, a partir do qual a distribuição de probabilidade estimada será a mais próxima possível da distribuição do total de observações. Para tal, uma nova estratégia geral de consulta de aprendizado ativo foi desenvolvida utilizando um arcabouço de Teoria da Informação. Diversos experimentos foram realizados com o objetivo de avaliar o desempenho da estratégia proposta. Os resultados obtidos confirmam a hipótese sobre o viés, mostrando que a proposta é superior às estratégias de referência em diferentes conjuntos de dados.

Abstract of Thesis presented to COPPE/UFRJ and to *École Centrale Paris* as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

ACTIVE LEARNING: AN UNBIASED APPROACH

Carlos Eduardo Ribeiro de Mello

June/2013

Advisors: Geraldo Zimbrão da Silva
Carlos Eduardo Pedreira
Marie-Aude Aufaure

Department: Computer Science Engineering

Active Learning arises as an important issue in several supervised learning scenarios where obtaining data is cheap, but labeling is costly. In general, this consists in a query strategy, a greedy heuristic based on some selection criterion, which searches for the potentially most informative observations to be labeled in order to form a training set. A query strategy is therefore a biased sampling procedure since it systematically favors some observations by generating biased training sets, instead of making independent and identically distributed draws. The main hypothesis of this thesis lies in the reduction of the bias inherited from the selection criterion. The general proposal consists in reducing the bias by selecting the minimal training set from which the estimated probability distribution is as close as possible to the underlying distribution of overall observations. For that, a novel general active learning query strategy has been developed using an Information-Theoretic framework. Several experiments have been performed in order to evaluate the performance of the proposed strategy. The obtained results confirm the hypothesis about the bias, showing that the proposal outperforms the baselines in different datasets.

Résumé de la Thèse présentée devant la COPPE/UFRJ et l'École Centrale Paris comme travail partial nécessaire pour obtenir le grade de Docteur en Sciences (D.Sc.)

L'APPRENTISSAGE ACTIF: UNE APPROCHE NON BIAISÉE

Carlos Eduardo Ribeiro de Mello

Juin/2013

Directeurs: Geraldo Zimbrão da Silva
Carlos Eduardo Pedreira
Marie-Aude Aufaure

Département: Informatique

L'apprentissage actif apparaît comme un problème important dans différents contextes de l'apprentissage supervisé pour lesquels obtenir des données est une tâche aisée mais les étiqueter est coûteux. En règle générale, c'est une stratégie de requête, une heuristique gloutonne basée sur un critère de sélection qui recherche les données non étiquetées potentiellement les plus intéressantes pour former ainsi un ensemble d'apprentissage. Une stratégie de requête est donc une procédure d'échantillonnage biaisée puisqu'elle favorise systématiquement certaines observations s'écartant ainsi des modèles d'échantillonnages indépendants et identiquement distribués. L'hypothèse principale de cette thèse s'inscrit dans la réduction du biais introduit par le critère de sélection. La proposition générale consiste à réduire le biais en sélectionnant le sous-ensemble minimal d'apprentissage pour lequel l'estimation de la loi de probabilité est aussi proche que possible de la loi sous-jacente prenant en compte l'intégralité des observations. Pour ce faire, une nouvelle stratégie générale de requête pour l'apprentissage actif a été mise au point utilisant la théorie de l'Information. Les performances de la stratégie de requête proposée ont été évaluées sur des données réelles et simulées. Les résultats obtenus confirment l'hypothèse sur le biais et montrent que l'approche envisagée améliore l'état de l'art sur différents jeux de données.

Chapter 1 **Introduction** 1
1.1 Motivation 1
1.2 General Proposal 3
1.3 Contributions 4
1.4 Thesis organization 5
Chapter 2 **Active Learning: A Brief Literature Review** 7
2.1 Introduction 7
2.2 Illustrative Example 8
2.3 Active Learning 9
2.4 Properties of data 10
2.5 Scenarios 11
2.6 Problem Formulation 13
2.7 Query strategies 15
2.7.1 Uncertainty Sampling 16
2.7.2 Searching Through the Hypothesis Space 20
2.7.3 Minimizing Expected Error and Variance 24
2.7.4 Variance reduction 25
2.8 Exploiting Structure in Data 27
2.8.1 Density-weighted strategy 27
2.8.2 Cluster-based strategy 28
2.8.3 Semi-supervised learning 31
2.9 Additional considerations 32
2.9.1 Different Labeling Cost 32
2.9.2 Unbalanced classes 32
2.9.3 Noisy oracles 33
2.9.4 Stopping criteria 33
2.10 Summary and Conclusions 34
Chapter 3 **The Proposed Active Learning** 36
3.1 General Motivation 36
3.2 General Proposal 40
3.3 Proposal Formalization 41
3.4 Theoretical foundation 42
3.5 The proposed general query strategy 47
3.6 Kernel Density Estimation 49
3.7 Divergence Metrics 50

3.8 ISE-based Query Strategy 51
3.8.1 Integrated Squared Error of Kernel Density Estimation 52
3.8.2 Selecting a new observation 53
3.8.3 Geometry of ISE-based Query Strategy 57
3.8.4 Tuned implementation 60
3.9 Summary and Conclusions 63
Chapter 4 **Experiments 66**
4.1 Simulated Datasets 66
4.1.1 Simple datasets 66
4.1.2 Clustered datasets 67
4.1.3 Non-convex datasets 69
4.2 Qualitative Analysis 69
4.2.1 Experimental Setup 70
4.2.2 Selecting 10 observations 70
4.2.3 Selecting 20 observations 73
4.2.4 Selecting 100 observations 76
4.3 Quantitative Analysis 79
4.3.1 Experimental Setup 79
4.3.2 Results in the simulated datasets 83
4.3.3 Results in real datasets 86
4.4 Summary and Conclusions 87
Chapter 5 **Conclusion 90**
5.1 Summary and Discussion 90
5.2 Future work 91
Bibliography 93

Chapter 1 **Introduction**

1.1 Motivation

Since the last two decades, large amounts of data have been stored. These are daily generated by information systems, Internet, social networks, mobile applications and so on. These data may hide potential useful patterns for several applications such as information filtering, fraud detection, content recommendation, market segmentation, medical diagnosis aid, DNA sequence analysis, social network analysis, among others(HAN *et al.*, 2006).

In this context, Machine Learning and Data Mining provide frameworks, techniques, methods, and algorithms in order to allow pattern discovery and learning models from data(BISHOP, 2007, DUDA *et al.*, 2000, HAN *et al.*, 2006). For instance, clustering algorithms, such as *k-means* or *k-medoids*, are able to find underlying group structures in data(BISHOP, 2007, DUDA *et al.*, 2000, HAN *et al.*, 2006).

Unfortunately, there are data available that may not be completely useful for all Machine Learning tasks. Despite the abundance of data, the majority of these are not useful for an important category of learning algorithms, namely *supervised learning*. This class of learning algorithms requires annotated datasets, composed of labeled observations, *a.k.a.* training sets, for training their models.

For instance, although Internet is plenty of webpages about different categories of subject, very few of these are explicitly labeled with their correspondent categories. So, to build a webpage classifier according to such categories, it is required a training set containing a considerable number of webpages assigned to their correspondent categories, *i.e.*, a training set of labeled observations.

In this way, Active Learning arises as a Machine Learning field to tackle this labeling issue in scenarios where obtaining data is cheap, but labeling is very costly(SETTLES, 2012). For instance, in the case of the webpage classifier, a human annotator would label the webpages by reading and assigning them to their correspondent categories.

However, labeling these data incurs cost or time (often done by a human annotator), since a considerable number of categorized webpages are often required for training a classifier(BALDRIDGE & OSBORNE, 2004, SETTLES, 2012).

Active Learning aims at providing a framework of techniques for selecting the potentially most informative observations to be labeled so as to generate a training set. Based on this set, one should be able to train accurate supervised learning models. Therefore, an active learner (*i.e.*, active learning algorithm/system) should reduce the labeling cost, since only informative observations are queried, thereby avoiding unnecessary labeling costs(RUBENS *et al.*, 2011, SETTLES, 2012).

Several active learning algorithms, methods and techniques have been proposed in the literature along the past decade(RUBENS *et al.*, 2011, SETTLES, 2012). These mostly consists in query strategies that perform a greedy heuristic based on assumptions about the data distribution and the supervised learning model(DASGUPTA, 2009). As a heuristic, this is only guaranteed to work if their assumptions hold in the played scenario. Otherwise, it may perform very badly, even worse than the average performance of the simple random sampling, *a.k.a.* passive learning.

From a sampling perspective, active learning is in fact a biased sampling procedure, in which observations are drawn according to the inherent probability distribution of the selection heuristic. This distribution is usually different from the underlying distribution of the population, as the selection usually favors observations according to some rule, instead of randomly choosing them.

Therefore, active learning conducts to the overexploitation of regions in the space of observations (the input space), resulting in biased training sets. In case the selection heuristic relies on some mistaken assumption about the data or the model, the generated training set may result in very poor supervised models. The active learner generates training sets that are both uninformative and unrepresentative of the population.

In this way, the philosophy behind active learning lies in intentionally introducing bias in the training set in an attempt to pick the most informative observations for improving the learning model, even taking the risk to select the wrong ones. The more biased the training set is, the higher the risk to fail. In other words, a training set becomes less representative of the population distribution as it gets more biased, thereby augmenting its chance to produce poor models.

1.2 General Proposal

This thesis concerns a novel general active learning query strategy, which relies on the selection of observations according to their representativeness of the population distribution. The main hypothesis behind this proposal is that unbiased or little biased training sets are more likely to generate accurate supervised learning models than biased ones.

The key idea of this proposal is to look for observations that are most representative of the population distribution by keeping the underlying sample distribution as close as possible to the underlying population distribution. In this way, the query strategy should be careful not only with the selected sample, but also with its estimated probability distribution.

A selection criterion is designed as a greedy heuristic to choose the observations that minimize the distance between the estimated probability distribution of sample (training set) and the underlying distribution of the population (pool of unlabeled observations). For that, an information-theoretic framework is used to handle the probability density estimation and the distance measure between probability density functions.

The theoretical foundation behind the proposed general query strategy lies in the reduction of the sample space related to the input variables (unlabeled observations). This increases the probability of obtaining accurate estimators for the input distribution. Consequently, the proposed strategy has theoretical lower bounds of performance, which are superior to the simple random sampling, a.k.a. passive learning. In other words, the proposed active learning is more likely to provide more accurate models than passive learning.

A query strategy based on the proposed general strategy is developed, namely ISE-based query strategy. This implements the general proposal by taking the Integrated Squared Error(BISHOP, 2007) as the distance measure between probability density functions. This measure leads to an analytical expression as a selection criterion of observations in a straightforward fashion. As a consequence, one is able to design a polynomial time active learning algorithm on the number of selected observations.

Several experiments in both simulated and real datasets were done in order to evaluate the performance of the proposed query strategy. The simulated datasets exploit different properties, which allows the evaluation of the query strategy behavior faced on such

properties. The experiments provide both qualitative and quantitative analyses in order to understand the results deeply. Moreover baselines were defined in order to establish a comparative analysis with empirical upper and lower bounds of performance.

1.3 Contributions

The general contributions of this thesis are as follows:

- *General Active Learning Query Strategy*: this is the core of the thesis. This general query strategy allows the development of a whole new family of specific active learning query strategies based on the idea of reducing the bias in the training sets. This general strategy may be even used without handling probability distributions in a straightforward way, instead estimators of parametric functions may be considered.
- *Theoretical bounds of performance*: we provide a theoretical proof that, any sampling procedure that generates N-sized samples D_N, from a joint distribution $p(XY)$, so that the estimation error on its marginal $p(X)$ is minimized, also minimizes the estimation error on the joint distribution $p(XY)$. As we shall see in section 3.4, this assures that the proposed general query strategy has lower bounds of variance compared to the passive learning.
- *ISE-based Query Strategy*: this query strategy based on the proposed general strategy provides a very successful heuristic not only for active learning, but also for the general problem of sampling design. This heuristic may be applied to reduce the amount of data by maintaining the underlying distribution still representative in the sample.

The specific contributions of this thesis are as follows:

- *Model and label independent active learning*: as the proposed query strategy is only concerned with the input space, it requires neither the supervised model nor the labels. The independence of both model and labels avoids limitations such as time to re-training the model.
- *Sequential and batch mode*: as a consequence of the latter, the proposed query strategy works in either sequential or batch mode. As it is label independent, one is able to proceed with the selection strategy without knowing the real labels.

- *Noisy and Unbalanced label distributions*: Another consequence of label independence lies in the performance faced a large amount of noise in the label distribution. The proposed general query strategy is able to handle either noisy or unbalanced label distributions.

1.4 Thesis organization

This thesis is organized in 5 chapters, where this is the first.

Chapter 2 presents an overview of the state-of-the-art of Active Learning. The main active learning strategies are discussed along this chapter.

In chapter 3, the proposed general query strategy is described. This is the key chapter of this thesis, containing all developments and the methodology for the proposed query strategy and the ISE-based query strategy.

Chapter 4 presents the experimental setup used to evaluate and compare the performance of the proposal. The experiments were conducted with simulated and real datasets.

Finally, chapter 5 concludes the thesis with a discussion about the main achievements as well as the future work.

Chapter 2 Active Learning: A Brief Literature Review

In this chapter we present a literature review of Active Learning, providing a short overview of the state-of-the-art and the work related to this thesis. This chapter is inspired in (SETTLES, 2012), where readers can find further material on the subject.

2.1 Introduction

There are many ways of colleting data from different sources such as Internet, e-mails, and social networks, among others(HAN *et al.*, 2006). In order to discover information or patterns from these data, Machine Learning (ML) has been largely used for several purposes such as Natural Language Processing (NLP)(MANNING & SCHÜTZE, 1999), Speech Recognition(JURAFSKY & MARTIN, 2008), Handwritten Recognition(BISHOP, 2007), information filtering(RICCI *et al.*, 2010), goods recommendation(RICCI *et al.*, 2010) *etc*. These applications usually consist in classification, *i.e.*, a supervised learning algorithm that aims at learning a model (a classifier) from a training set of labeled data(DUDA *et al.*, 2000).

Generally, the majority of the available data is unlabeled, requiring to be labeled by an 'oracle' so as to be used for classification. Despite these unlabeled data can be relatively cheap to gather, labeling them might be costly and time-consuming in many scenarios. For instance, an oracle might be a human annotator with specific expertise on some domain for labeling each observation. Another example could be an experimental analysis as an oracle, which incurs in time and cost to obtain enough experimental results (labeled observations) to build an accurate classifier. In fact, there are several different sorts of settings in which the expense with labeling is very important.

In order to reduce this annotation cost, Active Learning arises providing a framework for selecting the most informative observations to be labeled, thereby generating training sets from which accurate classifiers might be learned.

A classic and simple example in (SETTLES, 2012, DASGUPTA, 2009) illustrates the

idea of active learning, as presented in section 2.2.

2.2 Illustrative Example

Suppose one has, initially, a dataset of unlabeled one-dimensional continuous observations (a.k.a. instances or points) $x \in \mathbb{R}$, each associated with a hidden label y coming from the set $Y = \{+, -\}$. Labels are assigned to observations according to a hidden threshold θ on x, which perfectly splits observations into two continuous sets: positive labels $(+)$ for $x > \theta$ and negative otherwise. Thus there is no noise in labels and hence this setting is linearly separable.

Let us define a classifier (a hypothesis) as a function mapping $h: X \rightarrow Y$, parameterized by an estimated threshold $\hat{\theta}$:

$$h(x; \hat{\theta}) = \begin{cases} + & if\ x > \hat{\theta}, \\ - & otherwise. \end{cases} \qquad \text{Eq. 1}$$

We aim at classifying observations as accurately as possible with $h(x; \hat{\theta})$ for all domain X by obtaining the estimated threshold $\hat{\theta}$, from the labeled data, as close as possible to the actual threshold θ.

A way to build this classifier might be by randomly drawing a large number n of unlabeled observations in order to subject them to the evaluation of an 'oracle' able to correctly label them. From these n labeled observations one learns a threshold $\hat{\theta}$ by seeking the cut point of label signal change alongside the axis X, as illustrated in Figure 2.1.

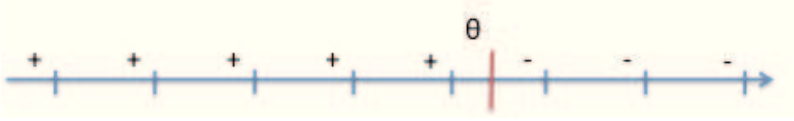

Figure 2.1

This scheme is known as *passive learning* and has been widely used in ML. Nevertheless, it turns out that obtaining accurate thresholds may get prohibitively

costly, since this scheme requires too many observations to be labeled.

As described in (DASGUPTA, 2009), according to the Probably Approximately Correct (PAC) learning framework (VALIANT, 1984), in a noiseless setting, the number of labeled observations in the training set to yield at most a generalization error ϵ has order $O(1/\epsilon)$. These observations are randomly drawn from the underlying data distribution(COHN *et al.*, 1994).

A desired scheme of selecting observations to be labeled should estimate a model with the same generalization error ϵ, but requiring fewer labeled observations. In the described example, a binary search could be performed so as to estimate the threshold θ(DASGUPTA, 2009). Thus, this would reduce the number of observations required for labeling to $O(\lceil \log_2 1/\epsilon \rceil)$, far better the passive learning(SETTLES, 2012, DASGUPTA, 2009).

2.3 Active Learning

As shown in the example presented in the previous section, the passive learning scheme draws independent and identically distributed observations (*iid*.) to be labeled by an oracle. In a linearly separable environment, the general accuracy improves linearly, as the number of labeled observations increases. In order to obtain an exponential improvement of accuracy, the learning algorithm (*i.e.* the learner) should actively query for label those observations that are potentially more informative, acting as an active learner(RUBENS *et al.*, 2011).

Generally speaking, Active Learning (a.k.a. Query Learning, Query Strategy, Active Query, or Optimal Experimental Design) aims at developing a framework to smartly select observations for training accurate supervised models(SETTLES, 2012). The main hypothesis of Active Learning concerns to learn accurate models from few informative selected observations, unlike passive learning, which draws observations at random.

Accordingly, the Active Learning process consists in deciding whether an observation should be labeled or not. This process might be in batch mode, in which a set of observations are queried for labels at once, or sequential, in which observations are queried for label sequentially(RUBENS *et al.*, 2011, DASGUPTA, 2009, SETTLES, 2012). In the latter, after labeling an observation, the training set is updated and the model is re-trained so as to select again another observation to be labeled. This process

proceeds until no more unlabeled observation is available or to reach some stopping criterion(VLACHOS, 2008).

There are several applications in which Active Learning would play an important hole, since unlabeled data are often abundant and labeled data are rare and expensive (or time-consuming) to obtain. For instance, we have classification and filtering of text documents (*e.g.* webpages, articles) or other sorts of medias (*e.g.* image, audio, and video files). These applications often require human annotation in order to build up training datasets for the supervised learning. In recommender systems, classifiers are built according to users' tastes on products. Then, the system should be wise when asking a user about its preferences and tastes, since users may get tedious and often answer very few queries(HARPALE & YANG, 2008, CARENINI *et al.*, 2003). In Computational Biology, for instance, learning a model to classify peptide chains needs a biology study carried out by a specialist so as to obtain labeled chains for training sets(BALDI & BRUNAK, 2001). Consequently, each peptide chain should be carefully chosen for analysis as it occurs cost and time.

2.4 Properties of data

Three interesting properties of observations should be taken into account by an active learner in order to maximize the selection effectiveness(RUBENS *et al.*, 2011). Figure 2.2 illustrates these properties, described as follows:

- ***Represented:*** when selecting a candidate observation to be labeled, one should consider if this observation is already represented in the input space, *i.e.* if this observation has already labeled observations in its neighborhood. The observation (b) in Figure 2.2 is an example of data already represented by the labeled observation of its group.
- ***Representative:*** this is complementary to the previous property; one should take into account how representative the candidate observation is, in the sense of how many unlabeled observations are represented by such candidate in the training set. For instance, the observations (d) and (c) in Figure 2.2 are good candidates to represent their groups.
- ***Results:*** one should care for the impact on the classification result in terms of accuracy or any other accomplishment with the addition of a labeled candidate

in the training set. The observation (a) in Figure 2.2, for instance, does not provide much information to help classify the other observations, since this is an outlier.

These properties provide a simple but useful way to identify the pros and cons of active learning strategies.

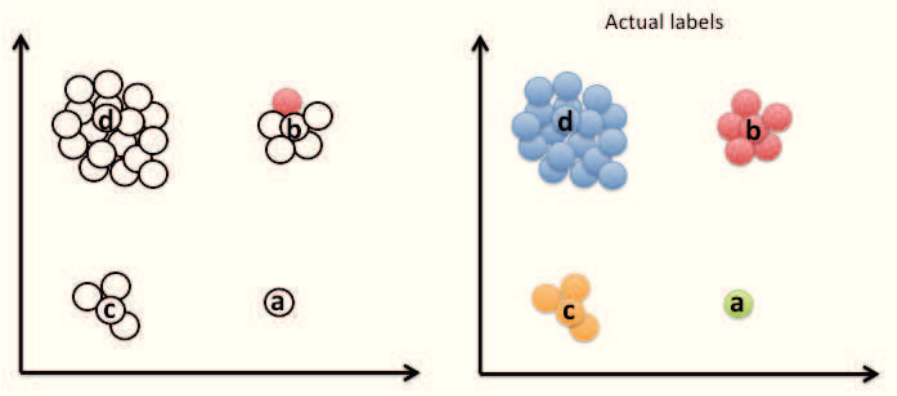

Figure 2.2

2.5 Scenarios

Active learning may not be worth in some scenarios, depending on the relative value of the labeling cost compared to the cost of developing and deploying the Active Learner (Active Learning system/algorithm). For instance, in a scenario in which to learn an accurate model requires a training set with few labeled observations, the cost of implementing and deploying an Active Learning system might be not worth. Also, there are scenarios where observations can be labeled almost 'for free' as the 'spam' flag in e-mail filtering or as in conversational systems, in which users receive some sort of incentive to label thousands of observations for free(CARENINI *et al.*, 2003).

Therefore, Active Learning usually suits properly scenarios where a large amount of data (unlabeled) is available and it is necessary to label a considerable number of these in order to learn an accurate supervised model. It is also generally assumed that there is an 'oracle' able to label any observation in these data, and the learning technique is previously defined, though these assumptions may not always hold.

In addition to these scenarios, there are many ways an Active Learner might query for labels. The three main query types considered in the literature are as follows(SETTLES, 2012):

- ***Query Synthesis:*** this is one of the first scenarios where Active Learning has been investigated. Here, active learner produces *membership queries* based only on the input space(ANGLUIN, 1988). Queries are synthesized *de novo*, generating never seen observations. Although this seems reasonable, it may not be feasible to get all these observations labeled. For instance, in image classification setting a synthetized query could be an awkward image, impossible to be labeled, though potentially informative for the learning model. Nevertheless, Query Synthesis still remains often required in settings where procedures are performed according to any possibility of query. For instance, in Biology studies, one synthetizes a substance from some possible combination of proteins by labeling it according to the presence of some target characteristic(KING *et al.*, 2004). The major issue related to this query type is that the underlying (unlabeled) data distribution cannot be exploited, which could be very informative for the learning model. The stream-based sampling and pool-based sampling tackle this problem.
- ***Stream-based selective sampling:*** this query type, unlike Query Synthesis, consists in a *selective sampling* in which unlabeled observations are drawn from the underlying distribution of the data(COHN *et al.*, 1994). When an unlabeled observation is sampled, the Active Learner decides whether to query or not for label such observation. This is also known as stream-based Active Learning as unlabeled observations are drawn one at a time as a stream. In case the underlying data distribution is uniform, Stream-based selective sampling may not present any advantage over Query Synthesis. Nevertheless, non-uniform distributions would provide important information to decide whether query or discard an observation. This decision is usually taken by using a utility function as a selection criterion. One may also define an uncertainty region, where only the observations that fall within it are queried. Most of the theoretical work in the literature concerns this query type as well as several real-world tasks have been used(SETTLES, 2012).

- ***Pool-based sampling:*** largely used in real applications, this query type selects observations to be labeled from a pool(LEWIS & GALE, 1994). Unlabeled data can be gathered without (or with low) cost by generating a pool of unlabeled observations from which Pool-based sampling selects which observations should be labeled. This usually proceeds in a sequential fashion, where each observation is sequentially picked from the pool, labeled by the oracle, and moved to the training set. The model is usually re-trained at each step. The observation selection is often carried out sorting observations in the pool according to a utility function. The main advantage of this query type is the possibility of exploiting the unlabeled data structure, which is often informative for the selection(DASGUPTA & HSU, 2008). Pool-based sampling is the most popular query type for applied research in Active Learning, whereas Query Synthesis and Stream-based selective sampling are mostly taken into account in theoretical works. This query type is assumed in our discussions in the remainder of this thesis.

2.6 Problem Formulation

As there are many different scenarios to be considered for Active Learning and many query types in the literature, in this section we define the active learning formulation of interest in this thesis.

We assume that unlabeled observations are abundant and that none, or very few, labeled observations are provided. We also consider that there is always an oracle able to correctly label any observation. Furthermore, we assume that the active learner queries observations in a Pool-based sampling way. We formalize these definitions as follows:

Let x be an unlabeled observation (or a data vector) from an input space X and y its correspondent label from a finite output space Y. Let $\mathcal{U}$ be a set of unlabeled observations, the pool. Let $\mathcal{L}$ be the training set of labeled observations defined by $\langle x, y \rangle \in \mathcal{L}$.

We define $f(x) = y$ as the target function and $h(x) = \hat{f}(x) = \hat{y}$ a model (hypothesis) learned from the set $\mathcal{L}$. The generalization error can be defined by the expectation:

$$\epsilon = E[\ell(h)] = \int_{-\infty}^{+\infty} \ell\big(h(x), f(x)\big)\, p(x), \qquad \text{Eq. 2}$$

where ℓ is a loss-function as the Squared Error(BISHOP, 2007):

$$\ell_{SE}\big(h(x), f(x)\big) = \big(h(x) - f(x)\big)^2. \qquad \text{Eq. 3}$$

The active learning problem consists in finding the smallest training set $\mathcal{L}$ such that the generalization error ϵ is minimized. In other words, for a fixed number of observations the active learner should find $\mathcal{L}$ that provides the minimum generalization error ϵ. When it is possible to obtain ϵ equals to zero, we say that the data is *separable*, otherwise *non-separable*.

One should note that the challenge lies in find $\mathcal{L}$ that produces the smallest ϵ by handling two important issues:

- The error ϵ cannot be directly computed since one does not know f at all, neither one has labeled enough observations to estimate a confident test error, otherwise one would use them to learn h.
- Even if one was able to get a good test error, it would still be difficult to choose observations for $\mathcal{L}$, since one would not know its actual label and estimating them would be very risky as long as the current model h could not be reliable.

In short, Active Learning algorithms consist in heuristics, which intend to establish a utility function for observations by selecting those with the highest utility. A utility function is mostly based on some theoretical selection criteria, which intend to exploit properties of observations that can be potentially informative for the supervised learning. In a Pool-based sampling setting, a selection criterion might take into account the current available data in the training $\mathcal{L}$, in the pool $\mathcal{U}$, as well as the current model h. In (DASGUPTA, 2009), a general active learning procedure is provided for the Pool-based sampling scenario as shown in Algorithm 2.1.

Algorithm General Active Learning Procedure

Input: $\mathcal{U}$ - pool of unlabeled observations

Output: $\mathcal{L}$ - training set of labeled observations

Randomly select few observations from $\mathcal{U}$

Repeat:

Query for labels the selected observations

Add the labeled observations into the training set $\mathcal{L}$

Train the model h from the training set $\mathcal{L}$

Select observations from $\mathcal{U}$ that minimize, or maximize, the utility $u(x)$

Algorithm 2.1 – General Active Learning Procedure

One should note that this greedy procedure selects observations to be labeled according to the utility function $u(x)$, which implements a selection criterion. There are different utility function proposals in the literature in order to produce better training sets by exploiting different properties of data. In the next subsections we present the main approaches of these query strategies.

2.7 Query strategies

There are several proposed active learning algorithms, also named query strategies or active learning heuristics, which aims to improve the training set by selecting observations according to some proposed selection criterion.

A recently work entitled 'Two faces of Active Learning' describes a useful analysis and organization of existent active learning strategies through out a theoretical standpoint(DASGUPTA, 2009). For that, two distinct narratives are provided: *efficient search through the hypothesis space* and *exploiting cluster structure in data*. The first concerns in shrinking the version space -consistent hypotheses with the training set- as fast as possible(MITCHELL, 1997). The second is based on the idea of using unsupervised learning like clustering in an attempt to select those most representative observations, thus avoiding a myopic view on the data.

Nevertheless, this two-fold organization does not embrace all heuristics, since many are empirical and lacks theoretical foundations. For this reason, in addition to these two categories, one still has *uncertainty sampling* and *minimizing expected error and*

variance. The former is concerned with the model and the latter with the direct minimization of the generalization error and the variance of the model output.

2.7.1 Uncertainty Sampling

Uncertainty sampling (also named *query by uncertainty*, *uncertainty reduction*, or *SIMPLE*) is probably the most popular active learning approach(LEWIS & CATLETT, 1994). This strategy exploits the uncertainty of the current hypothesis about its label prediction on a candidate observation. The idea is to ask the oracle to label those observations whose current hypothesis is least confident about. In this way, the strategy aims to avoid redundant observations for the model, as the observations with confident predictions are supposedly less informative.

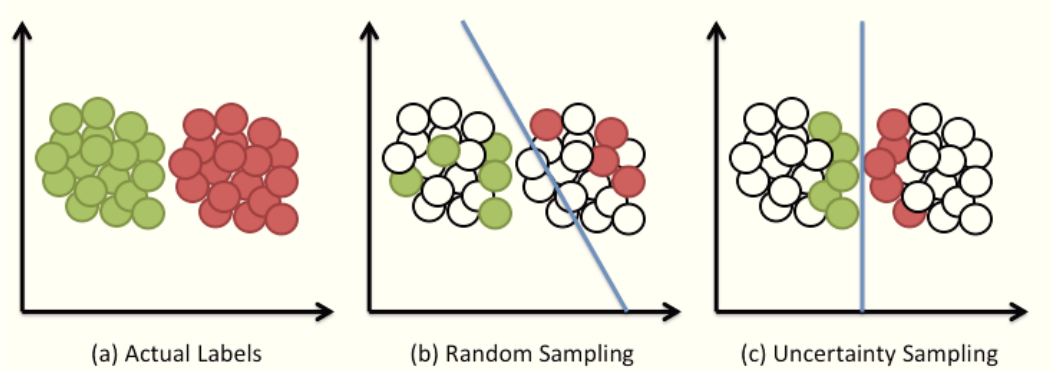

Figure 2.3

In Figure 2.3, a toy example illustrates the performance of an uncertainty sampling strategy compared to a simple random sampling in a binary classification task where 400 observations are represented in a 2D space as shown in (a). In (b), 30 observations are labeled by simple random sampling from the original data. A logistic regression model is trained and represented by the line. In (c), the uncertainty sampling strategy actively selected 30 observations by obtaining a more accurate hypothesis.

Uncertainty sampling is straightforward for probabilistic models, since these models provide probabilities associated with each label output. Thus, one can directly use the posterior distribution of a probabilistic model output $\Pr_\theta(Y|x)$, where θ is the set of current model parameters, as a measure of uncertainty for each observation x. For instance, in binary classification settings, the active learner should select the observation

whose $\Pr_\theta(\hat{y}|x)$ is closest to 0.5, where $\hat{y} = h(x)$ is the label output, *i.e.* $\hat{y} = \operatorname{argmax}_y \Pr_\theta(y|x)$.

Revisiting the illustrative example in subsection 2.2, the binary search used to guide the selection of observations to be labeled is in fact an uncertainty sampling strategy. In that example, the current hypothesis is given by the threshold θ. The closer the observation to θ, the less confident the classifier is. A measure to perform this uncertainty sampling strategy would be given by $|x - \theta|$. Therefore, the observation x with the smallest $|x - \theta|$ should be queried for a label. After the observation labeled and added in the training set, one updates the current hypothesis (the threshold θ), and the procedure re-starts.

In Algorithm 2.2, a generic uncertainty sampling strategy is described for the pool-based sampling scenario(SETTLES, 2012).

Algorithm Uncertainty Sampling Strategy

Input: $\mathcal{U}$ - pool of unlabeled observations

$\mathcal{L}$ - initial training set of labeled observations

Output: $\mathcal{L}$ - training set of labeled observations

For $t = 1,2$ ... **do**

$\theta = \text{train}(\mathcal{L})$

Select $x_* \in \mathcal{U}$, the most uncertainty observation according to model θ

Query the oracle to obtain label y_*

Add $\langle x_*; y_* \rangle$ to $\mathcal{L}$

Remove x_* from $\mathcal{U}$

End for

Algorithm 2.2 – Uncertainty Sampling Strategy

As one should obviously note, the key component of the uncertainty sampling is the measures of uncertainty. The design of such measures has been largely studied in order to handle multi-class classification and output structures as those for text classification(CULOTTA & MCCALLUM, 2005). Among the measures of uncertainty we highlight:

- *least confident* selects the observation x^* with the lowest output label probability: $x^* = \text{argmin}_x \Pr_\theta(\hat{y}|x)$;
- *margin* uses the difference of the probability between the two most likely label outputs so as to select the observation x^* with the smallest divergence between both classes: $x^* = \text{argmin}_x \Pr_\theta(\hat{y}_1|x) - \Pr_\theta(\hat{y}_2|x)$; and
- *entropy* selects the observation x^* whose the overall posterior output distribution of the current model has the highest entropy value, given by: $x^* = \text{argmax}_x - \sum_i \Pr_\theta(\hat{y}_i|x) \log \Pr_\theta(\hat{y}_i|x)$.

As one should note, all these measures lead to querying for the observations closest to the decision boundary, despite they may result in distinct performances since the rest of the probability space significantly differs one from another, as shown in Figure 2.4.

For instance, the entropy measure would select an observation whose label prediction is not the least confident, but, compared to the entire posterior output distribution, this observation is the most uncertain. Empirical results suggest that a measure may perform better than other depending to the played setting(KÖRNER & WROBEL, 2006, SCHEIN & UNGAR, 2007, SETTLES & CRAVEN, 2008).

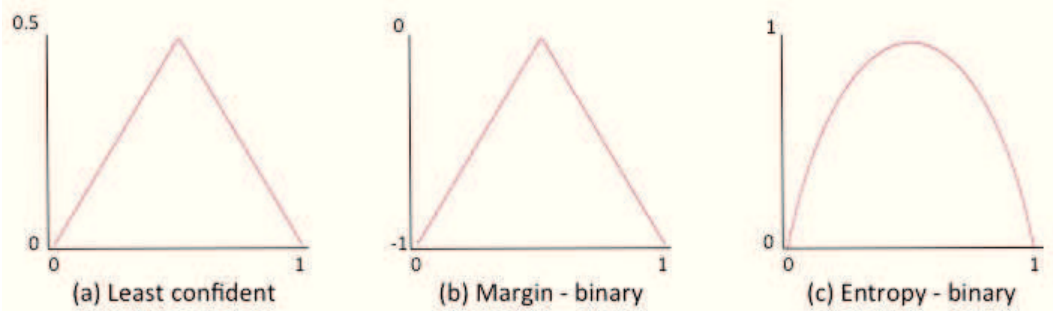

Figure 2.4

The main advantage of the Uncertainty sampling is its simplicity and the ability to handle different models as a 'black-box'. Even non-probabilistic models can be used with this heuristic, provided that they give any measure of uncertainty. For instance, in Nearest-Neighbors Classifiers(FUJII *et al.*, 1998), one may consider the proportion of sum of similarities between the labeled observations as an uncertainty measure. In Support Vector Machines(SCHOHN & COHN, 2000, TONG & KOLLER, 2002), one may select the observations according to their distance to the decision margin.

Uncertainty sampling strategies may also be appropriate not only for Pool-based sampling, but also for Stream-based selective sampling. A threshold of uncertainty should be set establishing a *region of uncertainty*. Those observations falling within this region should be queried. The concept of region of uncertainty is deeply exploited in *searching through the hypothesis space* strategies.

The main drawback of uncertainty sampling arises from the uncertainty scores based on the output of a single hypothesis. Hypotheses are often learned from very few observations, which may generate controversial uncertainty scores (see Figure 2.5). Thus unexploited regions in the input space might be overlooked, leading the strategy to a poor resulting training set. Moreover, uncertainty sampling is sensitive to outliers and noise (see Figure 2.6), conducting to an overexploitation of a noisy region, while other regions are overlooked. As this procedure introduces an inherent bias in the training set, it may even perform worse than a simple random sampling(WALLACE *et al.*, 2010).

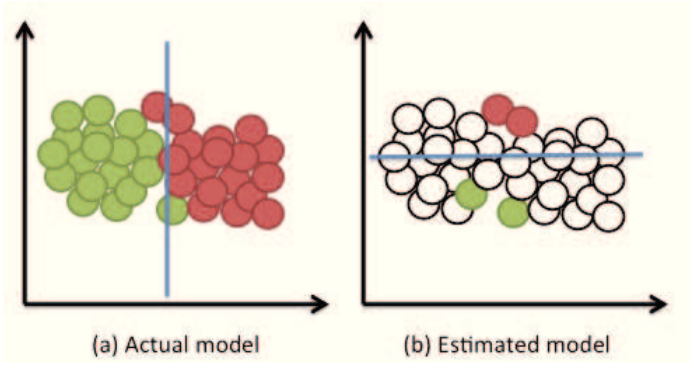

Figure 2.5

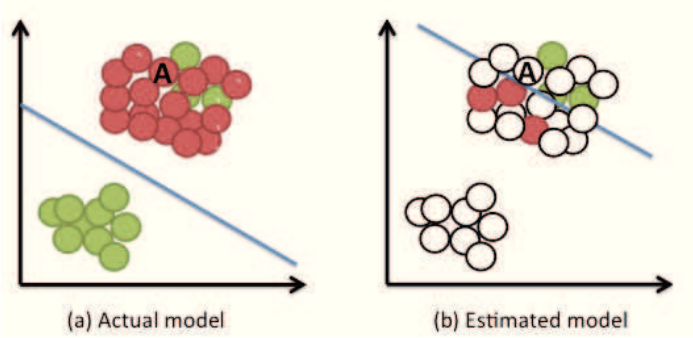

Figure 2.6

2.7.2 Searching Through the Hypothesis Space

Searching through the hypothesis space relies on the idea of selecting those observations that are good candidates to shrink the version space as fast as possible(DASGUPTA, 2009). Many theoretical results provide asymptotic upper bounds on the number of observations for this active learning framework(DASGUPTA, 2009, SETTLES, 2012).

A hypothesis $h(x)$ is a particular model learned from the training set $\mathcal{L}$ in order to generalize for unknown function $f(x)$. A learning algorithm aims at providing a hypothesis from a set of possible hypotheses $h \in \mathcal{H}$, called hypothesis space, in order to generalize the unknown function $f(x)$. By training a learning algorithm arises the version space, a subset of hypothesis $\mathcal{V} \subseteq \mathcal{H}$, which are *consistent* with the training set $\mathcal{L}$ from where the output hypothesis comes(MITCHELL, 1997). A hypothesis h is *consistent* if $h(x) = f(x)$ for all $x \in \mathcal{L}$. The shadow area represents the version space for a linear classifier in Figure 2.7. Therefore, the version space may be reduced when a new observation is added in the training set, since the hypotheses $h \in \mathcal{V}$ are subject not to be consistent with the new data.

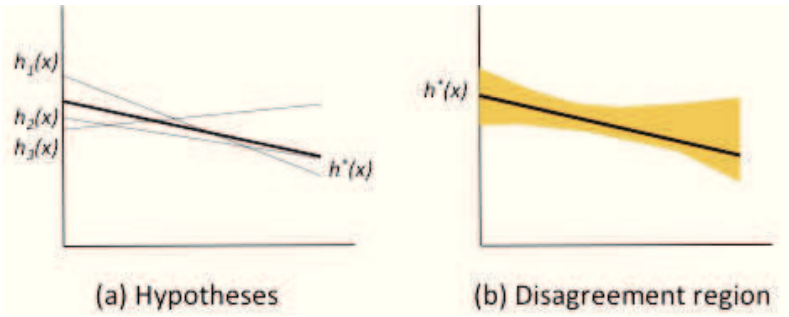

Figure 2.7

Region of disagreement

The definition of the version space arises the notion of *region of disagreement* in the input space, where any two hypotheses $h_1 \in \mathcal{V}$ and $h_2 \in \mathcal{V}$ disagree with the labeled prediction on an observation x, *i.e.* $DIS(\mathcal{V}) = \{x | \forall h_1, h_2, h_1(x) \neq h_2(x)\}$. Figure 2.8 illustrates an example for a square hypothesis model. Thus observations with positive labels should be inside the square and observations labeled as negative should be outside. The area of the difference between the most external square and the most internal defines the region of uncertainty $DIS(\mathcal{V})$.

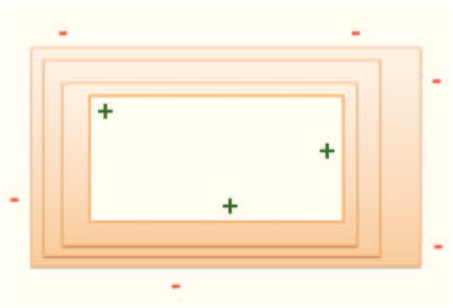

Figure 2.8

Relaxing the definition of consistent hypothesis for version space, one can also deal with noisy and non-separable data. One can assume a tolerance ε for error in the training set defined by $\Pr\big(h(x) \neq y(x)\big) \leq \varepsilon$ for all observations $x \in \mathcal{L}$.

Now that we have defined the version space, we are able to discuss about the approach *searching through the hypothesis space*. In fact, this approach aims at searching for observations, which reduces as fast as possible the version space. This goal emerges from the idea of reducing the probability of obtaining a hypothesis $h \in \mathcal{V}$ with high generalization error. As defined in the Probably Accurately Correct – PAC – learning framework(MITCHELL, 1997, VALIANT, 1984), for a low probability δ, a PAC classifier should satisfy

$$\Pr(error(h) > \epsilon | \, h \in \mathcal{V}) \leq \delta, \qquad \text{Eq. 4}$$

where $error(h) = \Pr\big(h(x) \neq f(x)\big), \forall x$ and ϵ is a given error threshold. Thus, reducing the version space as fast as possible guides the learning algorithm to obtain a hypothesis with low generalization error(TONG & KOLLER, 2002, DASGUPTA, 2009).

This theoretical framework is the basis of many works towards active learning algorithms aiming to reduce the version space. However, there are many practical difficulties in representing the version space, especially for higher dimensionality, non-separable and noisy settings.

Despite uncertainty sampling approach is conceptually different from searching observations to reduce the hypothesis space, for classifiers of max-margin such as

Support Vector Machines querying for observations closest to the margin can be equivalent to query for the observation, which is in the middle of the version space(SETTLES, 2012). In (TONG & KOLLER, 2002), the proposed active learner selects observations, which are closest to the margin. As long as the version space is symmetrical, the choice of observations closest to the margin bisects the version space. Hence, this active learning strategy works in fact as a binary search in the version space by exponentially reducing the label complexity upper bound on the number of observations. In addition, margin-based active learning has shown good results for practical applications(VIJAYANARASIMHAN *et al.*, 2013, SCHOHN & COHN, 2000, TONG & KOLLER, 2002).

Another active learning strategy that aims to shrink as fast as possible the version space is proposed in (COHN *et al.*, 1994), namely *Query by Disagreement*(QBD). This active learner was firstly proposed for Stream-based selective sampling. The idea is to only query for label those observations that fall within the region of disagreement $DIS(\mathcal{V})$. The algorithm is described in Algorithm 2.3.

Algorithm Query By Disagreement Strategy

Input: $\mathcal{U}$ - pool of unlabeled observations

$\mathcal{L}$ - initial training set of labeled observations

Output: $\mathcal{L}$ - training set of labeled observations

Set $\mathcal{V} \subseteq \mathcal{H}$ is the set of all "legal" hypotheses

For $t = 1,2$ ... **do**

Draw an observation $x \in \mathcal{U}$

If $h_1(x) \neq h_2(x)$ for $\forall h_1, h_2 \in \mathcal{V}$ **then**

Query label y for instance x

$\mathcal{L} = \mathcal{L} \cup \langle x; y \rangle$

$\mathcal{V} = \{h: h(x') = y', \forall \langle x'; y' \rangle \in \mathcal{L}\}$

Else

Do nothing; discard x

End if

End for

Algorithm 2.3 – Query By Disagreement Strategy

A major practical drawback of this algorithm is to maintain the current version space, as it may be infinite. In order to tackle this problem, one can keep the version space in an implicit way by training speculative consistent hypotheses h_i with $\mathcal{L} \cup \{\langle x, y_i \rangle\}$ for each possible $y_i \in Y$ (if one exists). Then, one should query for an observation x if $h_i(x) \neq h_j(x), \forall y_i, y_j \in Y$, otherwise do not. This alternative may be unpractical in case the training procedure of the learning algorithm is computationally expensive. Other alternative to tackle this problem, in case of binary classification, is to keep only the most general and the most specific hypothesis, $h_G(x)$ and $h_S(x)$. For that, the region of disagreement should be re-defined as $DIS(\mathcal{V}) = \{x | h_G(x) \neq h_S(x)\}$. The hypotheses $h_G(x)$ and $h_S(x)$ can be obtained by imputing artificial labels in the training set, creating conservative hypotheses for each class.

Another attempt to represent the version space in a feasible way is tackled by the strategy *Query by Committee* (QBC)(FREUND *et al.*, 1997). The idea consists in creating a committee of hypotheses drawn from the version space in order to evaluate whether an observation falls within a region of disagreement or not. Observations whose hypotheses of the committee highly disagree are queried, otherwise ignored. However, sampling hypotheses from the current version space is not feasible for many learning algorithms and even it would not work in noisy data. Alternatives to overcome this issue come from the idea of using ensemble methods such as bagging and boosting to build a committee of hypotheses(IYENGAR *et al.*, 2000, FREUND & SCHAPIRE, 1997, ABE & MAMITSUKA, 1998, MUSLEA *et al.*, 2000, MELVILLE & MOONEY, 2004). There are also several heuristics for measuring disagreement of hypotheses in the committee based on entropy(SHANNON, 2001), Kullback-Leiber divergence(KULLBACK & LEIBLER, 1951), and Jensen-Shannon divergence(MELVILLE *et al.*, 2005).

The QBC provides an alternative to search for observations that reduces the version space by keeping a committee of hypothesis, instead of the entire version space. However, the main drawback of QBC lies in the computational cost of training the committee each time a new observation is added in the training set. In addition, QBC can perform badly in the presence of outliers and noise.

2.7.3 Minimizing Expected Error and Variance

2.7.3.1 Expected Error Reduction

An active learning approach with strong theoretical foundation is based on the expected error reduction(ROY & MCCALLUM, 2001). Instead of reducing the version space, this active learning strategy aims to directly reduce the classification error. To perform such task, however, one would need to know which labels the oracle would set for each candidate observation as well as its future error produced by its correspondent updated hypotheses. In order to do that, the decision under uncertainty framework arises to estimate those values as expectations(BISHOP, 2007, SETTLES, 2012).

According to the statistical decision theory(BISHOP, 2007), the expected value is the less risky decision under uncertain one can take as this is a weighted sum over all possible outcomes $y_i \in Y$ and their correspondent probabilities(BISHOP, 2007, DUDA *et al.*, 2000). In this way, to compute the expected classification error, one needs the unknown probability distributions of both the output and the probability distribution of the future error. One reasonable solution for that is to use the posterior distribution of the model output as an approximation for those distributions. In addition, one might assume the unlabeled pool $\mathcal{U}$ available as representative of the underlying probability distribution of the data so as to estimate the test set.

Thus, the active learner should select the observation x^* that minimizes the expected classification or 0/1-loss by using the decision-theoretic measure as follows:

$$x^* = \min_x E_{Y|\theta,x}\left[\sum_{x' \in \mathcal{U}} E_{Y|\theta^+,x'}[y \neq \hat{y}]\right], \qquad \text{Eq. 5}$$

$$x^* = \min_x \sum_{y_i \in Y} \Pr_\theta(y_i|x)\left(\sum_{x' \in \mathcal{U}} 1 - \Pr_{\theta^+}(\hat{y}|x')\right), \qquad \text{Eq. 6}$$

where θ^+ is the model parameters after training with $\mathcal{L} \cup \{(x, y_i)\}$ and $E_{Y|\theta,x}$ is the expectation of Y conditioned on the current hypothesis θ and on the observation x. Other loss functions may be used, *e.g.* the log-loss:

$$x^* = \min_x \sum_{y_i \in Y} \Pr_\theta(y_i|x) \left(\sum_{x' \in \mathcal{U}} \left(- \sum_{y_j \in Y} \Pr_{\theta^+}(\hat{y}_j|x') \log \Pr_{\theta^+}(\hat{y}_j|x') \right) \right), \qquad \text{Eq. 7}$$

As one should note, this framework is intensively computationally costly, since for each candidate observation the model should be re-trained each time an observation is queried as well as computing the expected future error over the unlabeled pool for each query. To reduce this cost one may sample the unlabeled pool.

An optimistic variant of this active learner is based on using the most likely label according to the model, instead of the expected value(GUO & GREINER, 2007). Interestingly, it has shown good results besides reducing the computational cost.

The expected reduction approach is nearly optimal as it performs a myopic search to directly reduce the classification error. However, it may proceeds badly since noise and outliers are present in data. As its utility function is based on speculative hypotheses, the active learner might stick in certain regions of the input space by increasing the cost without information gain.

Although the error reduction strategy has a strong theoretical foundation since it directly reduces the error, it is very prohibitive to implement due to the need of re-training the model twice for each observation in the pool.

2.7.4 Variance reduction

An alternative strategy to directly reduce the expected error is to focus on reducing the output variance of the model. Unlikely the expected error, the output variance can be written in a closed-form for some models by avoiding the need of retraining (ZHANG & OLES, 2000, COHN *et al.*, 1996, 1994, COHN, 1996, MACKAY, 1992).

The key idea is to exploit the result provided in (GEMAN *et al.*, 1992), where the generalization error for the squared-loss function decomposes as follows:

$$E\left[\left(\hat{y}(x;\mathcal{D}) - y(x)\right)^2 \mid x\right] = E_{Y|x}[(y(x) - E[y|x])^2] + \left(E_{\mathcal{D}}[\hat{y}(x;\mathcal{D})] - E_{Y|x}[y|x]\right)^2 + E_{\mathcal{D}}[(\hat{y}(x;\mathcal{D}) - E_{\mathcal{D}}[\hat{y}(x;\mathcal{D})])^2]. \quad \text{Eq. 8}$$

The expectations $E_{Y|x}[\cdot]$ and $E_{\mathcal{D}}[\cdot]$ are over the distribution of $Y|x$ and the dataset distribution $\mathcal{D}$ and the expectation $E[\cdot]$ is over both. The first parcel in the right side is noise from the original data, the second is the model bias, and the last one is the variance. The Figure 2.9 illustrates the generalization error decomposition.

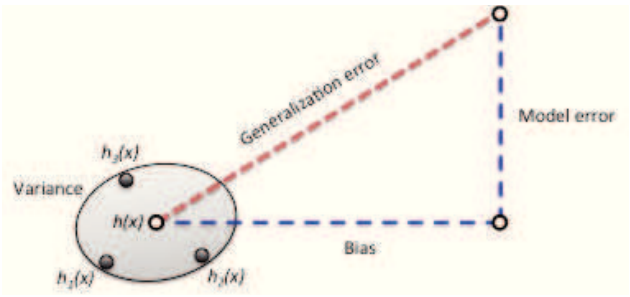

Figure 2.9

As one cannot reduce the parcel of noise and assuming that the model is unbiased, the error arises in the parcel corresponding to the output variance. In order to reduce the output variance, the well-known statistical framework of *optimal experimental design* provides a methodology to establish a closed-form for the output variance for many supervised learning models(COHN *et al.*, 1994).

Although Eq. 8 is concerned with the generalization error over both dataset distribution $\mathcal{D}$ and the unknown distribution $Y|x$, the proposals in the literature approach the variance reduction with $\mathcal{D}$ fixed, which may drive the model to overfitting(ZHANG & OLES, 2000, COHN *et al.*, 1996, 1994, COHN, 1996, MACKAY, 1992).

The main advantage of this strategy over the direct expected error reduction is to compute the expected future variance without knowing the actual labels of the candidate observations. In this way, it is not necessary retraining the model for each possible label in order to compute the expected variance(COHN *et al.*, 1994, 1996). As a consequence, variance reduction strategy allows query in batch as one may query for a fixed number of observations independently of their actual labels(MACKAY, 1992,

HOI *et al.*, 2006). In order to reach closed-forms for model variance, (ZHANG & OLES, 2000) uses the Fisher information and the Fisher score of the output variable conditioned by the training set distribution, which is related to the variance by the Cramer-Rao inequality(PRINCIPE, 2010).

The main drawback of the variance reduction strategy is the computational cost involved to compute the expected variances. Even with tuned implementations, it becomes prohibitive to compute these expectations for complex models and in high-dimensionality. Another important drawback is how to apply this strategy to non-statistical methods such as nearest-neighbors, decision trees and so on. As statisticians designed this strategy, most of its framework relies on statistical properties, which are not present in many models of Machine Learning(SETTLES, 2012). Furthermore, empirical results reported that this strategy framework has presented mixed performance among other active learners(SETTLES, 2012).

2.8 Exploiting Structure in Data

2.8.1 Density-weighted strategy

In the last sections, we described active learning strategies based on criteria that aim at reducing the model uncertainty, the version space, the expected error, and the output variance. However, none of these strategies explicitly take into account the data structure. Besides that, many of these strategies are sensible to outliers. For instance, in the uncertainty sampling strategy, unlabeled observations are myopic selected according to their closeness to the decision boundary, thus their representativeness is not considered in the strategy. Figure 2.10 shows a classic example where the observation *A* should be picked for labeling, as this is the closest to the decision boundary. However, labeling the observation *A* provides very poor information to the model, since it is an outlier.

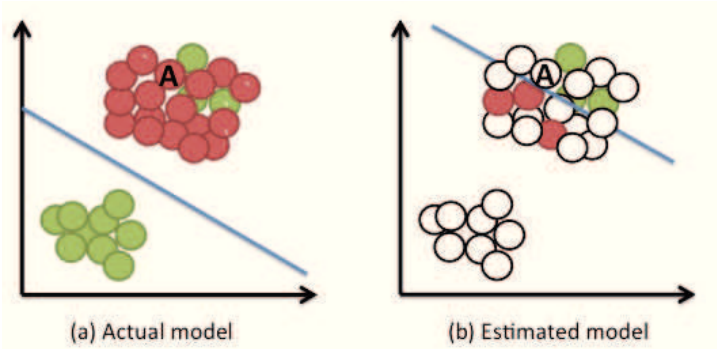

Figure 2.10

In order to take into account the data structure and input distribution while unlabeled observations are selected, *density-weighted strategies* were developed(SETTLES, 2012, XU *et al.*, 2007, FUJII *et al.*, 1998, MCCALLUM & NIGAM, 1998, SETTLES & CRAVEN, 2008, ROY & MCCALLUM, 2001). The key idea is to weight the utility function with the density of the input space. The general idea is given by

$$x^* = \max_x \phi(x) \left(\frac{1}{U} \sum_{u=1}^{U} sim(x, x_u) \right)^{\beta}, \qquad \text{Eq. 9}$$

where ϕ is an utility function like the uncertainty or the expected error reduction, and $sim(x, x_u)$ is a similarity measure between an observation x_u of the pool $\mathcal{U}$ and the candidate observation x. This formula can also be changed to consider the inverse of the similarity between the candidate observations and the observations in the training set. In this way, we could either consider the representative and the represented properties of the data (see subsection 2.4).

The main advantage of this strategy is its simple use. This can be easily combined with most of the strategies so far discussed in this chapter, incurring no greater increase in the computational cost, since one may cache the density weights(SETTLES & CRAVEN, 2008).

2.8.2 Cluster-based strategy

Another active learning strategy that aims to exploit the input data structure is the cluster-based active learner. In this strategy framework, clustering algorithms are used

to organize unlabeled observations of the pool into groups (clusters), so that the selection might be conducted according to their representativeness in the clusters.

This strategy assumes that both label and data structure are closely related such that clusters can provide rich information about which observations are supposedly more representative candidates, thus avoiding redundancy in the training set(DASGUPTA & HSU, 2008). Clustering has been also used for initially selecting the first observations to be labeled in the "warm starting" phase, *i.e.*, before starting with any active learning strategy(NGUYEN & SMEULDERS, 2004).

A hierarchical sampling is proposed based on hierarchical clustering structure in (DASGUPTA & HSU, 2008). The idea is to sequentially split clusters top down the clustering hierarchy as their labeled observations become heterogeneous within a cluster. Figure 2.11 illustrates the clustering cut: on the left side the second cluster becomes heterogeneous of labels. Then, this cluster is split out into two clusters with more homogeneity.

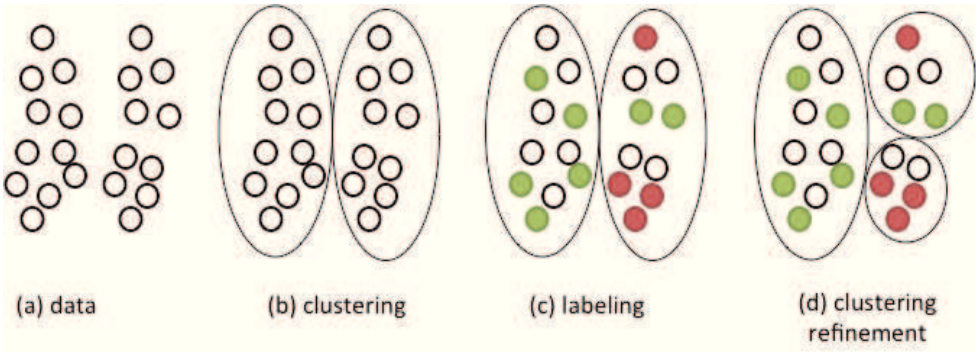

Figure 2.11

The hierarchical sampling is one of the few cluster-based strategies reported in the literature. Algorithm 2.4 describes this strategy procedure to select observations, where the function $\underline{cluster}$ returns a hierarchical tree of the cluster structure ready to be pruned and the function $\underline{select}$ selects the sub-tree $v \in \mathcal{P}$ in the hierarchy in which an observation should be queried for label. This latter function is essential for the procedure and basically two strategies are provided for that:

- selects the sub-tree v with probability proportional to its cardinality – similar to random sampling;

- selects the sub-tree v with probability proportional to its cardinality and variance of the labels inside – this strategy overexploits more controversial clusters in terms of their labels.

As the clusters get impure, in the sense of the labels inside becomes mixed, the algorithm prunes the tree by splitting out into clusters with more homogeneous labels. This process can be viewed as a clustering with constraints(HAN *et al.*, 2006). This procedure provides an asymptotic upper bound on the order of $O\left(\frac{|\mathcal{P}|\, d|\mathcal{P}|}{\epsilon}\right)$ for the labeled observations to reach a generalization error ϵ(DASGUPTA & HSU, 2008).

Although cluster-based strategies provide a different standpoint for Active Learning, its successful performance depends on whether exists a cluster structure in the data and these clusters are strongly related to the labels. When these assumptions hold in the data, this strategy outperforms random sampling and other active learning strategies(DASGUPTA & HSU, 2008, DASGUPTA, 2009).

Algorithm Hierarchical Sampling Strategy

Input: $\mathcal{U}$ - pool of unlabeled observations

$\mathcal{L}$ - initial training set of labeled observations

Output: $\mathcal{L}$ - training set of labeled observations

Do hierarchical clustering $T = cluster(\mathcal{U})$

Pruning $\mathcal{P} = \{root(T)\}$

For $t = 1,2$ ... **do**

Cluster node $v = select(\mathcal{P})$

Pick a random observation x from the subtree T_v and query its label

Update the label count for all cluster nodes u on a path from $x \rightarrow v$

Choose the best pruning $\mathcal{P}'_v$ and labeling L'_v for T_v

$\mathcal{P} = (\mathcal{P} - \{v\}) \cup \mathcal{P}'_v$

$L(u) = L'_v(u)$ for all $u \in \mathcal{P}'_v$

End for

For all $v \in \mathcal{P}$ **do**

Add $\langle x; L(v) \rangle$ to labeled set $\mathcal{L}$ for all $x \in \boldsymbol{T}_v$

End for

Algorithm 2.4 – Hierarchical Sampling Strategy

2.8.3 Semi-supervised learning

Another way of exploiting the information hidden in the unlabeled data is to use semi-supervised learning techniques such as *self-training*, *co-training*, and *multi-view learning*(CARLSON *et al.*, 2010, TOMANEK & HAHN, 2009, TUR *et al.*, 2005, ZHU *et al.*, 2003).

Semi-supervised learning and active learning share the same goal – build supervised models with fewer supervised data. However, semi-supervised learning struggles with this issue by automatically labeling observations, whereas active learning queries an oracle for more labels.

For instance, in semi-supervised learning like *self-training*, observations whose model is most certain about the prediction are labeled with that value, as long as an active learner as uncertainty sampling picks the most uncertainty observations to be labeled by

an oracle(TOMANEK & HAHN, 2009). In an active learning strategy as query-by-committee, observations with most disagreement of the committee should be queried for labels, whereas in semi-supervised learning the *co-training* technique labels those observations with high agreement within committee of models(MCCALLUM & NIGAM, 1998).

Consequently, active learning and semi-supervised learning are somehow philosophically complementary. Therefore, many active learning and semi-supervised learning strategies arise in the literature in an attempt to improve generalization accuracy in classification as much as possible(SETTLES, 2012, RUBENS *et al.*, 2011).

2.9 Additional considerations

In this section, we briefly point out some additional considerations for active learners such as different labeling cost, unbalanced label classes, noisy oracles, and stopping criteria.

2.9.1 Different Labeling Cost

None of the aforementioned active learning strategies approached in this chapter consider different labeling cost. However, in many scenarios one needs to take into account different costs when labeling observations. For instance, a certain experiment may be more expensive for some sort of setting than another, or there are users who are more likely to provide their personal information than another. So, in order to minimize the total labeling cost involved, the active learner should not only consider the potential information of a certain observation, but also the expense of labeling it. This motivates *cost-sensitive* active learning strategies(BALDRIDGE & OSBORNE, 2004, CULOTTA & MCCALLUM, 2005, KING *et al.*, 2004, KAPOOR *et al.*, 2007). These strategies take the label cost of each observation into consideration, providing a training set, which is most informative in a limited budget.

2.9.2 Unbalanced classes

In many applications and datasets, label classes might be very unbalanced such as fraud detection, spam filtering, and so on. A skewed label distribution introduces an especial difficulty for both the supervised learner and the active learner.

Skewed label distribution may lead the active learner to a performance no better than a random sampling, or even worse, since this usually performs biased sampling based on some assumptions about the label and model distribution. In order to struggle with this issue, (ATTENBERG & PROVOST, 2010, LOMASKY *et al.*, 2007) provides a guide strategy, namely *active class selection*, in which the oracle is queried for observations of a certain label, instead of labeling a given observation. This approach has been shown complementary to the active learning, offering a new perspective for the supervised data acquisition problem.

2.9.3 Noisy oracles

An important issue concerning the active learning setting is the reliability of the oracles. Active learning strategies are supposed to obtain 'correct' labels when queried an oracle. However, this may not hold in some scenarios where oracles are humans who might get bored, distracted or tired with the annotation task by mistakenly labeling observations. In order to avoid this issue, one may query different experts about the same observation(CARLSON *et al.*, 2010, MINTZ *et al.*, 2009, SNOW *et al.*, 2008). In this way, one hopes to produce gold-standard quality training sets by averaging out the different labels obtained from different experts (oracles). Other strategies consist in re-labeling so as to clean up the noise in labels(AMATRIAIN *et al.*, 2009).

2.9.4 Stopping criteria

Another key issue in the active learning process is how to stop. The generalization error decreases as the size of the training set rises. However after an exponential growth of accuracy, the learning algorithm reaches a plateau where no greater improvement on the generalization error happens, see Figure 2.12.

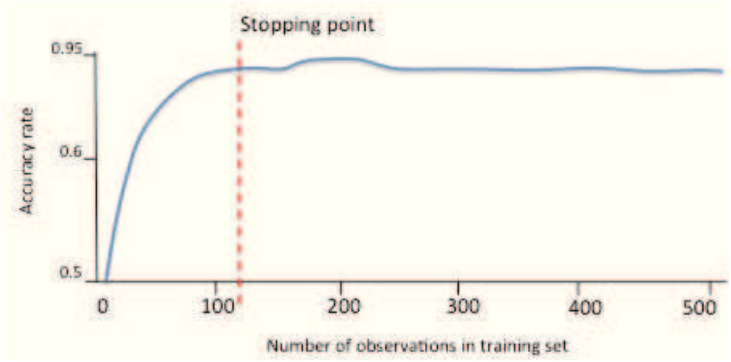

Figure 2.12

The idea of the stopping criterion is to provide a way of detecting when the learning algorithm has reached this stopping point. Several stopping criterion has been proposed in the literature(BLOODGOOD & VIJAY-SHANKER, 2009, OLSSON & TOMANEK, 2009, VLACHOS, 2008). Nevertheless, these are all very similar, usually based on some measure of model stability or confidence. As many of them are based on the model, their main drawback is to prematurely stop due to the active learning strategy gets stuck in a region of the input space.

2.10 Summary and Conclusions

In this chapter, a brief review of the state-of-the-art of Active Learning was presented. The application scenarios of active learning and the principle type of queries were described. The main active learning strategies based on different frameworks, assumptions and models were discussed by presenting their motivation and theoretical foundation. These are usually concerned with reducing or maximizing some selection criterion (a heuristic) such as the expected error, variance, version space, model uncertainty and so on. As a consequence, the active learner introduces a strong bias in the training set so as to avoid uninformative, redundant or noisy observations.

All active learning strategies relies on either the current model or label distribution to decide which observations should be labeled next. Therefore, all active learning strategies are subject to fail as the current model and the label distribution might be mistaken. Moreover, many active learners are computationally expensive and those, which are not, may have no theoretical guarantee of better performance even against random sampling.

Chapter 3 The Proposed Active Learning

In this chapter we present the methodology of this thesis, which provides support for the proposal of a novel general query strategy. The main contributions of this chapter are twofold: 1) the general active learning query strategy, which provides the theoretical foundation of this thesis as well as theoretical performance bounds over passive learning, and 2) a novel specific active learning query strategy and its tuned implementation based on the general proposal. We start with a general motivation, and follow, in more specific mode, with the novel active strategy and its developments.

3.1 General Motivation

The key idea behind the general active learning procedure is to perform a greedy query strategy, which selects, according to a specific criterion, the potentially 'most informative' unlabeled observations out of a pool(DASGUPTA, 2009). Once labeled, this ensemble of observations forms a training set from which supervised models can be learned. The goal is then to obtain training sets as informative as possible for supervised learning. Hence, a key ingredient and major concern for a successful query strategy is an appropriate choice of the selection criterion.

As detailed in chapter 2, there exist several works in the literature proposing different approaches for selection criteria. These are mostly heuristics that select observations according to a given utility function. This usually relies on assumptions (or hypotheses) about the played setting such as the data distribution and the current supervised model(SETTLES, 2012). The candidate observations with highest utility are selected for labeling.

For instance, the *uncertainty sampling* query strategy (for details see chapter 2) selects observations according to the model uncertainty on label predictions. The hypothesis behind this strategy relies on the idea that regions of the input space where the model output is most uncertain should be the most informative. Analogously, observations in

regions with confident model predictions might be supposed to be redundant for learning.

Figure 3.1 illustrates an example in which the *uncertainty sampling* selects the observations closest to the model decision boundary (b), reaching a accurate model (c).

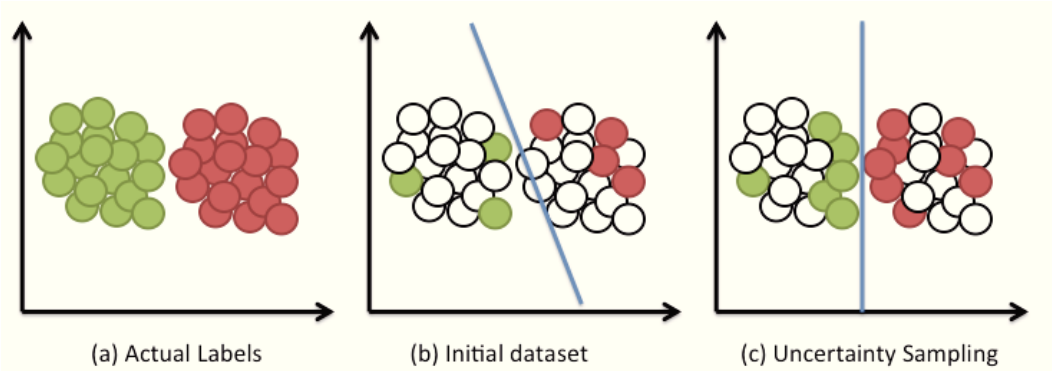

Figure 3.1

Note that the *uncertainty sampling* strategy strongly depends on the current supervised model. Thus, an inappropriate model may lead the *uncertainty sampling* to selecting uninformative observations, as depicted in Figure 3.2.

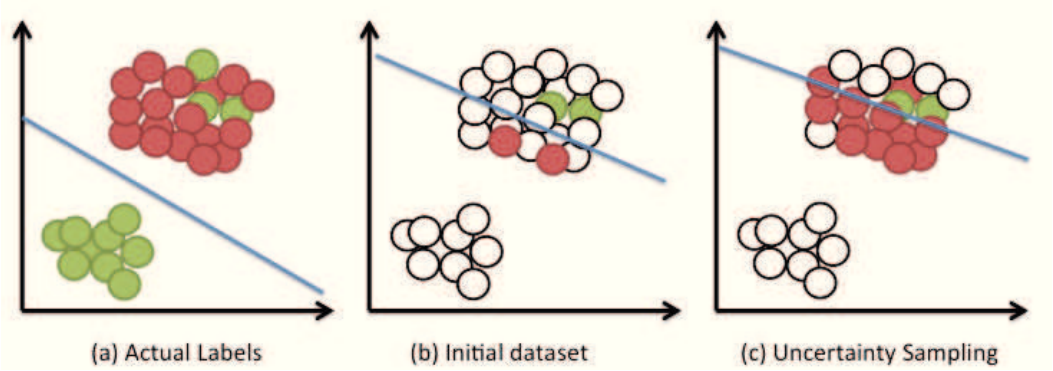

Figure 3.2

Therefore, the general performance of active learners may vary widely, since the assumptions behind the selection criteria may not hold in all played scenarios. In cases where the selection criterion does not work properly due to inappropriate assumptions, the correspondent query strategy may generate very poor training sets. Hence, the associated active learner may yield a general performance even worse than one would obtain with the simple random sampling (SRS), *i.e.* passive learning. This happens

because the query strategy generates biased training sets, *i.e.* samples with distribution different from the population.

Note that, from the sampling standpoint, active learning is actually a biased sampling procedure since it *systematically* favors some observations over others through the selection criterion. This leads to an overexploitation of some regions of the input space, which results in a sample with underlying density distribution different from the distribution the samples are drawn, *i.e.* the actual population.

In this way, biased samples are hereby considered as those that the underlying distribution of the input variables differs from the original population distribution. As labels are supposedly unknown at the moment of the selection, these are therefore independent and identically distributed, *i.e.* drawn according to the label population distribution.

Accordingly, active learning generates biased training sets in order to generalize supervised models for the entire input space(DASGUPTA, 2009). The philosophy behind it lies in intentionally introducing bias in the training set according to a heuristic so as to obtain the most informative observations for learning models with great generalization ability. Thus, active learning draws observations from a distribution different from the original population.

Nevertheless, the presence of bias in the training set may also conduct to bad models, whenever the selection criterion is mistaken. A query strategy may form training sets with regions without representativeness in the input space, leading to poor models for these regions. This occurs because uninformative observations are selected and the informative ones are left behind. Hence, the introduced bias increases the risk of loosing informative observations as the underlying distribution of the sample differs from the population.

Therefore, the amount of bias in the training set reflects how much aggressive the active learning strategy is. The more risky the bet, the higher the reward is. In this way, there is an inherently tradeoff between the reward (i.e. the general model accuracy) and the chance to fail by favoring uninformative observations due to mistaken assumptions about the scenario.

In Figure 3.3, an example is illustrated where the uncertainty sampling query strategy, based on a mistaken model, overexploits a cluster of observations.

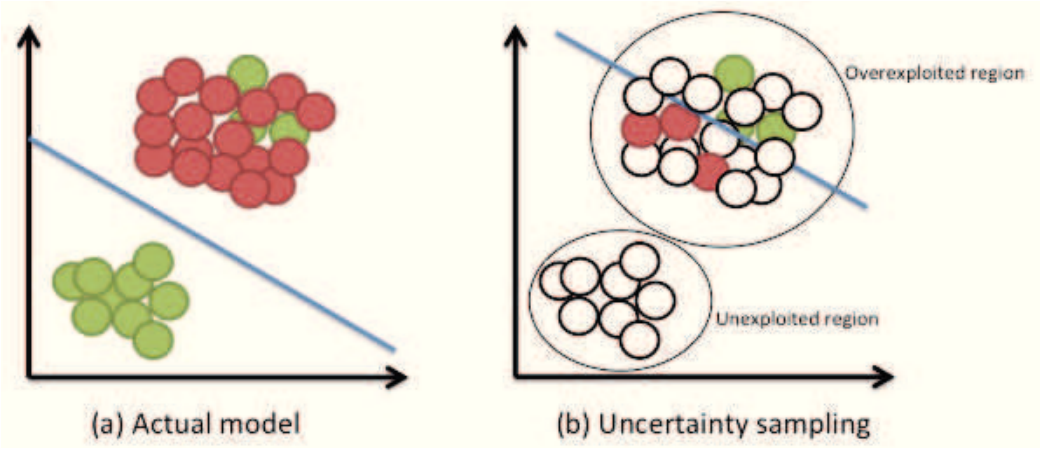

Figure 3.3

Figure 3.4 depicts the resulting densities of both population and the training set generated by uncertainty sampling. Note that, the underlying distribution of the training set is away different from the population.

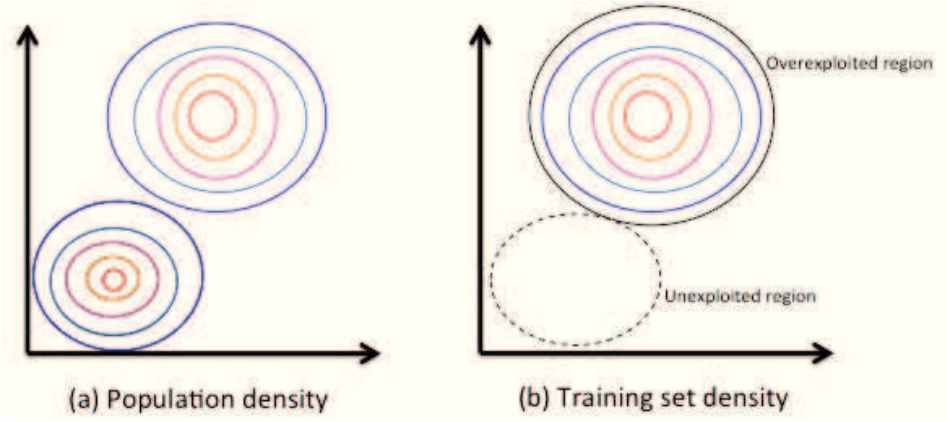

Figure 3.4

In this context, a major motivation for the proposed general query strategy of this thesis concerns the bias in the training set. As generating biased training sets is associated with the risk of failing, our proposal is to provide a general query strategy that avoids bias. In this way, the risk of overexploiting regions of uninformative observations and leaving behind others with informative ones is minimized. Moreover, this is a key benefit in the proposed strategy, since no assumption about data and model is required.

This proposal may seem controversial as the philosophy behind active learning relies on the tradeoff between bias and informative observations. However, this proposal of

active learning aims at selecting informative observations and, at the same time, maintaining the training set unbiased.

3.2 General Proposal

Part of this thesis methodological contribution consists in proposing a novel general active learning query strategy based on the aforementioned motivation, *i.e.* avoiding bias in training sets. As the bias increases the chance to obtain poor models, our proposal aims at producing unbiased training sets by selecting observations that better reproduces the original population distribution. For that, the selection criterion should favor observations that generate training sets with the best fit to the population distribution of the input space. In other words, the proposed query strategy aims at selecting the observations that keep the sample distribution as close as possible to the original population distribution. Thus, the choice of a candidate observation is done according to how representative the training set is of the population distribution, resulting in a sample as distributed as the population.

At a first glance, this proposal reminds a simple random sampling (SRS), which generates a set of independent and identically distributed observations from the population. However, the SRS procedure is subject to sampling error due to the randomness of the selection. As a consequence, its performance varies widely for small samples. To obtain a robust training set, the SRS requires a large number of observations so as to minimize the randomness of the estimation (WASSERMAN, 2003). Notice that, a large set may become prohibitive by taking into account the labeling cost.

For instance, let us consider a sample obtained by independent launches of a fair coin (*i.e.* probability of 0.5 for head and tail). Let us suppose that one draws a sample of size 10 by SRS. It would be perfectly acceptable, for this small number of launches, to get, for instance, 3 tails and 7 heads, whose proportion is far away from the expected population distribution. Of course, as the sample size grows, the estimated proportion of heads and tails tends to the expected (0.5-0.5).

Accordingly, the proposed query strategy will be concerned with both the sampling bias and the sampling error. To do so, this query strategy provides samples from which the estimated distribution is as close as possible to the underlying distribution of the

population. In this way, this proposal minimizes both the sampling bias and sampling error.

In this context, a utility function is provided to measure how close the shape of the estimated distribution of the sample is to the population distribution. Hence, one is able to look for those observations that shorten the distance between both distributions. For that, distance measures between probability density functions are considered, namely divergences.

In order to handle the sample and the population distributions, one estimates their correspondent probability density functions (pdf) from the observations in the current sample and in the initial pool, respectively. As new observations are picked for the sample, its estimated pdf changes due to these new observations. Kernel density estimation methods and the information-theoretic framework are used to estimate both pdfs (sample and population) and to handle the distance between them.

Query strategies based on this proposed general strategy may be time consuming and computationally costly, as it requires to re-estimate pdfs and to compute their distance from all observations of the pool at each time one is selected. To avoid this issue, a query strategy has been proposed based on an analytical solution that yields a straightforward mathematical expression as a utility function, simplifying the proposal complexity. In addition, this utility function allows an interesting geometric interpretation for the selected observations.

In the next sections, we present the proposed general strategy, some measures to compare probability distributions (divergences), probability density estimation methods, and finally the active learning query strategy based on the proposed general query strategy.

3.3 Proposal Formalization

In this section, we formally describe the general active learning query strategy.

The key idea is to reduce as much as possible the distance between the probability density functions (pdfs) of the training set (sample) and the pool (population).

Let $P \stackrel{\text{def}}{=} \{x_i \mid i = 1 \dots N_1\}$ be a set of observations drawn from a population that obeys a pdf $p(X)$.

We denote by $\Delta\big(p(X), q(X)\big)$ a distance measure between two pdfs $p(X)$ and $q(X)$.

Let $Q \subseteq P$ be a sample of size $N_2 \leq N_1$ generated by a pdf $q(X)$.

The goal is therefore to find a set of observations Q such that

- $\#Q \ll \#P$, where # denotes the cardinality, and
- the distance $\Delta\big(p(X), q(X)\big)$ is minimized.

In this way, the proposed query strategy aims at searching for a sub-set Q from a pool of observations P, such that $\#Q$ is as small as possible and at the same time the pdfs $p(X)$ and $q(X)$ are as close as possible.

This proposal can be seen as finding the sample with the best goodness-of-fit of probability distribution of a finite population(WASSERMAN, 2003). The proposal is therefore approaching the estimated pdf of Q to the pdf of P, aiming at producing a sample with the best goodness-of-fit for the pdf of the population. In this way, one hopes to obtain a training set Q containing the most representative observations for the entire population P.

3.4 Theoretical foundation

The theoretical foundation behind the proposed query strategy relies on the reduction of the sample space for estimating supervised models. The proposed strategy indeed cuts off the sample space of all possible training sets by eliminating samples that do not lead to learning accurate models. Consequently, this increases the probability of obtaining training sets from which accurate models can be learned.

In order to provide a theoretical support for this proposal, we use the statistical learning framework, where a supervised model consists in a joint probability distribution of input and output variables.

Let us consider the continuous random variable $X \in \mathbb{R}^d$ with pdf $p(X)$ as input and the discrete random variable $Y \in \{y_1, y_2, \dots, y_m\}$ with probability mass function (pmf) $Pr(Y)$ as output.

Let D_N be a sample of N independent and identically distributed observations (x, y) drawn from the joint pdf $p(X, Y)$, *i.e.* D_N is drawn by the SRS.

Let Ω_N be the sample space of all possible outcomes of training sets D_N and let $\hat{\theta}_X$ and $\hat{\theta}_{XY}$ be the unbiased estimators of the population parameters θ_X and θ_{XY} of $p(X)$ and $p(X,Y)$, respectively.

As $\hat{\theta}_X$ and $\hat{\theta}_{XY}$ are functions of D_N, these are associated with different elements of the sample space Ω_N, and hence are random variables with probability distributions $p(\hat{\theta}_X|D_N)$ and $p(\hat{\theta}_{XY}|D_N)$, respectively, and joint distribution given by $p(\hat{\theta}_X, \hat{\theta}_{XY}|D_N)$.

A classifier (or hypothesis) $h(x)$ assigns a label $y \in Y$ to a given observation $x \in X$ according to the following rule:

$$h(x) = \underset{\forall y_i}{\operatorname{argmax}}\, Pr(Y = y_i|X = x). \qquad \text{Eq. 10}$$

By applying the Bayes rule to the conditional pmf $Pr(Y|X)$ in Eq. 10, one obtains

$$h(x) = \underset{\forall y_i}{\operatorname{argmax}} \frac{p(X = x, Y = y_i)}{p(X = x)}. \qquad \text{Eq. 11}$$

Note that, $p(X) = \sum_{\forall y_i \in Y} p(X, Y = y_i)$ and hence the distribution $Pr(Y|x)$, the core of the supervised model, in fact relies only on the estimation of the joint pdf of the input and output variables, *i.e.* $p(X,Y)$.

In this context, the major issue of active learning consists in developing query strategies so as to generate training sets D_N^* from which one is able to estimate as accurate as possible the joint pdf $p(X,Y)$. The goal is to define selection criteria that favor training sets $D_N^* \in \Omega_N$ in which the estimator $\hat{\theta}_{XY}$ of $p(X,Y)$ is as close as possible to the 'true' population parameter θ_{XY}. This implies that the distribution $p(\hat{\theta}_{XY}|D_N^*)$ should be as sharp as possible on θ_{XY}.

Accordingly, our objective is to prove that the proposed general query strategy of this thesis selects training sets D_N^* so that the distribution $p(\hat{\theta}_{XY}|D_N^*)$ is sharper on the correspondent 'true' parameter θ_{XY} than $p(\hat{\theta}_{XY}|D_N)$, provided by the SRS. We want to

show that the variability of the estimator $\hat{\theta}_{XY}$ obtained from D_N^* is smaller than the variability obtained with D_N.

As the proposed selection criterion consists in minimizing the distance measure $\Delta\big(p(X), q(X)\big)$ between the distributions of the sample and the pool, this generates sets D_N^* in which the associated estimator $\hat{\theta}_X$ is as close as possible to the 'true' population parameter θ_X. Hence, the proposed criterion produces training sets D_N^* with an underlying distribution of $\hat{\theta}_X$ sharp on θ_X, since $\hat{\theta}_X$ is unbiased (*i.e.* $E\big[\hat{\theta}_X\big] = \theta_X$). Consequently, the variance of $\hat{\theta}_X$ is equal or smaller than the variance of $\hat{\theta}_X$ obtained by the SRS.

In this context, we propose the following theorem:

Theorem 1: Let D_N^* and D_N be random sets of the sample space Ω_N with underlying probability distributions of the estimator $\hat{\theta}_X$ such that

$$var\big(\hat{\theta}_X | D_N^*\big) \leq var\big(\hat{\theta}_X | D_N\big) \qquad \text{Eq. 12}$$

Then, it turns out that

$$var\big(\hat{\theta}_{XY} \mid D_N^*\big) \leq var\big(\hat{\theta}_{XY} \mid D_N\big). \qquad \text{Eq. 13}$$

Proof: Let $\Delta_{\hat{\theta}}^2 \overset{\text{def}}{=} \big(\hat{\theta} - \theta\big)^2$ be the squared error of an estimator $\hat{\theta}$. Hence, by the definition of variance, one has that

$$var\big(\hat{\theta}\big) = E\left[\big(\hat{\theta} - \theta\big)^2\right] = E\big[\Delta_{\hat{\theta}}^2\big] = \int_{-\infty}^{+\infty} \Delta_{\hat{\theta}}^2 \, p\big(\hat{\theta}\big) \, d\hat{\theta}. \qquad \text{Eq. 14}$$

By using Eq. 14 for $\hat{\theta}_X$ obtained from D_N^*, one gets

$$var\left(\hat{\theta}_X \mid D_N^*\right) = \int_{-\infty}^{+\infty} \Delta_{\hat{\theta}_X}^2 \, p\left(\hat{\theta}_X \mid D_N^*\right) d\hat{\theta}_X. \qquad \text{Eq. 15}$$

By using Eq. 15 in Eq. 12, one obtains the inequality

$$\int_{-\infty}^{+\infty} \Delta_{\hat{\theta}_X}^2 \, p\left(\hat{\theta}_X \mid D_N^*\right) d\hat{\theta}_X \leq \int_{-\infty}^{+\infty} \Delta_{\hat{\theta}_X}^2 \, p\left(\hat{\theta}_X \mid D_N\right) d\hat{\theta}_X. \qquad \text{Eq. 16}$$

According to Eq. 16, the density of $p\left(\hat{\theta}_X \mid D_N^*\right)$, compared with $p\left(\hat{\theta}_X \mid D_N\right)$, is higher for small values of $\Delta_{\hat{\theta}_X}^2$ and lower for larges $\Delta_{\hat{\theta}_X}^2$. Therefore, the area under the curve $p\left(\hat{\theta}_X | D_N^*\right)$ is more concentrated (sharper) on the population parameter θ_X than $p\left(\hat{\theta}_X | D_N\right)$, as $\Delta_{\hat{\theta}_X}^2 = \left(\hat{\theta}_X - \theta_X\right)^2$.

Analogous to Eq. 15, the variance of $\hat{\theta}_{XY}$ can be written as

$$var\left(\hat{\theta}_{XY} \mid D_N^*\right) = \int_{-\infty}^{+\infty} \Delta_{\hat{\theta}_{XY}}^2 \, p\left(\hat{\theta}_{XY} \mid D_N^*\right) d\hat{\theta}_{XY}. \qquad \text{Eq. 17}$$

As the estimators $\hat{\theta}_{XY}$ and $\hat{\theta}_X$ are associated with the same D_N^*, it turns out that

$$p\left(\hat{\theta}_{XY} \mid D_N^*\right) = \int_{-\infty}^{+\infty} p\left(\hat{\theta}_{XY}, \hat{\theta}_X \mid D_N^*\right) d\hat{\theta}_X. \qquad \text{Eq. 18}$$

By re-writing Eq. 17 with Eq. 18, one gets

$$var\left(\hat{\theta}_{XY} \mid D_N^*\right) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \Delta^2_{\hat{\theta}_{XY}} \, p\left(\hat{\theta}_{XY}, \hat{\theta}_X \mid D_N^*\right) d\hat{\theta}_X \, d\hat{\theta}_{XY}. \qquad \text{Eq. 19}$$

By applying the Bayes rule in Eq. 19, one obtains

$$var\left(\hat{\theta}_{XY} \mid D_N^*\right) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \Delta^2_{\hat{\theta}_{XY}} \, p\left(\hat{\theta}_{XY} \mid \hat{\theta}_X, D_N^*\right) p\left(\hat{\theta}_X \mid D_N^*\right) d\hat{\theta}_X \, d\hat{\theta}_{XY}. \qquad \text{Eq. 20}$$

As $p\left(\hat{\theta}_X \mid D_N^*\right)$ gets sharper, $p\left(\hat{\theta}_{XY}, \hat{\theta}_X | D_N^*\right)$ also changes, becoming sharper toward the axis $\hat{\theta}_X$. Hence, $p\left(\hat{\theta}_{XY} | D_N^*\right)$ may also become shaper if there is any dependence between $\hat{\theta}_{XY}$ and $\hat{\theta}_X$. Consequently, the integral in Eq. 20 must be equal or smaller for D_N^* than for D_N since $p\left(\hat{\theta}_X \mid D_N^*\right)$ is sharper than $p\left(\hat{\theta}_X \mid D_N\right)$. Therefore, one has that

$$\begin{aligned} &\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \Delta^2_{\hat{\theta}_{XY}} \, p\left(\hat{\theta}_{XY} | \hat{\theta}_X, D_N^*\right) p\left(\hat{\theta}_X \mid D_N^*\right) d\hat{\theta}_X \, d\hat{\theta}_{XY} \\ &\qquad \leq \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \Delta^2_{\hat{\theta}_{XY}} \, p\left(\hat{\theta}_{XY} | \hat{\theta}_X, D_N\right) p\left(\hat{\theta}_X \mid D_N\right) d\hat{\theta}_X \, d\hat{\theta}_{XY}, \end{aligned} \qquad \text{Eq. 21}$$

Consequently, one obtains that

$$var\left(\hat{\theta}_{XY} \mid D_N^*\right) \leq var\left(\hat{\theta}_{XY} \mid D_N\right) \blacksquare \qquad \text{Eq. 22}$$

The equality in Eq. 22 only holds whenever $\hat{\theta}_{XY}$ and $\hat{\theta}_X$ are independent. However, supervised learning relies on the hypothesis that X and Y are dependent, and hence $\hat{\theta}_{XY}$ and $\hat{\theta}_X$ are dependent. Therefore, under this assumption, one gets

$$var(\hat{\theta}_{XY} \mid D_N^*) < var(\hat{\theta}_{XY} \mid D_N). \quad \text{Eq. 23}$$

In addition, the higher the dependence between $\hat{\theta}_{XY}$ and $\hat{\theta}_X$, the more information $\hat{\theta}_X$ transfers to $\hat{\theta}_{XY}$. In case of total dependence, one has that knowing $\hat{\theta}_X$ implies to completely know $\hat{\theta}_{XY}$.

Therefore, the smaller the $var(\hat{\theta}_X \mid D_N^*)$, the smaller the $var(\hat{\theta}_{XY} \mid D_N^*)$. However, depending on how dependent $\hat{\theta}_{XY}$ is of $\hat{\theta}_X$, $var(\hat{\theta}_{XY} \mid D_N^*)$ can be even smaller.

In subsection 3.5, the proposed query strategy is described. This strategy selects observations in order to generate training sets D_N^* such that **theorem 1** holds.

3.5 The proposed general query strategy

In this section we describe the general procedure that allows the implementation of the proposed query strategy. Therefore, this procedure aims to discover the smallest subset Q of a given pool P of observations such that the distance between their correspondent pdfs is minimized. This procedure is described as follows:

Algorithm General Query Strategy Proposal

Input: P - pool of unlabeled observations

Output: Q - training set of labeled observations

Set $Q = \emptyset$.

Estimate the pdf $\hat{p}$ using the set P of observations.

Repeat

For each $x_i \in P - Q$ do

Estimate the pdf $\hat{q}$ using the set $Q \cup \{x_i\}$

Compute $\Delta(\hat{q}, \hat{p})$

$u(x_i) \coloneqq \Delta(\hat{q}, \hat{p})$

End

$x_* \coloneqq \arg\min_{x_i} u(x_i)$

$Q \coloneqq Q \cup \{x_*\}$

Estimate the pdf $\hat{q}$ using the new Q.

Until $\Delta(\hat{q}, \hat{p}) > \delta$

Algorithm 3.1 – General Query Strategy Proposal

Note that this is a greedy procedure independent of knowledge about the labels of the observations added into Q and of any associated supervised model. The proposed query strategy relies only on the distance measure $\Delta(\hat{q}, \hat{p})$ between the pdfs $\hat{q}$ and $\hat{p}$. Neither the labeled observations nor the classification model are required in the selection process.

A tolerance value δ for the distance between the pdfs is defined as a stopping criterion, otherwise it would stop when $\hat{q} = \hat{p}$. However, one could alternatively establish a maximum number of observations in the set Q, without taking $\Delta(\hat{q}, \hat{p})$ into account to stop.

There are two important issues concerned with the implementation of this procedure: 1) the estimation of the involved probability densities, and 2) the computation of the distance between them.

Furthermore, note that this procedure is time consuming and computationally costly. A pair of nested loops constitutes this procedure. First, for each observation $x_i \in P$, the

procedure forms a new set $Q \cup \{x_i\}$, estimates its pdf $\hat{q}$, and calculates the distance $\Delta(\hat{q}, \hat{p})$ to choose the optimal observation, which is transferred from P to Q. The second loop repeats the first one but with the updated set Q. Therefore, the proposed procedure complexity is $O(n^2)$.

In order to handle these computing issues, a feasible query strategy using the proposed general strategy is provided by using the Information Theoretic Learning framework (PRINCIPE, 2010) for analytically building an utility function that implements the target selection criterion. In the next sections, we describe this framework.

3.6 Kernel Density Estimation

There are several nonparametric methods for probability density estimation available in the literature. Among them is the widely used *Kernel Density Estimation* (KDE), also known as *Parzen* windowing (DUDA *et al.*, 2000). This method empirically estimates the probability density function by taking into account the local density of each observation in the feature space $\mathcal{F}$.

For a given set of *iid* observations $\{x_1, \ldots, x_N\}$ drawn from an unknown pdf f, the KDE provides a pdf estimate $\hat{f}(\mathrm{x})$ given by:

$$\hat{f}(x) = \frac{1}{N}\sum_{t=1}^{N} K_h(x, x_t), \qquad \text{Eq. 24}$$

where K_h is a Kernel function (or Kernel, for simplicity), which is a symmetric function that integrates one (DUDA *et al.*, 2000).

The key idea is to evaluate the density $\hat{f}(x)$ for a given observation x by computing the average proportion of the number of observations falling in a hyper-volume in the feature space $\mathcal{F}$, a.k.a. Kernel space. The hyper-volume shape is intrinsically related to the Kernel function K_h, where h is a scale factor that acts as a smoother parameter for the KDE. The larger the h value, the smoother the pdf estimate is. The value of this parameter should be carefully chosen, since it might lead to over-fitting or under-fitting of the pdf estimate(BISHOP, 2007, DUDA *et al.*, 2000).

The Kernel function $K_h(x, x_t)$ can be expressed as an inner product in the feature space $\mathcal{F}$ in the form $K_h(x_a, x_b) = \phi(x_a)^T \phi(x_b) = \langle \phi(x_a), \phi(x_b) \rangle$ (here $\langle \cdot, \cdot \rangle$ denotes the inner product), where the function ϕ defines the mapping $\phi: X \to \mathcal{F}$. This is known as 'Kernel trick', a way to measure the similarity between two points x_a and x_b of the input space X in the much higher dimensionality feature space $\mathcal{F}$, without explicitly computing the mapping ϕ (DUDA *et al.*, 2000). This technique is widely used in many Machine Learning algorithms, particularly in Support Vector Machines (SVM)(BISHOP, 2007, DUDA *et al.*, 2000).

Although several Kernels could be chosen, we opted for a Gaussian kernel defined by $G_h(x, x_0) = \frac{1}{h\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-x_0}{h}\right)^2}$. This is indeed a Gaussian *pdf* with standard deviation h, centered at the observation x_0. This function is especially interesting due to the property of the convolution of Gaussians, which will be described shortly (PRINCIPE, 2010, JENSSEN *et al.*, 2006).

3.7 Divergence Metrics

There are several measures of 'distance' between probability distributions known as divergences(PRINCIPE, 2010). These measures are pseudo-metrics since they do not satisfy some of the metrics axioms, such as symmetry and triangle inequality. In this section, we present two divergences based on information theory concepts that are used in the query strategy proposed in this thesis.

The ***Cauchy-Schwarz*** **divergence** is a symmetric measure that allows the comparison between two probability distributions p and q. It can be directly derived from the Cauchy-Schwarz (CS) inequality

$$\left| \int_{-\infty}^{+\infty} p(x)q(x)dx \right|^2 \leq \int_{-\infty}^{+\infty} |p(x)|^2 dx \int_{-\infty}^{+\infty} |q(x)|^2 dx. \qquad \text{Eq. 25}$$

Clearly, in Eq. 25, the equality holds if and only if $p(x) = q(x)$ for all the domain of x.

The CS divergence(JENSSEN *et al.*, 2006) is then defined as

$$D_{CS}(p,q) \overset{\text{def}}{=} -\log \frac{\int_{-\infty}^{+\infty} p(x)q(x)dx}{\sqrt{\int_{-\infty}^{+\infty} p^2(x)dx \int_{-\infty}^{+\infty} q^2(x)dx}}. \quad \text{Eq. 26}$$

In this way, the divergence $D_{CS}(p,q)$ vanishes as p approaches q. One should note that $0 \leq D_{CS}(p,q)$.

The ***Integrated Squared Error*** (ISE) (BISHOP, 2007, JENSSEN *et al.*, 2006, PRINCIPE, 2010) is also an alternative of measure the distance between two *pdfs*. It computes the total area under the function that represents the squared difference between the two pdfs as follows:

$$\begin{aligned} ISE(p,q) &\overset{\text{def}}{=} \int_{-\infty}^{+\infty} [p(x) - q(x)]^2 \, dx \\ &= \int_{-\infty}^{+\infty} p^2(x) \, dx - 2\int_{-\infty}^{+\infty} p(x)q(x) \, dx + \int_{-\infty}^{+\infty} q^2(x) \, dx. \end{aligned} \quad \text{Eq. 27}$$

As one should note, the ISE(p, q) is a non-negative symmetric function and shrinks to 0 as p approaches q. This measure allows an analytical solution to implement the proposed query strategy, providing a trick for fast computing the utility function compared with the D_{CS}.

3.8 ISE-based Query Strategy

In this section, we propose the ISE-based Query Strategy, which is a query strategy based on the proposed general query strategy by using the ISE as the distance measure between pdfs, *i.e.* $\Delta(\hat{q},\hat{p}) = ISE(\hat{p},\hat{q})$. In order to handle the estimated pdfs, the Information Theoretic framework described in (PRINCIPE, 2010) is used by allowing an analytical solution for the proposed query strategy. We start by describing some theoretical foundations introduced in (PRINCIPE, 2010) and follow by presenting the developments proposed in this thesis aiming to provide a feasible implementation of the proposed query strategy.

3.8.1 Integrated Squared Error of Kernel Density Estimation

By the Kernel definition of Eq. 24, we have the following estimates for the pdfs from the sample sets P and Q, respectively:

$$\hat{p}(x) = \frac{1}{N_1}\sum_{i=1}^{N_1} G_{h_1}(x, x_i), \qquad \text{Eq. 28}$$

and

$$\hat{q}(x) = \frac{1}{N_2}\sum_{j=1}^{N_2} G_{h_2}(x, x_j). \qquad \text{Eq. 29}$$

where G_h is the Gaussian kernel.

By substituting the densities estimates $\hat{p}(x)$ and $\hat{q}(x)$ in Eq. 27, one obtains the following ISE estimator:

$$\begin{aligned}\widehat{ISE}(p,q) = & \int \left[\frac{1}{N_1}\sum_{i=1}^{N_1} G_{h_1}(x, x_i)\right]^2 dx \\ & - 2\int \left[\frac{1}{N_1}\sum_{i=1}^{N_1} G_{h_1}(x, x_i)\right]\left[\frac{1}{N_2}\sum_{i=1}^{N_2} G_{h_2}(x, x_j)\right] dx \\ & + \int \left[\frac{1}{N_2}\sum_{i=1}^{N_2} G_{h_2}(x, x_j)\right]^2 dx,\end{aligned} \qquad \text{Eq. 30}$$

where the bounds of the integrals are omitted for simplicity as each one integrates over $]-\infty; +\infty[$.

Re-writing Eq. 30 with summations of integrals and expanding the squares, one gets the following expression:

$$\widehat{ISE}(p,q) = \frac{1}{N_1^2}\sum_{i,i\prime=1}^{N_1,N_1}\int G_{h_1}(x,x_i)G_{h_1}(x,x_{i\prime})\,dx - 2\frac{1}{N_1}\frac{1}{N_2}\sum_{i,j=1}^{N_1,N_2}\int G_{h_1}(x,x_i)G_{h_2}(x,x_j)dx + \frac{1}{N_2^2}\sum_{j,j'=1}^{N_2,N_2}\int G_{h_2}(x,x_j)G_{h_2}(x,x_{j'})\,dx. \quad \text{Eq. 31}$$

The convolution theorem for Gaussians (PRINCIPE, 2010) states that

$$\int G_{h_1}(x,x_t)G_{h_2}(x,x_l)\,dx = G_{h_1+h_2}(x_t,x_l). \quad \text{Eq. 32}$$

and by applying Eq. 32 into Eq. 31, one gets the expression(JENSSEN *et al.*, 2006)

$$\widehat{ISE}(p,q) = \frac{1}{N_1^2}\sum_{i,i'=1}^{N_1,N_1}G_{2h_1}(x_i,x_{i'}) - 2\frac{1}{N_1}\frac{1}{N_2}\sum_{i,j=1}^{N_1,N_2}G_{h_1+h_2}(x_i,x_j) + \frac{1}{N_2^2}\sum_{j,j'=1}^{N_2,N_2}G_{2h_2}(x_j,x_{j'}). \quad \text{Eq. 33}$$

Interestingly, one should highlight that Eq. 33 provides an analytic expression for the $\widehat{ISE}$ as a function exclusively of the observations in P and Q.

3.8.2 Selecting a new observation

Here, the key idea is to use the estimate $\widehat{ISE}$ as the measure $\Delta(\hat{q},\hat{p})$. Therefore the ISE-based Query Strategy aims to minimize the $\widehat{ISE}$ as observations of P are added into Q.

Let $x_* \in P$ be a candidate observation to be added into Q. We should therefore analyze the impact of x_* on $\widehat{ISE}(p,q)$.

The estimate $\hat{q}$ in Eq. 29 should be updated by the addition of this hypothetical observation x_*. Hence, this in fact means to add a new parcel in the summation as follows:

$$\hat{q}(x) = \frac{1}{N_2+1}\left(\sum_{i=1}^{N_2} G_{h_2}(x, x_j) + G_{h_2}(x, x_*)\right). \qquad \text{Eq. 34}$$

By updating Eq. 31 with Eq. 34 and propagating this new parcel up to Eq. 33, one obtains

$$\begin{aligned}\widehat{ISE}(p,q) = {} & \frac{1}{N_1^2}\sum_{i,i\prime=1}^{N_1,N_1} G_{2h_1}(x_i, x_{i\prime}) \\ & - 2\frac{1}{N_1}\frac{1}{N_2+1}\left[\sum_{i,j=1}^{N_1,N_2} G_{h_1+h_2}(x_i, x_j) + \sum_{i=1}^{N_1} G_{h_1+h_2}(x_i, x_*)\right] \\ & + \frac{1}{(N_2+1)^2}\left\{\sum_{j,j'=1}^{N_2,N_2} G_{2h_2}(x_j, x_{j'}) + 2\sum_{j=1}^{N_2} G_{2h_2}(x_j, x_*)\right. \\ & \left. + G_{2h_2}(x_*, x_*)\right\}.\end{aligned} \qquad \text{Eq. 35}$$

By eliminating the brackets, one gets

$$\widehat{ISE}(p,q) = \frac{1}{N_1^2}\sum_{i,i\prime=1}^{N_1,N_1} G_{2h_1}(x_i,x_{i\prime}) - 2\frac{1}{N_1(N_2+1)}\sum_{i,j=1}^{N_1,N_2} G_{h_1+h_2}(x_i,x_j)$$
$$- 2\frac{1}{N_1(N_2+1)}\sum_{i=1}^{N_1} G_{h_1+h_2}(x_i,x_*)$$
$$+ \frac{1}{(N_2+1)^2}\sum_{j,j'=1}^{N_2,N_2} G_{2h_2}(x_j,x_{j'})$$
$$+ 2\frac{1}{(N_2+1)^2}\sum_{j=1}^{N_2} G_{2h_2}(x_j,x_*) + \frac{1}{(N_2+1)^2}G_{2h_2}(x_*,x_*). \quad \text{Eq. 36}$$

Re-organizing the parcels, one gets the following expression:

$$\widehat{ISE}(p,q) = \frac{1}{N_1^2}\sum_{i,i\prime=1}^{N_1,N_1} G_{2h_1}(x_i,x_{i\prime}) - 2\frac{1}{N_1(N_2+1)}\sum_{i,j=1}^{N_1,N_2} G_{h_1+h_2}(x_i,x_j)$$
$$+ \frac{1}{(N_2+1)^2}\sum_{j,j'=1}^{N_2,N_2} G_{2h_2}(x_j,x_{j'})$$
$$- 2\frac{1}{N_1(N_2+1)}\sum_{i=1}^{N_1} G_{h_1+h_2}(x_i,x_*)$$
$$+ 2\frac{1}{(N_2+1)^2}\sum_{j=1}^{N_2} G_{2h_2}(x_j,x_*) + \frac{1}{(N_2+1)^2}G_{2h_2}(x_*,x_*). \quad \text{Eq. 37}$$

One can note that the first three parcels of Eq. 37 do not depend on x_*. A quantity $V(fg)$, called 'Information Potential', is defined in (PRINCIPE, 2010) as the potential energy between two *pdfs* f and g.

The estimator of $V(fg)$ is defined by

$$\hat{V}(fg) = \frac{1}{N_1 N_2} \sum_{i,j=1}^{N_1,N_2} G_{h_1+h_2}(x_i, x_j). \qquad \text{Eq. 38}$$

Let us define

$$\alpha \stackrel{\text{def}}{=} \frac{N_2}{(N_2+1)}. \qquad \text{Eq. 39}$$

And by applying Eq. 38 and Eq. 39 into Eq. 37, one obtains

$$\begin{aligned} \widehat{ISE}(p,q) = \; & \hat{V}(p^2) - 2\,\alpha\,\hat{V}(pq) + \alpha^2\,\hat{V}(q^2) \\ & \overbrace{- 2(1-\alpha)\frac{1}{N_1}\sum_{i=1}^{N_1} G_{h_1+h_2}(x_i, x_*)}^{\mathbf{A}} \\ & \overbrace{+ 2(1-\alpha)\,\alpha\frac{1}{N_2}\sum_{j=1}^{N_2} G_{2h_2}(x_j, x_*)}^{\mathbf{B}} + \overbrace{(1-\alpha)^2\,G_{2h_2}(x_*, x_*)}^{\mathbf{C}}. \end{aligned} \qquad \text{Eq. 40}$$

Now, we revisit the KDE definition of Eq. 24, and apply it in Eq. 40. In addition, one can note that $G_{2h_2}(x_*, x_*) = c$ is constant, as the value c is defined by the variance of the Gaussian. In this way, we obtain the following expression:

$$\begin{aligned} \widehat{ISE}(p,q) = \; & \hat{V}(p^2) - 2\,\alpha\,\hat{V}(pq) + \alpha^2\,\hat{V}(q^2) - \overbrace{2\,(1-\alpha)\,\hat{p}(x_*)}^{\mathbf{A}} \\ & + \overbrace{2\,(1-\alpha)\,\alpha\,\hat{q}(x_*)}^{\mathbf{B}} + \overbrace{(1-\alpha)^2 c}^{\mathbf{C}}. \end{aligned} \qquad \text{Eq. 41}$$

Re-organizing the terms by isolating the constants, one gets

$$\widehat{ISE}(p,q) = \hat{V}(p^2) - 2\,\alpha\,\hat{V}(pq) + \alpha^2\,\hat{V}(q^2) + (1-\alpha)^2 c + 2\,(1-\alpha)[\alpha\,\hat{q}(x_*) - \hat{p}(x_*)]. \quad \text{Eq. 42}$$

By Eq. 42, the observation x_* that minimizes the $\widehat{ISE}$ is the one that also minimizes $[\alpha\,\hat{q}(x_*) - \hat{p}(x_*)]$. Thus the ISE-based Query Strategy should use the utility function $H_{ISE}(x_*)$ given by

$$H_{ISE}(x_*) = \alpha\,\hat{q}(x_*) - \hat{p}(x_*) \quad \text{Eq. 43}$$

Hence, this provides a heuristic that selects the observation $x_* \in P - Q$, which is the most likely according to the pdf p and also the most unlikely according to the pdf q, taking into account the coefficient α.

In Eq. 42, one can clearly note that, as α approaches one, the addition of a new observation into Q minimizes less the $\widehat{ISE}$, since the coefficient of the last parcel tends to zero. Therefore, the more observations added in Q according to the utility function H_{ISE}, the less impact a new observation has on the $\widehat{ISE}$.

3.8.3 Geometry of ISE-based Query Strategy

As aforementioned, the Kernel function K_h is represented as the inner product between the observations mapped in the feature space by the function ϕ, *i.e.*

$$K_h(x_a, x_b) = \langle \phi(x_a), \phi(x_b) \rangle. \quad \text{Eq. 44}$$

By substituting Eq. 44 into Eq. 24 (*pdf* estimation), one obtains the following expression:

$$\hat{f}(x) = \frac{1}{N}\sum_{t=1}^{N}\langle\phi(x_t), \phi(x)\rangle \quad \text{Eq. 45}$$

By the property of linearity of the inner product in the first argument,

$$\langle a, c\rangle + \langle b, c\rangle = \langle a + b, c\rangle, \quad \text{Eq. 46}$$

one can re-write Eq. 45 as:

$$\hat{f}(x) = \langle\frac{1}{N}\sum_{t=1}^{N}\phi(x_t), \phi(x)\rangle. \quad \text{Eq. 47}$$

The mean of the observations x_t in the Kernel space is expressed by

$$m = \frac{1}{N}\sum_{t=1}^{N}\phi(x_t). \quad \text{Eq. 48}$$

Hence, the pdf estimation $\hat{f}$ for an observation x consists in computing the inner product between x and mean vector of the observations in the Kernel space, *i.e.*

$$\hat{f}(x) = \langle m, \phi(x)\rangle. \quad \text{Eq. 49}$$

By applying the pdf estimate of Eq. 49 into Eq. 43, and defining

$$m_p = \frac{\sum_{i=1}^{N_1} \phi(x_i)}{N_1} \quad \text{Eq. 50}$$

and

$$m_q = \frac{\sum_{j=1}^{N_2} \phi(x_j)}{N_2}, \quad \text{Eq. 51}$$

the heuristic $H_{ISE}(x_*)$ can be written as

$$H_{ISE}(x_*) = \alpha \langle m_q, \phi(x_*) \rangle - \langle m_p, \phi(x_*) \rangle. \quad \text{Eq. 52}$$

One should note that, Eq. 52 allows an interesting geometric interpretation. By using the property of linearity of the inner product in the first argument in Eq. 46, one obtains:

$$H_{ISE}(x_*) = \langle \alpha m_q - m_p, \phi(x_*) \rangle. \quad \text{Eq. 53}$$

As the ISE-based Query Strategy should select the observation with the smallest $H_{ISE}(x_*)$, it means, from the Geometric point of view, we are searching for the observation that is orthogonal to the vector $\overrightarrow{\alpha m_q - m_p}$ in the Kernel space. Figure 3.5 depicts this geometric interpretation of H_{ISE}.

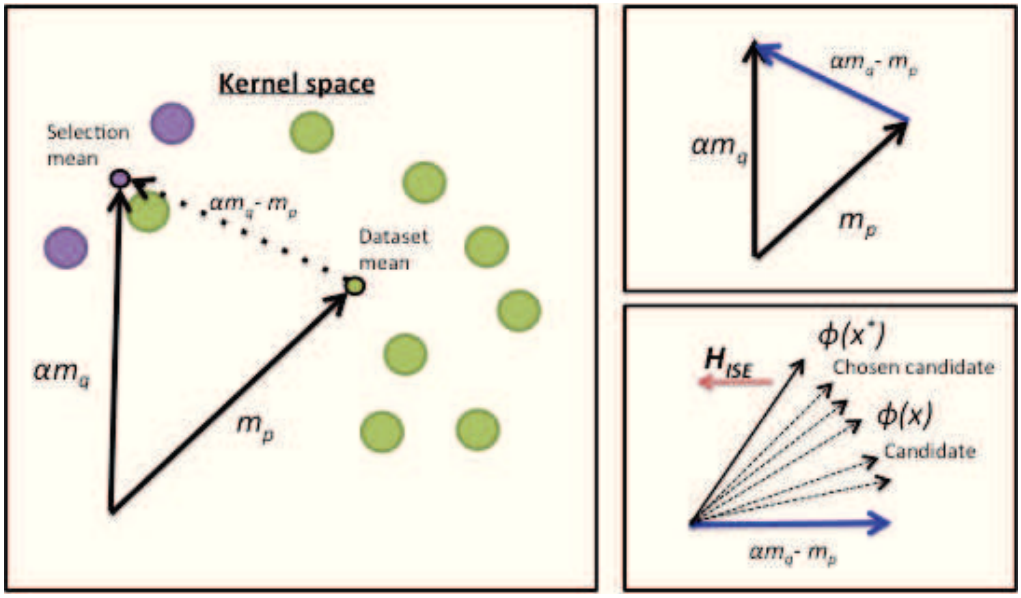

Figure 3.5

3.8.4 Tuned implementation

As described in the previous subsection, the ISE-based Query Strategy depends on the computation of the function $H_{ISE}(x_*)$, which requires the estimation of $\hat{p}$ and $\hat{q}$ for all N observations in P by the KDE described in Eq. 24. Hence, the complexity for estimating each pdfs is of $O(N^2)$. Consequently, the complexity of the ISE-based Query Strategy for adding n observations from the pool is $O(N^2 + nN^2)$ as we estimate $\hat{p}$ just once.

In order to speed up the ISE-based Query Strategy, we propose an implementation of complexity $O(N^2 + nN)$, where n is the number of observations to be selected and N is the initial number of observations in the pool. This implementation is based on a recursive trick, which avoids estimating the pdf $\hat{q}$ by the KDE whenever Q is updated. Instead of the KDE, the update of the pdf $\hat{q}$ for a new observation added in Q is performed with linear complexity of $O(N)$. In this way, the general complexity of ISE-based Query Strategy is given by $O(N^2 + nN)$. Note that, the first parcel of the complexity, N^2, is related to the complexity of the KDE for estimating $\hat{\mathrm{p}}$, performed once and independently of the number of observations to be selected.

Algorithm ISE-based Query Strategy

Input: P vector of N unlabeled observations

w Parzen-window

n number of observations to be selected

Output: Q vector of selected observations

Set $q[1 \dots N] = 0$ and $Q = \emptyset$;

Set $p[1 \dots N] = KDE(P, P, w)$;

While $|Q| \leq$ n **do**

$\alpha = \frac{|Q|}{|Q|+1}$;

$pos = min(\alpha * q - p)$;

$x_* = P(pos)$;

$q = \alpha\, q + (1 - \alpha)\, K_w(P, x_*)$;

$Q = Q \cup \{x_*\}$;

End while

Algorithm 3.2 – ISE-based Query Strategy

The function $KDE(P, P, w)$ computes the probability density of the N observations in P. For that, each observation is taken as the Kernel center, and its density is computed using w as the kernel bandwidth. This function is time consuming as it implements Eq. 24, *i.e.*, its complexity is $O(N^2)$. The probability densities of each element of P are then stored in the vector p of size N.

The vector q stores the probability densities of all unlabeled observations in P with Kernels centered in the observations in Q. As Q is supposed to be initially empty, the vector q initiates with all components null.

At each step of the loop, an observation x_* of P is added in Q. The observation x_* is chosen according to its correspondent $H_{ISE}(x_*)$. For that, the function $\underline{min}$ returns the index of the observation $x_* \in P$, which has the least H_{ISE}.

The pdf $\hat{q}$ estimated from Q is updated with linear complexity $O(N)$. Therefore, the KDE is avoided reducing the complexity of the algorithm. This reduction is due to a recursive trick provided by the following equation:

$$\hat{q}_{N+1}(x) = \alpha\, \hat{q}_N(x) + (1-\alpha) K_h(x, x_{N+1}), \qquad \text{Eq. 54}$$

where $\hat{q}_N(x)$ is the current estimated pdf of q and $\hat{q}_{N+1}(x)$ is the updated pdf of q by the addition of the observation x_{N+1} in Q, now with $N+1$ observations. This equation allows us to update $\hat{q}$ by one pass over the N observations in the pool, instead of performing all the KDE procedure for the new set Q. As follows we provide the proof for Eq. 54.

Proof of Eq. 54**:** By the definition of the KDE, we have that

$$\hat{q}_N(x) = \frac{1}{N}\sum_{t=1}^{N} K_h(x, x_t), \qquad \text{Eq. 55}$$

and, therefore

$$\hat{q}_{N+1}(x) = \frac{1}{N+1}\sum_{t=1}^{N+1} K_h(x, x_t). \qquad \text{Eq. 56}$$

By splitting the term corresponding to the observation x_{N+1}, one gets

$$\hat{q}_{N+1}(x) = \frac{1}{N+1}\left[\left(\sum_{t=1}^{N} K_h(x, x_t)\right) + K_h(x, x_{N+1})\right]. \qquad \text{Eq. 57}$$

By re-organizing the terms and applying $\hat{q}_N(x)$ in the expression, one obtains

$$\hat{q}_{N+1}(x) = \left[\left(\frac{N}{N+1}\hat{q}_N(x)\right) + \frac{1}{N+1}K_h(x, x_{N+1})\right]. \quad \text{Eq. 58}$$

By applying $\alpha = \frac{N}{N+1}$ in the last expression, one finally gets

$$\hat{q}_{N+1}(x) = \alpha\hat{q}_N(x) + (1-\alpha)K_h(x, x_{N+1})\blacksquare \quad \text{Eq. 59}$$

3.9 Summary and Conclusions

In this chapter we described a novel general query strategy, which relies on selecting observations in order to generate the most informative training sets free of bias. For that, a general procedure was developed to select observations by forming training sets from which the estimated pdf of the input variables, $\hat{p}(X)$, is as close as possible to the underlying pdf $p(X)$ of the pool.

A specific query strategy based on this general strategy was proposed by using the Integrated Square Error (ISE) as the distance measure between pdfs, a key ingredient of the proposed general procedure. This measure allowed reaching an analytical expression that provides a straightforward utility function for the selection criterion used by this query strategy.

A tuned implementation of the ISE-based query strategy was developed with linear complexity on the number of observations to be labeled. The main disadvantage of this query strategy is to adjust the kernel bandwidth in KDE, a well-known tough task that gets harder as the dimensionality grows.

In addition, the theoretical foundation that supports the proposed query strategy was provided. A formal proof was presented by giving guaranties of better generalization performance of the proposed general query strategy compared with the passive learning. This proof allows the development of a novel family of query strategies based on the idea of bias reduction in the input variables. Also, the proposed general query strategy allows us to propose specific query strategies for different divergence measures

provided by the information theory literature (PRINCIPE, 2010). The more precise the distance measure used, the better the query strategy.

Although the query strategy was developed for continuous variables, one is able to apply it for discrete variables as well. However, adaptions should be necessary in order to handle probability mass functions (pmf) instead of probability density functions (pdf). Besides, distance measures between pmfs should be necessary.

Chapter 4 **Experiments**

In this section we present the experiments in order to evaluate the proposed query strategy empirically. These experiments were performed in two parts: 1) a qualitative analysis of the proposed query strategy in simulated datasets and 2) a quantitative analysis comparing the query strategy with the passive learning both in simulated and real datasets. The ISE-based Query Strategy is considered in all experiments, since it implements the proposed query strategy.

We start with a brief description of our experimental setup and the simulated datasets and follow by providing a qualitative analysis and its correspondent quantitative results. We conclude this chapter by providing a performance comparison over two publically available real datasets.

4.1 Simulated Datasets

For examining the proposed strategy, eleven simulated datasets were designed. These datasets are 2 dimensional in order to allow the visualization of the results and to keep control of interest properties. Each dataset consists of 10.000 observations, in which half constitutes the initial pool for the quantitative analysis.

Although the datasets look simple for supervised learning, these allow us to examine and understand how the query strategy behaves as observations are labeled. Moreover, each dataset illustrates a set of properties of interest alive in any real dataset. Thus one is able to analyze the proposal from different standpoints, one by one.

The simulated datasets are organized in three categories: simple, cluster, and non-convex. In the next subsection, we describe each one.

4.1.1 Simple datasets

The simple datasets are composed of observations drawn at random from one or a mixture of Gaussian distributions. In these datasets, the goal is to simulate very simple distributions and their overlaps in the input space by generating a little of noise for the classification. Moreover, one aims to simulate nonlinearly separable datasets.

Figure 4.1, Figure 4.2 and Figure 4.4 illustrates these synthetic datasets, generated by separated Gaussian distributions. The dataset in Figure 4.3 was generated by two overlap Gaussian distributions. The number of observations drawn from each Gaussian was the same, keeping the classes balanced.

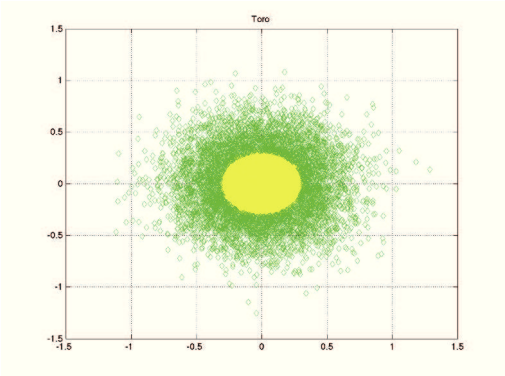

Figure 4.1

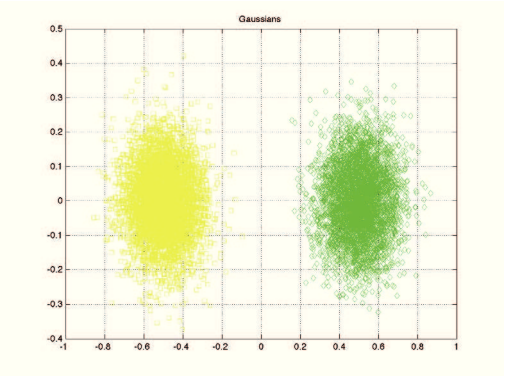

Figure 4.2

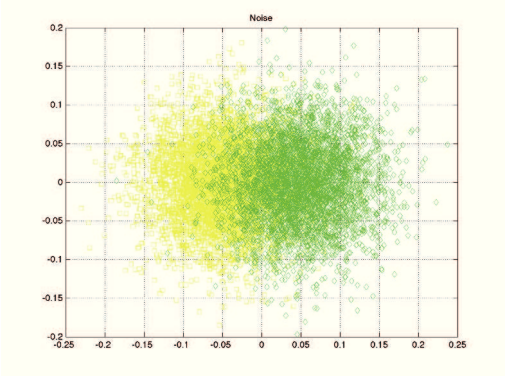

Figure 4.3

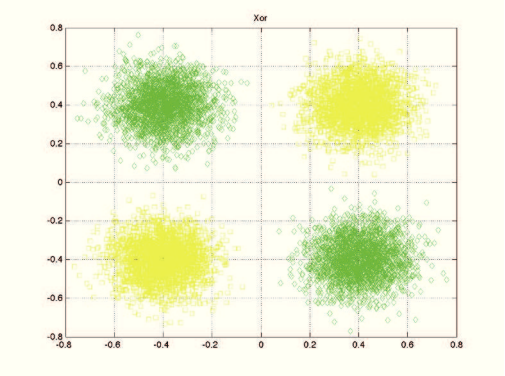

Figure 4.4

4.1.2 Clustered datasets

In this category, the simulated datasets are more complex, generated by a mixture of Gaussian distributions with different covariance. The number of Gaussian distributions and of observations drawn from each one is varied, by simulating cluster structure.

In Figure 4.5, the dataset was drawn from 4 Gaussian distributions with the same covariance matrix. The number of observations drawn from the two closest distributions

is a quarter the number of observations drawn from the farthest pair of Gaussian distributions.

Figure 4.6 depicts the dataset drawn from 2 pairs of Gaussian distributions with the same covariance matrix. The number of observations drawn from the distributions with the smaller covariance was a quarter the number drawn from the pair with larger covariance.

In Figure 4.7, the dataset was generated by 6 Gaussian distributions. Each Gaussian distribution presents the same covariance matrix and composes clusters of very close pairs and far away from one another. The number of observations drawn from each distribution is the same.

In Figure 4.8 the dataset is generated by 5 Gaussian distributions. There are 4 of them positioned as satellites around the fifth central Gaussian. Each satellite has a quarter the number of observations in the center distribution. The satellites have the same covariance, which is smaller than the covariance of the central Gaussian.

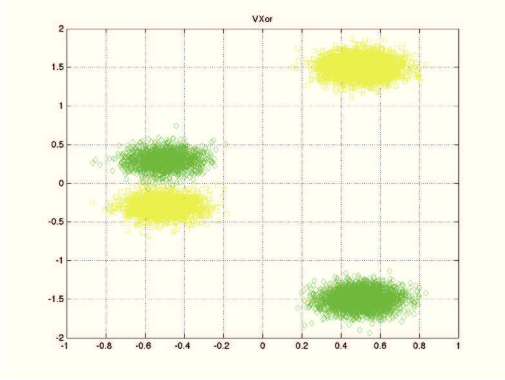

Figure 4.5

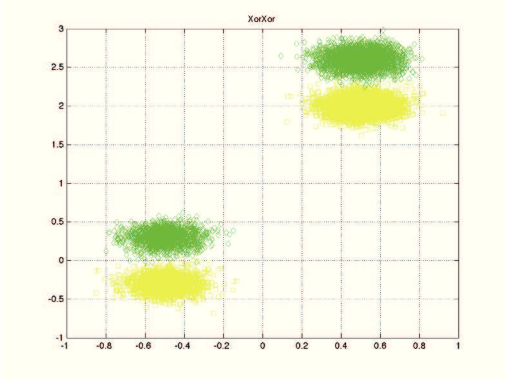

Figure 4.6

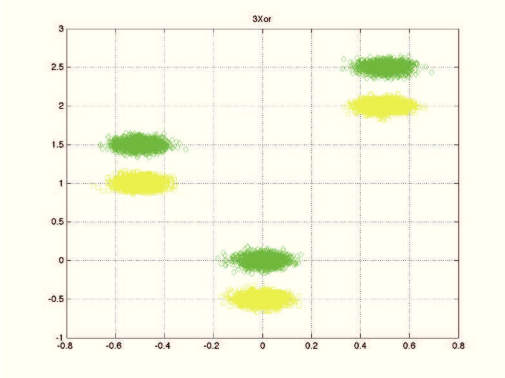

Figure 4.7

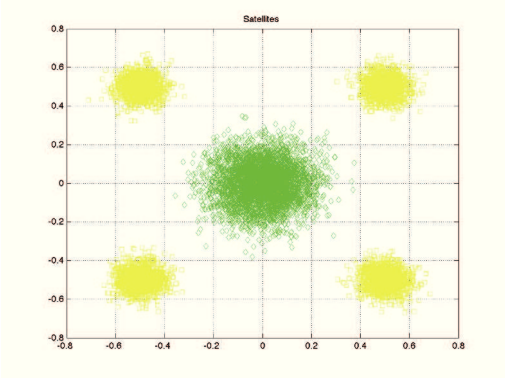

Figure 4.8

4.1.3 Non-convex datasets

These datasets aims to exploit difficulties related to the properties of datasets with non-convex shapes. Figure 4.9 and Figure 4.10, there are 2 half moons with the equal number of observations. However, the label distributions are different in both figures.

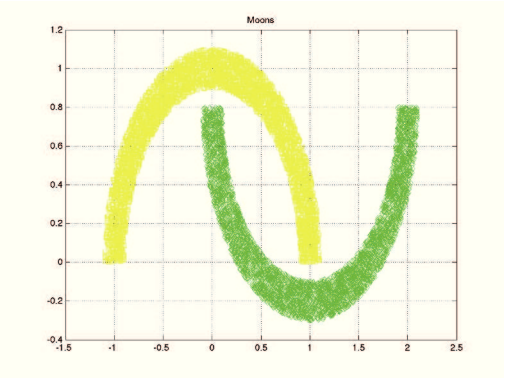

Figure 4.9

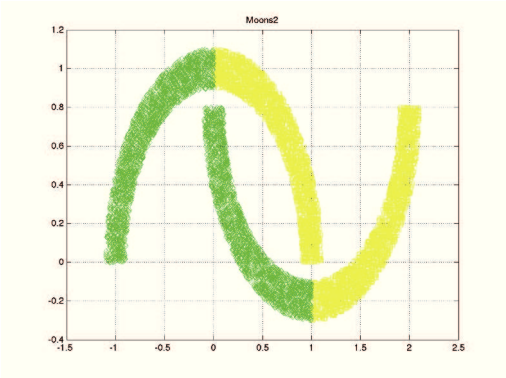

Figure 4.10

4.2 Qualitative Analysis

The qualitative analysis aims at examining the behavior of the proposed query strategy in the simulated datasets. We start by describing the experimental setup for the analysis and follow by the obtained results.

4.2.1 Experimental Setup

The experiment consists in performing the ISE-based Query Strategy in each simulated dataset. One starts with a pool of observations and an empty sample. The query strategy then selects observations one by one from the pool to the sample.

The pool is initially set with all 10.000 observations of the dataset. Although the observations are moved to the sample one by one, one is interested in analyzing snapshots of the current sample for 10, 20 and 100 observations. In this way, one is able to verify whether the selected observations are in agreement with expected behavior of query strategy.

To perform the ISE-based Query Strategy, one needs to set up the kernel bandwidth h for the Kernel Density Estimation method. This parameter was arbitrarily set as $h = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$ for all simulated datasets. The choice of this parameter implicates the estimation error. Thus fixing the parameter for all simulated datasets eliminates any doubt about the provenience of the result.

4.2.2 Selecting 10 observations

By selecting 10 observations with the proposed strategy, we found samples of 10 observations that are very representative for the full dataset. These selected observations are marked in red on the original datasets.

4.2.2.1 Simple datasets

One should note that the selected observations provide the position and the extent of the Gaussian distributions in the datasets. In Figure 4.12, there is one observation at the mean of the Gaussian, providing their position, and other observations at the extremes of the Gaussian, providing the extent of the deviation in each dimension. Figure 4.11, Figure 4.13, and Figure 4.14 do not illustrate it so well, as the Gaussian distributions are not properly represented by the selected observations. This is probably due to the number of observations in the selection or the error incurred in the probability estimation. However, one is still able to realize that the selection remains quite representative of the datasets.

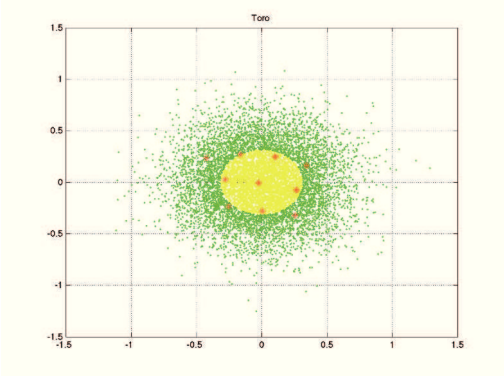

Figure 4.11

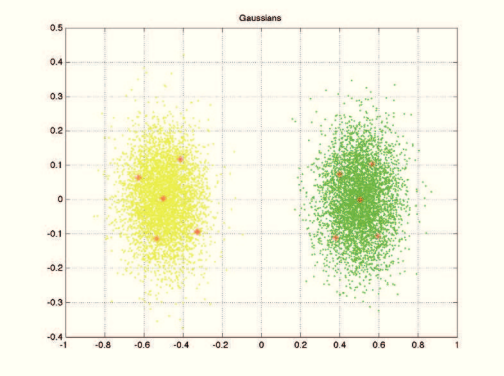

Figure 4.12

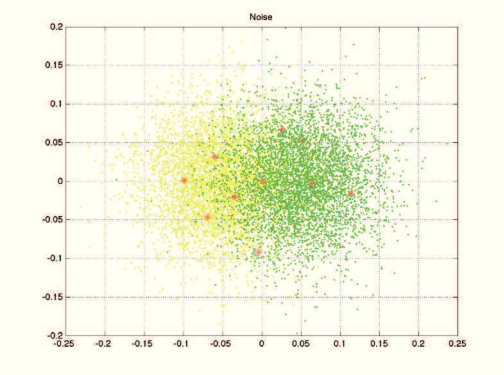

Figure 4.13

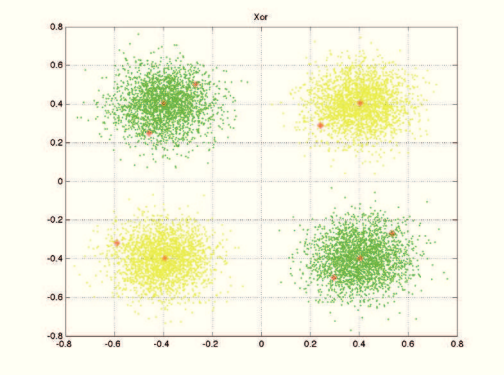

Figure 4.14

4.2.2.2 Clustered datasets

Here, one should note that each Gaussian contains at least one representative selected observation, see for instance Figure 4.17 and Figure 4.18. One can verify that observations from the same Gaussian distribution are selected in such way to provide the position of the mean and the extent of the distributions in each dataset, see Figure 4.15, Figure 4.16, Figure 4.17, and Figure 4.18.

Interestingly, the proportion of the number of observations generated by each Gaussian distribution was preserved in the sample. In Figure 4.15 and Figure 4.16 one clearly

sees that the number of the selected observations in each Gaussian obeys the proportion of a quarter as in the original datasets.

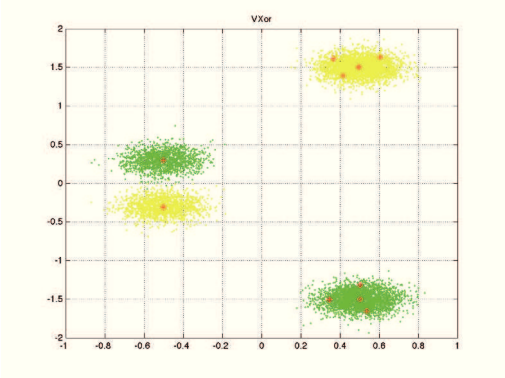

Figure 4.15

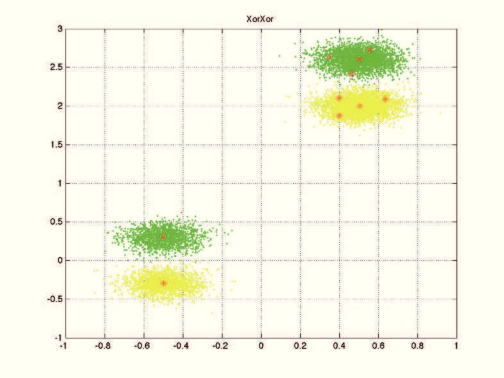

Figure 4.16

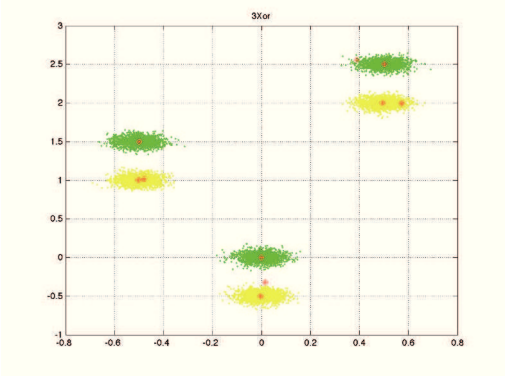

Figure 4.17

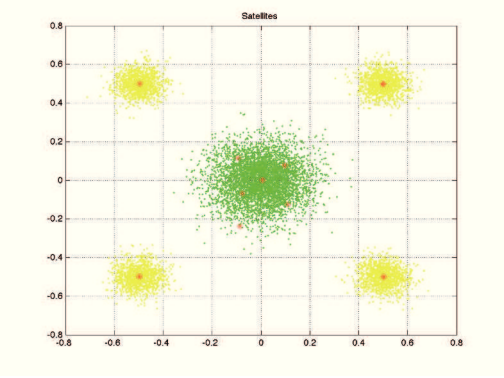

Figure 4.18

4.2.2.3 Non-convex datasets

Here, the selected observations seems to present the shape of the original dataset.

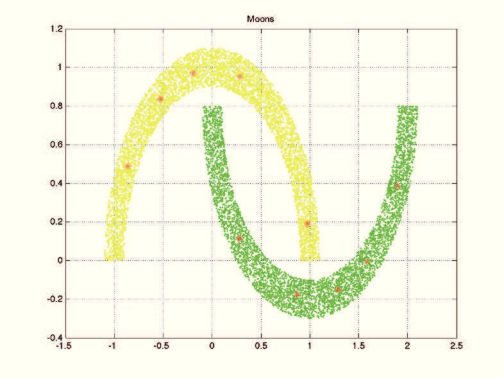

Figure 4.19

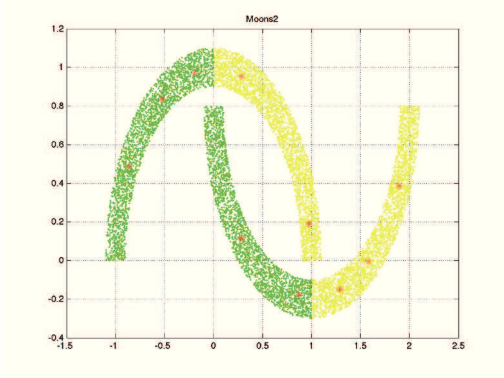

Figure 4.20

4.2.3 Selecting 20 observations

The samples of 20 observations selected by the proposed query strategy are extremely representative of its correspondent datasets. These samples are depicted in red on the original datasets.

4.2.3.1 Simple datasets

Here, one confirms that the proposed query strategy selects observations, which clearly represent the shape of the original pool. Now, even in the last dataset (Figure 4.24), the Gaussian distributions have their shapes well depicted.

Interestingly, the selected observations are equally far way one another according to the density of observations in the pool. This provides evidences that the distributions of the pool and the sample are getting closer.

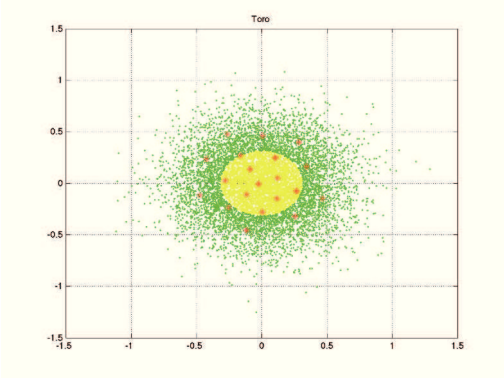

Figure 4.21

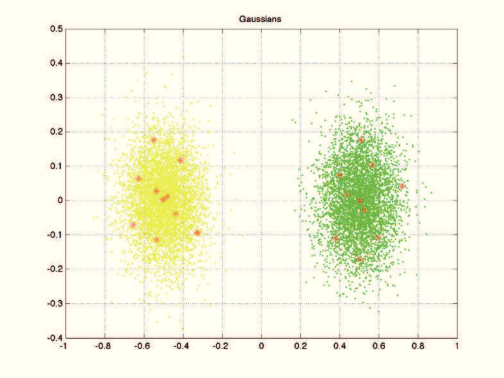

Figure 4.22

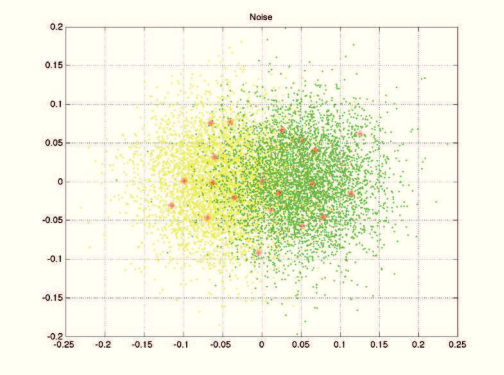

Figure 4.23

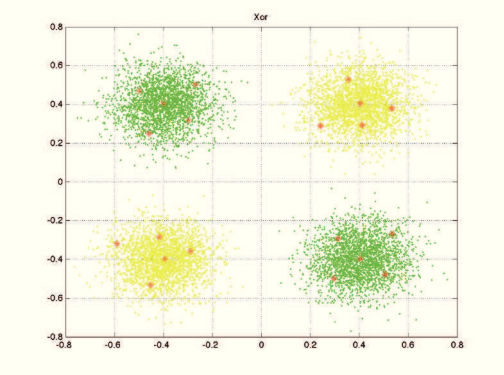

Figure 4.24

4.2.3.2 Clustered datasets

In these datasets, the addition of 20 observations by the proposed query strategy clearly improved the representativeness of the selection for the datasets. The shapes of the Gaussian distributions are slightly better than the sample of 10-sized, providing a better representative sample of the pool.

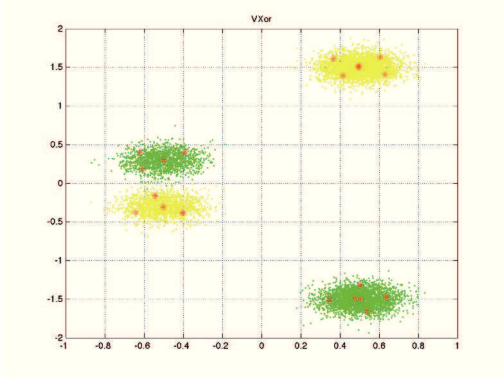

Figure 4.25

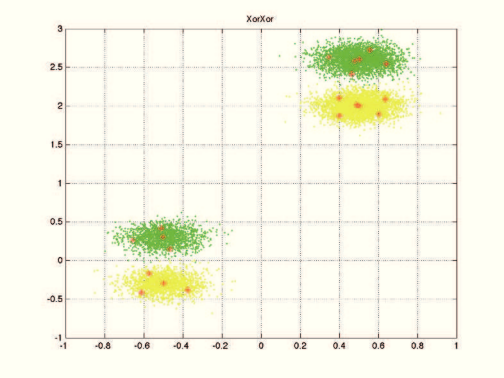

Figure 4.26

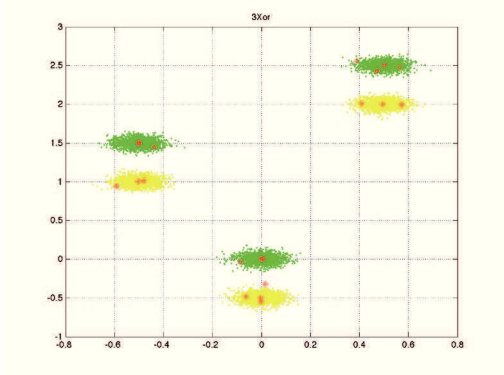

Figure 4.27

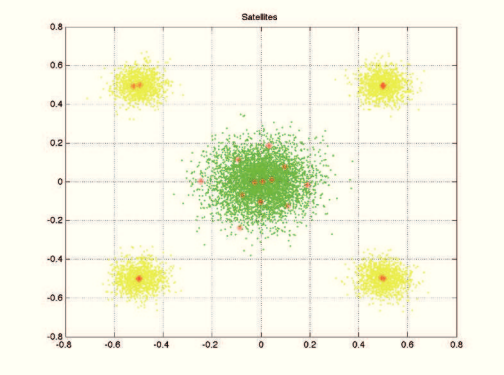

Figure 4.28

4.2.3.3 Non-convex datasets

In the non-convex datasets, the addition of 20 observations by the query strategy improves even further the shape of the clusters. In Figure 4.29 and Figure 4.30 illustrates the 'two moons', where their arcs clearly depicted by the selected observations.

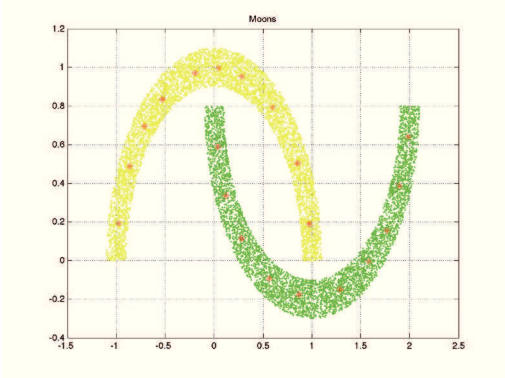

Figure 4.29

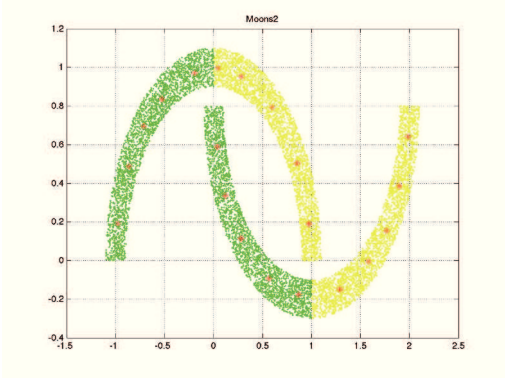

Figure 4.30

4.2.4 Selecting 100 observations

Now, the proposed strategy selects 100 observations from the pool. Once more, one should note that these selected observations are still in agreement with the shape of the original dataset.

4.2.4.1 Simple datasets

Now, one should note that, in addition to the ability of reproducing with a small selection of observations the position and the shape of original clusters, now the set of selected observations starts to reveal the frequency distributions of the original datasets. For instance, one notes that there are more selected observations closer to the mean of the Gaussian distribution than on border. This evidences that the more observations the proposed strategy selects, the more similar to the original distribution the selected sample becomes.

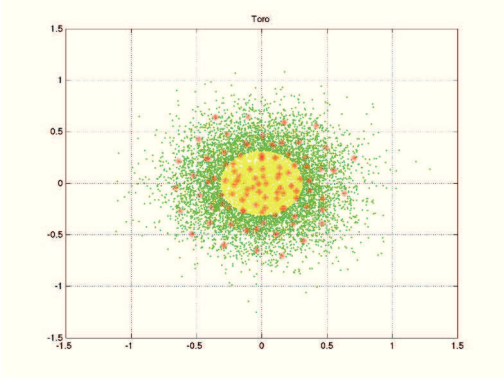

Figure 4.31

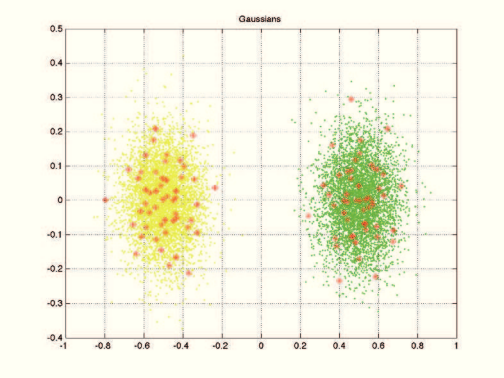

Figure 4.32

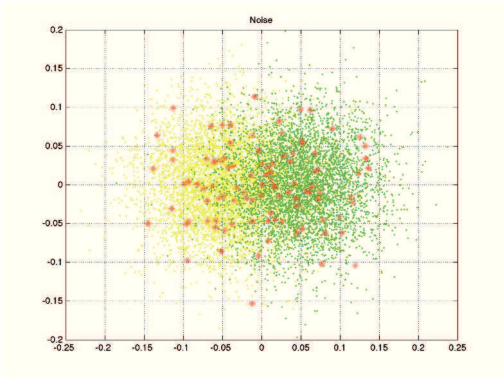

Figure 4.33

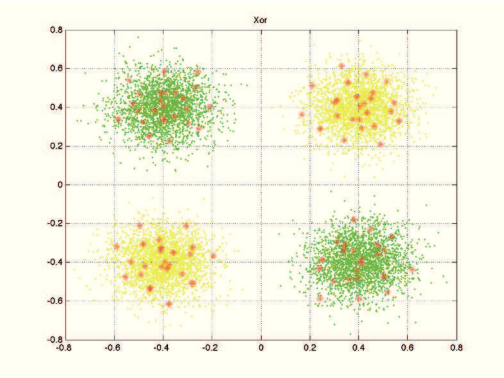

Figure 4.34

4.2.4.2 Clustered datasets

Here, all the clusters are fully represented by the selected observations. We could even perform a clustering algorithm(BISHOP, 2007) and found out the same cluster structure of the original dataset without actually using it.

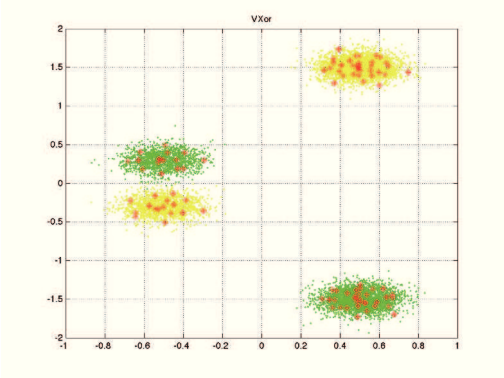

Figure 4.35

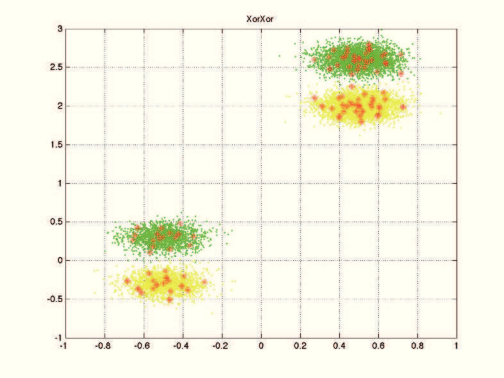

Figure 4.36

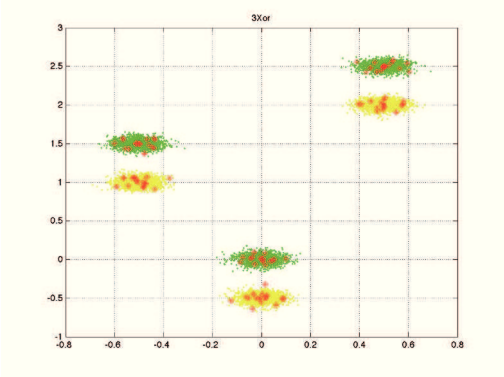

Figure 4.37

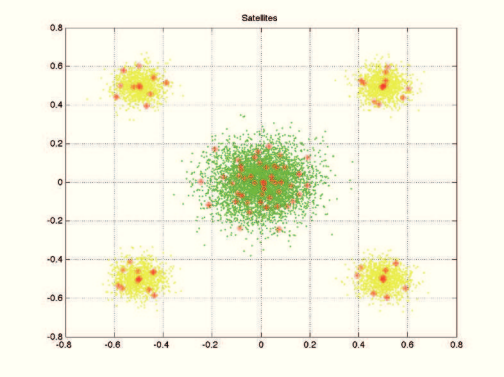

Figure 4.38

4.2.4.3 Non-convex datasets

Here, we can see that the densities become to be well represented. In Figure 4.40 and Figure 4.39 the tight of the moons are represented by selected observations, by allowing for accurate classification.

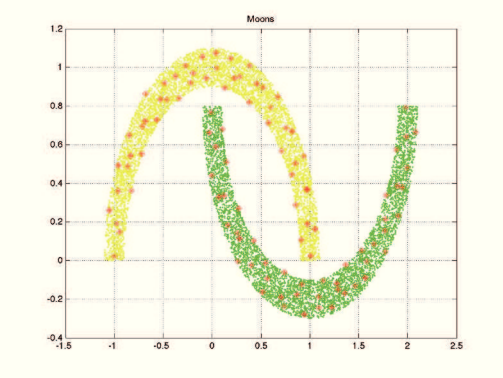

Figure 4.39

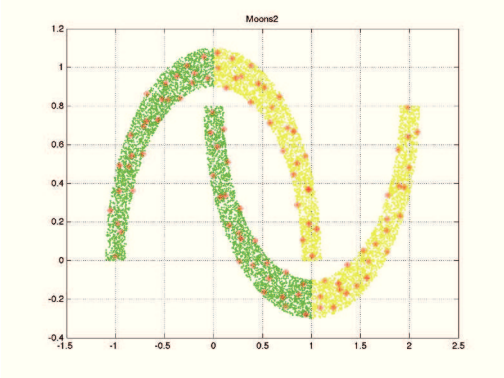

Figure 4.40

4.3 Quantitative Analysis

In this subsection, we present a quantitative analysis of performance comparing the proposed query strategy with passive learning. The goal is to evaluate the classification accuracy along the active learning procedure on both synthetic and real datasets. In this way, one is able to establish a benchmark of accuracy for the proposed query strategy with passive learning as a baseline.

4.3.1 Experimental Setup

The experiment consists in performing the ISE-based Query Strategy in both simulated and real datasets. The goal is to evaluate the prediction accuracy of a supervised model as the number of observations in its training set grows. In this way, a performance comparison between the proposed query strategy and the passive learning may be established.

The pool is initially set with half of the number of observations of the original dataset. The other half of observations is used as test set in order to measure the model accuracy.

The observations are then selected one by one from the pool to the training set by the query strategy. The model is re-trained each time the training set is updated and its accuracy on the test set is computed. This is also performed for the baselines we established for comparing the proposed query strategy.

The experiment produce is described as follows:

Algorithm: Experiment Procedure

Input: D dataset

N maximum size of the training set

Output: accISEQ, accTheBestBaseline, accTheMeanBaseline, accTheWorstBaseline

Set $[P, \text{TestSet}] = \text{split}(D)$; %split the dataset into half for pool and the other for test set

$\text{TrainSet} = \emptyset$;

For i = 1 to N **do**

%ISE-based query strategy selects an observation of the pool

$x_* = \text{ISEQuery}(P)$;

Assigns y_* to x_*;

$TrainSet = TrainSet \cup \{(x_*; y_*)\}$;

%Training the supervised model $\mathcal{M}$ from TrainSet

$\mathcal{M} = train(\text{TrainSet})$;

%Computing the accuracy of the model $\mathcal{M}$ on TestSet

$\text{accISEQ}[i] = \text{accuracy}(\text{TestSet}, \mathcal{M})$;

%Computing the baselines

$[\text{thebest}, \text{themean}, \text{theworst}] = \text{baselines}(P, i, \text{TestSet})$;

$\text{accTheBestBaseline}[i] = \text{thebest}$;

$\text{accTheMeanBaseline}[i] = \text{themean}$;

$\text{accTheWorstBaseline}[i] = \text{theworst}$;

End for

Algorithm 4.1 – Experiment Procedure

The ISE-based Query Strategy is performed with the Kernel covariance matrix arbitrarily set as $h = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}$ for all simulated datasets. For the real datasets, a further discussion is provided shortly.

The experiments were performed by varying the number of observations in the training set from 1 up to 100 observations, *i.e.* $N = 100$.

4.3.1.1 Supervised model

In all experiments, the supervised learning model consists in a Bayesian classifier, where the posterior probability density functions are estimated by the Kernel density estimation (KDE) with multivariate Gaussian kernel.

Although the Bayesian classifier may be simple compared with other models, it does not require tuning so many parameters. Actually, as it uses KDE with multivariate Gaussian kernel for computing the posterior probabilities, one needs to set only the kernel covariance matrix. This parameter is fixed along the experiment and is set with the same covariance matrix used in the KDE to estimate the underlying *pdf* of the pool in the proposed query strategy.

The goal is to evaluate the active learning query strategy, instead the supervised model. Therefore, the supervised model setting is kept fixed along the experiment in order to guarantee as much as possible that the provenience of the results are due to the choice of the training sets. Otherwise, one is not be able to assign the performance to the training sets, generated by the active learners.

4.3.1.2 Performance metrics

For measuring the classification accuracy, a test set $TestSet$ with the same number of observations as the initial pool is randomly selected from the dataset. As the datasets are quite large, this test set allows the estimation of the generalization accuracy of the model $h(x)$, instead of more expensive scheme such as k-fold cross validation. The estimated accuracy on the test set is given by

$$\text{acc} = \sum_{\forall x \in TestSet} \ell\big(h(x), f(x)\big), \qquad \text{Eq. 60}$$

where $\ell\big(h(x), f(x)\big)$ is the 0/1 loss-function

$$\ell\big(\text{h(x)}, \text{f(x)}\big) = \begin{cases} 1 \; if \; h(x) = f(x) \\ 0 \; if \; h(x) \neq f(x) \end{cases}. \qquad \text{Eq. 61}$$

As the observations classes are balanced, we judge unnecessary to use other measures such as the ROC curve once these experiments are time consuming.

4.3.1.3 Baselines

In order to establish baselines of accuracy for the proposed query strategy, we opted for using the following procedure.

For each one observation selected from the pool to the training set by the ISE-based Query Strategy, one randomly draws 1.000 sets of observations with the same size of the current training set. These sets are then used for training an ensemble of 1.000 classifiers and accuracy rate of each classifier is also computed.

Thus, for each size of training set, there will be the accuracy rate of 1.000 classifiers associated with equal number of training sets.

Metrics from each ensemble of classifiers are defined as baselines:

1) *the mean baseline* takes the average of accuracy of all classifiers in the ensemble, being fair empirical generalization accuracy for the classifier;
2) *the worst baseline* takes the worst accuracy of all classifiers in the ensemble, providing a fair empirical lower bound of performance; and
3) *the best baseline* takes the best accuracy of all classifier in the ensemble, providing a fair empirical upper bound of performance

The procedure for these baselines is described as follows:

Algorithm Computing Baselines

Input: P pool

n number of observations in the training set

$TestSet$ test set

Output: MeanBaseline, BestBaseline, WorstBaseline

For i = 1 **to** 1000 **do**

Draw at random n observations from P and add into TrainSet;

Label all observations in TrainSet;

$\mathcal{M} = \text{train}(\text{TrainSet})$;

$acc[i] = \text{accuracy}(\text{TestSet}, \mathcal{M})$;

End for

MeanBaseline = $\text{mean}(acc)$;

BestBaseline = $\max(acc)$;

WorstBaseline = $\min(acc)$.

Algorithm 4.2 – Computing Baselines

The key idea of these baselines is to exhaustively exploit the possible training sets one may obtain from the pool, instead of comparing with several query strategies of the literature. For instance, the best existent query strategy in the literature cannot be much better than a random exhaustive search in the sample space, *i.e.*, the best baseline. Moreover, the mean baseline represents the theoretical lower bound of performance of the proposed query strategy.

4.3.2 Results in the simulated datasets

In this subsection, the results obtained from the experiments on the simulated datasets are discussed.

The results clearly show the advantage of the ISE-based query strategy over the worst case and the average case in all synthetic datasets. This confirms the idea that our selection criterion is less subject to either sampling error or bias, as it yielded classification accuracies nearly close to the best possible accuracy during all the experimental procedure.

The results were not so good for proposed strategy along the 10-first observations in the dataset of Figure 4.41. This happened due to the label classes of the observations are unknown in the 10-first observations. In fact, though the proposed query strategy reduces the variance on the input space variables, for the output variable it is still there. In other words, the proposed query strategy takes around 10 observations before starts to select observations with labels in the outer border. However, after it starts to select them, the classification accuracy fast grows for a value close to the best case.

The results are depicted in the figures as follows:

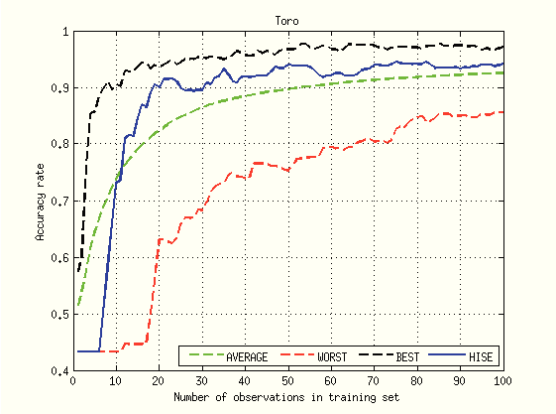

Figure 4.41

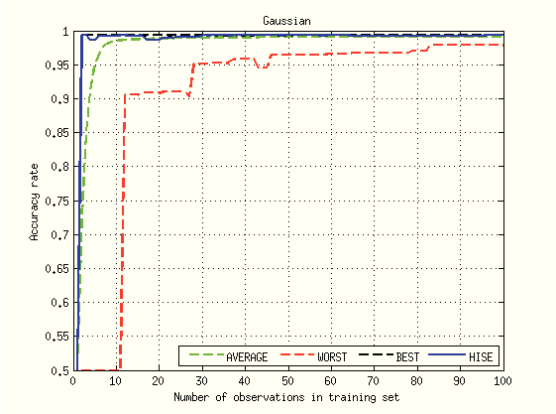

Figure 4.42

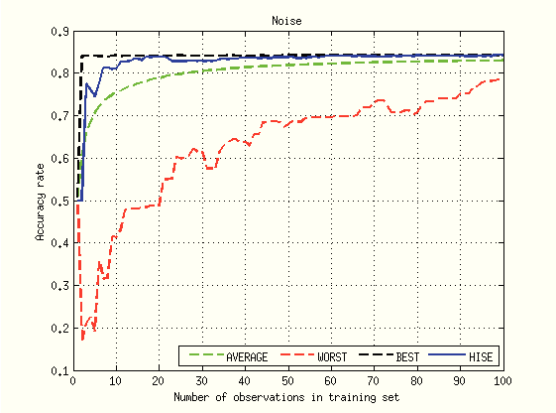

Figure 4.43

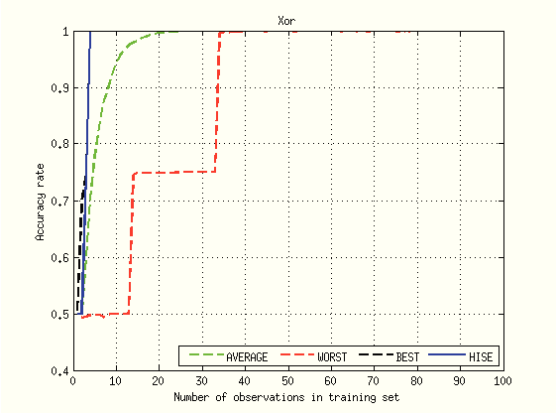

Figure 4.44

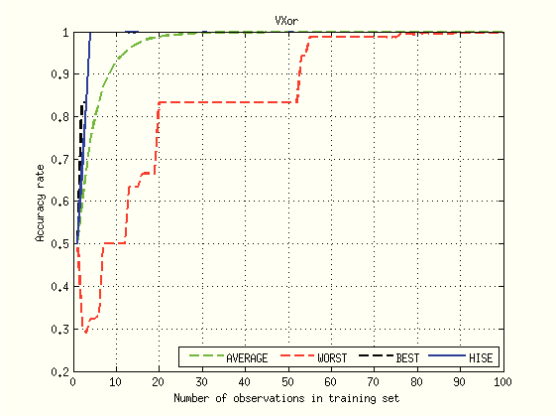

Figure 4.45

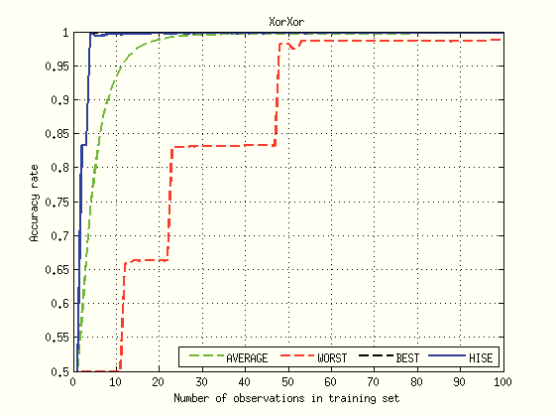

Figure 4.46

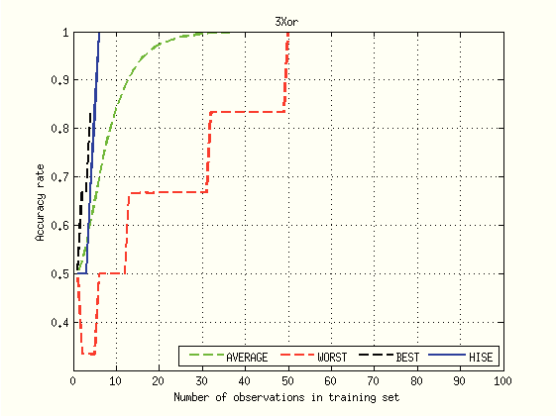

Figure 4.47

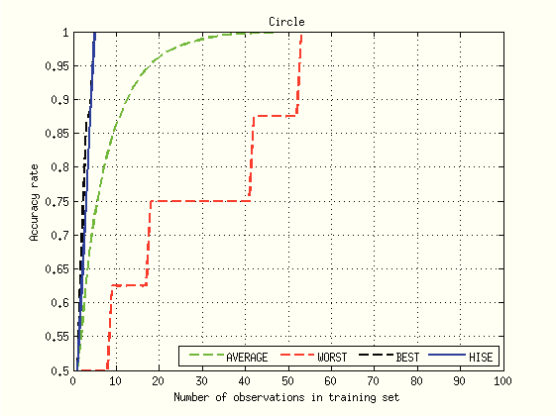

Figure 4.48

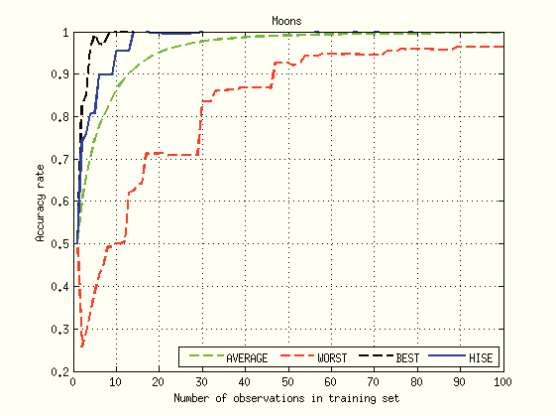

Figure 4.49

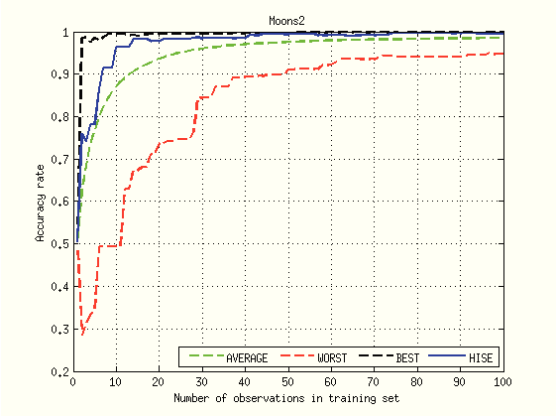

Figure 4.50

4.3.3 Results in real datasets

In this subsection, real datasets are considered for the evaluation of the proposed active learner.

The datasets known as ALEX and IBN_SINA are public available datasets for download. Both ones are part of the Active Learning challenge launched by Pascal2 challenges(GUYON *et al.*, 2011).

ALEX is a dataset for binary classification containing 11 features. In this dataset, there are 5.000 observations for the pool and 5.000 observations for the test set.

IBN_SINA is a handwriting recognition dataset formatted in a feature representation of 92 variables divided into two classes(FARRAHI MOGHADDAM *et al.*, 2010). It is used 10.361 observations for the pool and 10.361 observations for the test set.

Different from the simulated datasets, the number of observations in the training set was increased up to 1.000 for ALEX and up to 500 for the IBN_SINA. These amounts were set up by empirical evaluation. Besides that, the parameter of the Gaussian kernel was manually tuned in order to provide a good estimation of the underlying pdf of the pool.

The results are depicted in Figure 4.51 and Figure 4.52. As one should note, the ISE-based query strategy reached accuracy very close to the best-case baseline, especially of the IBN_SINA. Thus, our method has proved to work well in both real datasets in large dimensionalities.

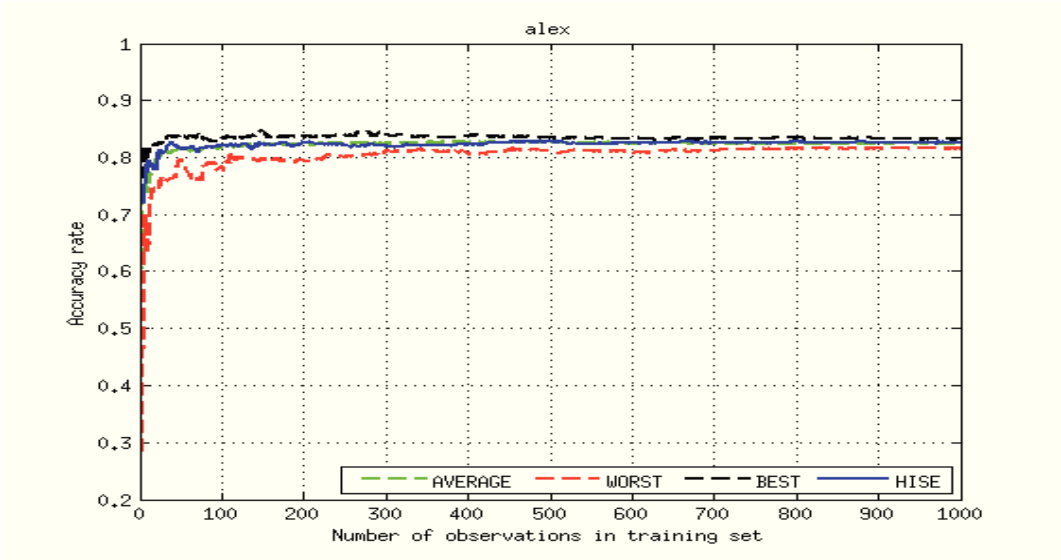

Figure 4.51

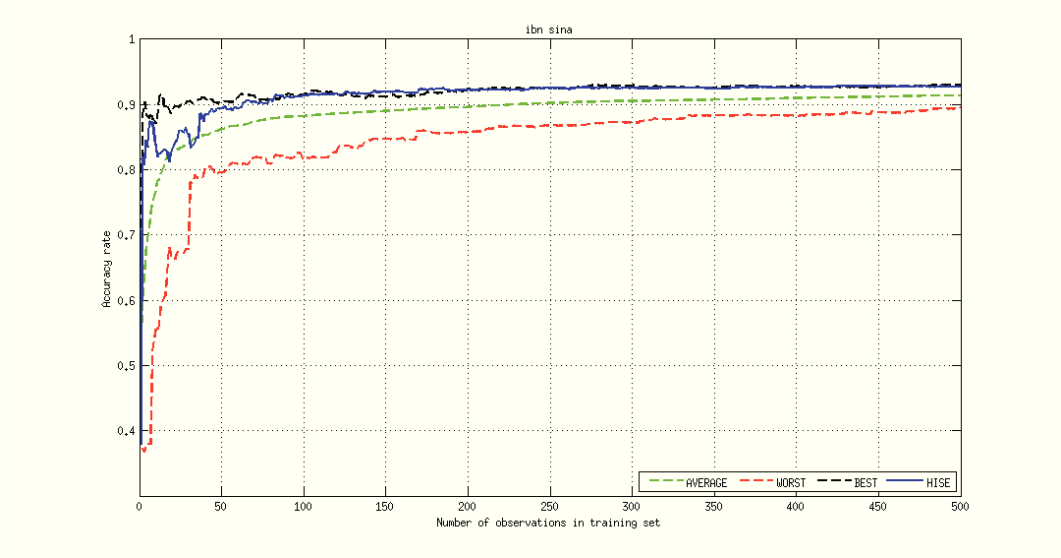

Figure 4.52

4.4 Summary and Conclusions

In this chapter, we presented the experiments performed in order to evaluate the proposed query strategy by performing and comparing the ISE-based query strategy with baselines based on the passive learning.

The results in both qualitative and quantitative analyses are in agreement with the theoretical foundation of the proposed query strategy. The proposed query strategy was able to provide training sets from which accurate models were learned.

The experiments performed in real datasets also presented better results in favor to the ISE-based query strategy. In these datasets, one notes that the proposed strategy is able to handle high dimensionality spaces.

Chapter 5 **Conclusion**

5.1 Summary and Discussion

In this thesis, the active learning issue has been studied and a novel active learning query strategy has been proposed.

The majority of existent active learning strategies in the literature consist in greedy heuristics. These select unlabeled observations of a pool in order to maximize (or minimize) some utility function based on assumptions about either data or the supervised model. Consequently, the training sets generated by these procedures may have the underlying probability distributions different from the population, since the observations are not independent and identically distributed. Therefore, an active learning query strategy is in fact a biased sampling procedure, which systematically favors observations among others according to its selection criterion.

Although many active learning query strategies perform successfully in several scenarios, there is always an inherent risk to fail associated with the choice of the selection criterion. As this is a heuristic based on assumptions about the played scenario, whenever such assumptions do not hold, the query strategy may obtain very poor training sets. This occurs because the generated training sets are little representative of the population distribution as these are generated by a biased sampling procedure.

In this context, the main hypothesis of thesis concerns the bias introduced in the training set. The key idea consists in selecting the most representative observations of the underlying distribution of the pool in order to reduce as much as possible the amount of bias in the training set. In this way, a general query strategy is developed in order to tackle such goal.

The general query strategy proposed in this thesis aims at keeping the probability distributions of the sample and the pool as close as possible. A general procedure is defined for that, in which the key idea consists in measuring, for each candidate observation, the distance between the estimated probability distribution of the sample (i.e. the training set) and the estimated probability distribution of the initial pool. An

information-theoretical framework has been used to handle the probability density estimation and the distance measure between probability density functions (pdf).

A specific query strategy based on the proposed general procedure has been developed, namely ISE-based Query Strategy. This strategy uses the Integrated Squared Error (ISE) as a distance measure between pdfs. This measure allows the development of an analytical expression to the selection heuristic, then providing a tuned algorithm implementation of the general procedure.

A theoretical discussion is provided about the proposed query strategy, resulting in a theoretical lower bound of performance upon the passive learning. Thus, the proposed heuristic is statistically guaranteed of performing better than the passive learning. This means that the variance of the estimators of the supervised learning model is smaller than those generated by passive learning.

In order to evaluate the proposed query strategy, experiments were conducted with the ISE-based Query Strategy in simulated and real datasets. Such experiments performed both a qualitative and quantitative analysis, providing an investigation of the behavior of the proposed query strategy. Baselines were built by carrying on a random exhaustive search in the sample space, in order to establish empirical upper, average, and lower bounds of performance. The results in both qualitative and quantitative analyses have shown favorable performance to the proposal. Moreover, the proposed query strategy has outperformed the average baseline and has been close to the upper bound baseline along almost all the experiment in both simulated and real datasets.

The main disadvantage of the proposed query strategy based on the ISE is to handle the estimation of pdfs. The Kernel Density Estimation (KDE) method requires adjusting the kernel bandwidth, which is tough task. As the dimensionality grows, the bandwidth tuning becomes harder.

Therefore, the general query strategy proposed in this thesis presented both theoretical and empirical advantages.

5.2 Future work

The proposed query strategy of this thesis may be exploited in several different ways. These are some of them:

- Developing a new family of unbiased query strategies by using different distance measures available in the literature of information theory;
- Developing specific query strategies for parametric probability models such as Hidden Markov Models(BISHOP, 2007);
- Developing hybrid strategies mixing the proposed strategy with others;
- Developing a stopping criterion based on the distance between the pdfs of the sample and the population;
- Adapting the proposed general query strategy for other frameworks different from the statistical one. For instance, to consider using a quad-tree(CORMEN *et al.*, 2009) for fast computing densities;
- Developing a data compression algorithm based on the ISE-based Query Strategy; and
- Exploiting different applications, where either active learning or sampling design is required.

Bibliography

ABE, N., MAMITSUKA, H., 1998. "Query Learning Strategies Using Boosting and Bagging". In: *Proceedings of the Fifteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 1998. pp. 1–9.

AMATRIAIN, X., PUJOL, J.M., TINTAREV, N., et al., 2009. "Rate it again: increasing recommendation accuracy by user re-rating". In: *Proceedings of the third ACM conference on Recommender systems*. New York, NY, USA: ACM. 2009. pp. 173–180.

ANGLUIN, D., 1988, "Queries and concept learning". In: *Machine Learning*. v. 2, n. 4, pp. 319–342.

ATTENBERG, J., PROVOST, F., 2010. "Why label when you can search?: alternatives to active learning for applying human resources to build classification models under extreme class imbalance". In: *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM. 2010. pp. 423–432.

BALDI, P., BRUNAK, S., 2001, *Bioinformatics: The Machine Learning Approach, Second Edition (Adaptive Computation and Machine Learning)*. 2. S.l., The MIT Press.

BALDRIDGE, J., OSBORNE, M., 2004. "Active Learning and the Total Cost of Annotation". In: LIN, Dekang & WU, Dekai (eds.), *Proceedings of EMNLP 2004*. Barcelona, Spain: Association for Computational Linguistics. July 2004. pp. 9–16.

BISHOP, C.M., 2007, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. 1st ed. 2006. Corr. 2nd printing 2011. S.l., Springer.

BLOODGOOD, M., VIJAY-SHANKER, K., 2009. "A method for stopping active learning based on stabilizing predictions and the need for user-adjustable stopping". In: *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2009. pp. 39–47.

CARENINI, G., SMITH, J., POOLE, D., 2003. "Towards more conversational and collaborative recommender systems". In: *Proceedings of the 8th international conference on Intelligent user interfaces*. New York, NY, USA: ACM. 2003. pp. 12–18.

CARLSON, A., BETTERIDGE, J., WANG, R.C., et al., 2010. "Coupled semi-supervised learning for information extraction". In: *Proceedings of the third ACM*

international conference on Web search and data mining. New York, NY, USA: ACM. 2010. pp. 101–110.

COHN, D., ATLAS, L., LADNER, R., 1994, "Improving Generalization with Active Learning". In: *Machine Learning*. v. 15, n. 2, pp. 201–221.

COHN, D.A., 1996, "Neural Network Exploration Using Optimal Experiment Design". In: *Neural Networks*. v. 9, n. 6, pp. 1071 – 1083.

COHN, D.A., GHAHRAMANI, Z., JORDAN, M.I., 1996, "Active Learning with Statistical Models". In: *J. Artif. Intell. Res. (JAIR)*. v. 4, pp. 129–145.

CORMEN, T.H., LEISERSON, C.E., RIVEST, R.L., et al., 2009, *Introduction to Algorithms, Third Edition*. 3rd. S.l., The MIT Press.

CULOTTA, A., MCCALLUM, A., 2005. "Reducing labeling effort for structured prediction tasks". In: *Proceedings of the 20th national conference on Artificial intelligence - Volume 2*. Pittsburgh, Pennsylvania: AAAI Press. 2005. pp. 746–751.

DASGUPTA, S., 2009. "The Two Faces of Active Learning". In: *Proceedings of the 12th International Conference on Discovery Science*. Berlin, Heidelberg: Springer-Verlag. 2009. pp. 35–35.

DASGUPTA, S., HSU, D., 2008. "Hierarchical sampling for active learning". In: *Proceedings of the 25th international conference on Machine learning*. New York, NY, USA: ACM. 2008. pp. 208–215.

DUDA, R.O., HART, P.E., STORK, D.G., 2000, *Pattern Classification*. 2. S.l., Wiley-Interscience.

FARRAHI MOGHADDAM, R., CHERIET, M., ADANKON, M.M., et al., 2010. "IBN SINA: a database for research on processing and understanding of Arabic manuscripts images". In: *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*. New York, NY, USA: ACM. 2010. pp. 11–18.

FREUND, Y., SCHAPIRE, R.E., 1997, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting". In: *Journal of Computer and System Sciences*. v. 55, n. 1, pp. 119–139.

FREUND, Y., SEUNG, H.S., SHAMIR, E., et al., 1997, "Selective Sampling Using the Query by Committee Algorithm". In: *Machine Learning*. v. 28, n. 2-3, pp. 133–168.

FUJII, A., TOKUNAGA, T., INUI, K., et al., 1998, "Selective sampling for example-based word sense disambiguation". In: *Computational Linguistics*. v. 24, n. 4, pp. 573–597.

GEMAN, S., BIENENSTOCK, E., DOURSAT, R., 1992, "Neural networks and the bias/variance dilemma". In: *Neural Computation*. v. 4, n. 1 (Jan.), pp. 1–58.

GUO, Y., GREINER, R., 2007. "Optimistic active learning using mutual information". In: *Proceedings of the 20th international joint conference on Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 2007. pp. 823–829.

GUYON, I., CAWLEY, G.C., DROR, G., et al., 2011, "Results of the Active Learning Challenge.". In: *Journal of Machine Learning Research - Proceedings Track*. v. 16, pp. 19–45.

HAN, J., KAMBER, M., PEI, J., 2006, *Data Mining: Concepts and Techniques, Second Edition (The Morgan Kaufmann Series in Data Management Systems)*. 2. S.l., Morgan Kaufmann.

HARPALE, A.S., YANG, Y., 2008. "Personalized active learning for collaborative filtering". In: *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. Singapore, Singapore: ACM. 2008. pp. 91–98.

HOI, S.C.H., JIN, R., ZHU, J., et al., 2006. "Batch mode active learning and its application to medical image classification". In: *Proceedings of the 23rd international conference on Machine learning*. New York, NY, USA: ACM. 2006. pp. 417–424.

IYENGAR, V.S., APTE, C., ZHANG, T., 2000. "Active learning using adaptive resampling". In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM. 2000. pp. 91–98.

JENSSEN, R., PRINCIPE, J., ERDOGMUS, D., et al., 2006, "The Cauchy–Schwarz divergence and Parzen windowing: Connections to graph theory and Mercer kernels". In: *Journal of the Franklin Institute*. v. 343, n. 6 (Sep.), pp. 614–629.

JURAFSKY, D., MARTIN, J.H., 2008, *Speech and Language Processing*. 2. S.l., Pearson Prentice Hall.

KAPOOR, A., HORVITZ, E., BASU, S., 2007. "Selective supervision: guiding supervised learning with decision-theoretic active learning". In: *Proceedings of the 20th international joint conference on Artificial intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 2007. pp. 877–882.

KING, R.D., WHELAN, K.E., JONES, F.M., et al., 2004, "Functional genomic hypothesis generation and experimentation by a robot scientist". In: *Nature*. v. 427, n. 6971, pp. 247–252.

KÖRNER, C., WROBEL, S., 2006. "Multi-class Ensemble-Based Active Learning". In: FÜRNKRANZ, Johannes, SCHEFFER, Tobias & SPILIOPOULOU, Myra (eds.), *Machine Learning: ECML 2006*. Berlin, Heidelberg: Springer Berlin Heidelberg. Lecture Notes in Computer Science. pp. 687–694.

KULLBACK, S., LEIBLER, R.A., 1951, "On Information and Sufficiency". In: *The Annals of Mathematical Statistics*. v. 22, n. 1, pp. 79–86.

LEWIS, D.D., CATLETT, J., 1994. "Heterogeneous uncertainty sampling for supervised learning". In: COHEN, William W. & HIRSH, Haym (eds.), *Proceedings of the 11th International Conference on Machine Learning*. New Brunswick, US: Morgan Kaufmann Publishers, San Francisco, US. 1994. pp. 148–156.

LEWIS, D.D., GALE, W.A., 1994. "A sequential algorithm for training text classifiers". In: *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: Springer-Verlag New York, Inc. 1994. pp. 3–12.

LOMASKY, R., BRODLEY, C.E., AERNECKE, M., et al., 2007. "Active Class Selection". In: *Proceedings of the 18th European conference on Machine Learning*. Berlin, Heidelberg: Springer-Verlag. 2007. pp. 640–647.

MACKAY, D.J.C., 1992, "Information-based objective functions for active data selection". In: *Neural Computation*. v. 4, n. 4 (Jul.), pp. 590–604.

MANNING, C.D., SCHÜTZE, H., 1999, *Foundations of statistical natural language processing*. Cambridge, MA, USA, MIT Press.

MCCALLUM, A., NIGAM, K., 1998. "Employing EM and Pool-Based Active Learning for Text Classification". In: *Proceedings of the Fifteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 1998. pp. 350–358.

MELVILLE, P., MOONEY, R.J., 2004. "Diverse ensembles for active learning". In: *Proceedings of the twenty-first international conference on Machine learning*. New York, NY, USA: ACM. 2004. pp. 74–.

MELVILLE, P., YANG, S.M., SAAR-TSECHANSKY, M., et al., 2005. "Active learning for probability estimation using jensen-shannon divergence". In: *Proceedings of the 16th European conference on Machine Learning*. Berlin, Heidelberg: Springer-Verlag. 2005. pp. 268–279.

MINTZ, M., BILLS, S., SNOW, R., et al., 2009. "Distant supervision for relation extraction without labeled data". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2009. pp. 1003–1011.

MITCHELL, T.M., 1997, *Machine Learning*. 1. New York, NY, USA, McGraw-Hill, Inc.

MUSLEA, I., MINTON, S., KNOBLOCK, C.A., 2000. "Selective Sampling with Redundant Views". In: *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*. S.l.: AAAI Press. 2000. pp. 621–626.

NGUYEN, H.T., SMEULDERS, A., 2004. "Active learning using pre-clustering". In: *Proceedings of the twenty-first international conference on Machine learning*. New York, NY, USA: ACM. 2004. pp. 623–630.

OLSSON, F., TOMANEK, K., 2009. "An intrinsic stopping criterion for committee-based active learning". In: *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2009. pp. 138–146.

PRINCIPE, J.C., 2010, *Information Theoretic Learning: Renyi's Entropy and Kernel Perspectives*. 1st. S.l., Springer Publishing Company, Incorporated.

RICCI, Francesco, ROKACH, Lior, SHAPIRA, Bracha & KANTOR, Paul B. (eds.), 2010, *Recommender Systems Handbook*. 1st ed. S.l., Springer.

ROY, N., MCCALLUM, A., 2001. "Toward Optimal Active Learning through Sampling Estimation of Error Reduction". In: *Proceedings of the Eighteenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 2001. pp. 441–448.

RUBENS, N., KAPLAN, D., SUGIYAMA, M., 2011. "Active Learning in Recommender Systems". In: RICCI, Francesco, ROKACH, Lior, SHAPIRA, Bracha & KANTOR, Paul B. (eds.), *Recommender Systems Handbook*. Boston, MA: Springer US. pp. 735–767.

SCHEIN, A., UNGAR, L., 2007, "Active learning for logistic regression: an evaluation". In: *Machine Learning*. v. 68, n. 3, pp. 235–265.

SCHOHN, G., COHN, D., 2000. "Less is More: Active Learning with Support Vector Machines". In: *Proceedings of the Seventeenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 2000. pp. 839–846.

SETTLES, B., 2012, "Active Learning". In: *Synthesis Lectures on Artificial Intelligence and Machine Learning*. v. 6, n. 1, pp. 1–114.

SETTLES, B., CRAVEN, M., 2008. "An analysis of active learning strategies for sequence labeling tasks". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2008. pp. 1070–1079.

SHANNON, C.E., 2001, "A mathematical theory of communication". In: *SIGMOBILE Mob. Comput. Commun. Rev.* v. 5, n. 1 (Jan.), pp. 3–55.

SNOW, R., O'CONNOR, B., JURAFSKY, D., et al., 2008. "Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2008. pp. 254–263.

TOMANEK, K., HAHN, U., 2009. "Semi-supervised active learning for sequence labeling". In: *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*. Stroudsburg, PA, USA: Association for Computational Linguistics. 2009. pp. 1039–1047.

TONG, S., KOLLER, D., 2002, "Support vector machine active learning with applications to text classification". In: *J. Mach. Learn. Res.* v. 2, pp. 45–66.

TUR, G., HAKKANI-TÜR, D., SCHAPIRE, R.E., 2005, "Combining active and semi-supervised learning for spoken language understanding". In: *Speech Communication*. v. 45, n. 2, pp. 171–186.

VALIANT, L.G., 1984, "A theory of the learnable". In: *Commun. ACM*. v. 27, n. 11, pp. 1134–1142.

VIJAYANARASIMHAN, S., JAIN, P., GRAUMAN, K., 2013, "Hashing Hyperplane Queries to Near Points with Applications to Large-scale Active Learning". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. v. 99, n. PrePrints.

VLACHOS, A., 2008, "A stopping criterion for active learning". In: *Comput. Speech Lang.* v. 22, n. 3, pp. 295–312.

WALLACE, B.C., SMALL, K., BRODLEY, C.E., et al., 2010. "Active learning for biomedical citation screening". In: *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: ACM. 2010. pp. 173–182.

WASSERMAN, L., 2003, *All of Statistics: A Concise Course in Statistical Inference*. S.l., Springer.

XU, Z., AKELLA, R., ZHANG, Y., 2007. "Incorporating diversity and density in active learning for relevance feedback". In: *Proceedings of the 29th European conference on IR research*. Berlin, Heidelberg: Springer-Verlag. 2007. pp. 246–257.

ZHANG, T., OLES, F., 2000. "A probability analysis on the value of unlabeled data for classification problems". In: *Proc. 17th International Conf. on Machine Learning*. S.l.: s.n. 2000. pp. 1191–1198.

ZHU, X., LAFFERTY, J., GHAHRAMANI, Z., 2003. "Combining Active Learning and Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions". In: *ICML 2003 workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*. S.l.: s.n. 2003. pp. 58–65.