Explicit Utility in Supervised Learning

James L. Carroll
Department of Computer Science
Brigham Young University
jlcarroll@gmial.com

Neil Toronto
Department of Computer Science
Brigham Young University
ntoronto@cs.byu.edu

Robbie Haertel
Department of Computer Science
Brigham Young University
rah67@cs.byu.edu

Kevin D. Seppi
Department of Computer Science
Brigham Young University
kseppi@cs.byu.edu

Abstract

We use a graphical model of the supervised learning problem to explore the theoretical effect of utility in the form of end use and sample cost on supervised learning, No-Free-Lunch, sample complexity, and active learning. There are two sources of utility that can affect the above problems: utility that comes from end use and utility that comes from sample costs. We explore which parts of these problems depend on utility, and which parts are utility free. Further, we propose a novel interpretation of the No-Free-Lunch theorems that is independent of utility. We propose that sample complexity should be redefined in terms of expected sample costs to achieve a given threshold on expected end use effectiveness (which would be defined in terms of end use utility). finally, we explore the effects of the sample cost function and the end use utility function on active learning techniques both theoretically (through the optimal active learning equations) and through several examples including a synthetic data set and a real life part of speech tagging scenario.

1 Preface, by James L. Carroll, 2012

This paper was initially written back in 2008, and accepted to the NIPS 2008 Workshop on Cost-Sensitive Machine Learning. In the nature of workshops, many of the ideas presented herein were preliminary, and our thinking has now shifted somewhat from what was presented here. However, the work was influential, inasmuch as these ideas largely inspired what eventually became Chapters 2-6 of my dissertation [1]. We encourage those seriously interested in these ideas to consult that document, as it contains a slightly different, and more accurate, perspective in some instances.

However, we are still making this document available on our web page, mostly for historical reasons, as it demonstrates the evolution of some of our ideas on these issues. We also wanted to insure that those who might search for the document based upon earlier citations to it will still be able to find it, and know where they should go next.

2 Introduction

Two important ways in which cost affects supervised learning are sampling cost and what we will call “end use.” Sample cost involves the cost of acquiring labeled training examples[2][3][4] while end use involves the way in which the output of the supervised learner will be used to aid in decision making. Researchers do not always know how their algorithms will be used. It makes sense to want to be able to create a supervised learning technique, or active learning sampler that will perform reasonably well over many sample cost functions and over many end uses. However, it has been unclear exactly what parts of the machine learning problem are truly utility independent, and which depend on sample cost and end use.

Traditionally, misclassification error rate, sometimes known as a 0-1 loss function, has been used to characterize end use However, 0-1 loss is not the only end use scenario one might care about, in fact it may be quite rare in practice. For example, any time the utilities for false positives and false negatives are not exactly evenly balanced the 0-1 loss function does not hold.

We will mathematically define the supervised learning problem itself by presenting the formulas for the optimal solutions (with respect to expected utility). Since supervised learning deals explicitly with uncertainty, it involves probability theory. This focus on utility, costs, and probabilities naturally leads to a Bayesian decision theoretic approach. In practice these problems are not normally solved optimally, given the computational complexity of the optimal solution. However, from our perspective, the optimal solutions *are* the mathematical definition of the problems we are trying to solve. Understanding the optimal formulation is equivalent to understanding the mathematical definitions of the supervised learning problem. Such an understanding can be important theoretically, can improve our understanding of why existing heuristics work, when and why they sometimes fail, and can aid in the creation of future heuristics.

We will model learning process as a graphical model, and given those assumptions we will explore the theoretical results that follow. These results will be expressed in terms of a series of theorems and observations. Initially, formal proofs were constructed for each theorem but have been omitted due to space constraints. We hope that the proofs will eventually be published in a forthcoming paper [they can now be found in Carroll's dissertation[1]]. Some of the theorems present new information while others are already known. A few are even trivial, however, many machine learning researchers have ignored or misinterpreted this information, and we believe that presenting these theorems will be a valuable contribution to the understanding of how utility affects supervised learning.

In Section 3 we will present our graphical model of the supervised learning problem. In Section 4 we will explore the implications of end use assumptions on the No-Free-Lunch Theorem, Traditional No-Free-Lunch was proved for a specific utility function, we present a new variation on No-Free-Lunch that holds for all end use utility functions. In Section 5 we explore the parts of the supervised learning problem that can be solved independently of utility and illustrate the need for distributional class probability outputs. In Section 6 we argue that sample complexity should be re-defined to take end use and sample costs into account. We also discuss the sample cost and end use utility assumptions made by active learning, and conclude in Section 7.

3 The Unified Bayesian Decision Theoretic Model (UBDTM)

Supervised learning involves a training set $\mathscr{D}_{train}$ which consists of feature inputs $\mathbf{x}$, class outputs y, and the unknown function that maps them $f : \mathbf{x} \rightarrow y$, and a test set $\mathscr{D}_{test}$ which consists of feature inputs $\mathbf{x}'$, and unobserved class outputs y'. The goal of supervised learning is to determine the posterior predictive, $p(y'|\mathbf{x}', \mathscr{D}_{train})$, that is, the probability of a class y' in the test set given its features $\mathbf{x}'$ and the training set $\mathscr{D}_{train}$.

Given the intuitive relationships among these random variables given in the graphical model of figure 1, then the rules of probability provide an optimal technique for classification Formally:

$$p(y'|\mathbf{x}', \mathscr{D}_{train}) = \int p(y'|\mathbf{x}', f)p(f|\mathscr{D}_{train})df, \qquad (1)$$

where

$$p(f|\mathbf{x}, y) = \frac{p(y|\mathbf{x}, f)p(f)}{\int p(y|\mathbf{x}, f)p(f)df}. \qquad (2)$$

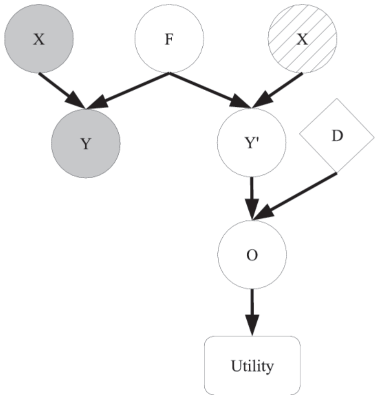


Figure 1: Complete decision theoretic model.

Equation 2 can be used iteratively in order to compute $p(f|\mathscr{D}_{train})$.

We will model end use as: 1. a decision space $\mathbb{D}_D$, 2. an outcome space $\mathbb{D}_O$, 3. a probability function over outcomes given a decision and class $p(o|d, y')$, and 4. a utility function $U(o)$ (See Figure 1). Given the intuitive relationships among these random variables given in the graphical model of figure 1, then the rules of probability and decision theory indicate that the expected utility of a given decision can be found as follows:

$$E[U|d] = \int_{y' \in \mathbb{D}_Y} \int_{o \in \mathbb{D}_O} U(o) p(o|y', d) p(y'|\mathbf{x}', \mathscr{D}_{train}) dy' do, \tag{3}$$

when the outcome is probabilistic, which simplifies to

$$E[U|d] = \int_{y' \in \mathbb{D}_Y} U(y', d) p(y'|\mathbf{x}', \mathscr{D}_{train}) dy', \tag{4}$$

when the outcome given y' and d is deterministic. The optimal decision $d^* \in \mathbb{D}_D$ that will maximize the expected utility can be found as follows:

$$d^* = \underset{d \in \mathbb{D}_D}{\operatorname{argmax}} E[U|d] \tag{5}$$

In the past, researchers have confounded the supervised learning problem (producing the posterior predictive) with its end use (finding the optimal d^* given the posterior predictive). This is easy to do when the decision is simply to select the most likely class. However, in our model, the decision need not even have the same domain as the class. Selection of the posterior predictive and selection of the maximum expected utility decision given the posterior predictive are two distinct portions of the problem. One that does not depend on utility at all (producing the posterior predictive) and another that does (producing the best action given the utility function and the posterior predictive).

A justification of this model for supervised learning as well as an extension of the model to empirical function optimization has been presented elsewhere [5]. In the sections that follow, we take these model assumptions to be axiomatic.

4 Utility, No-Free-Lunch, and *a priori* Distinctions Between Learning Algorithms

No-Free-Lunch asks the question, "how do learning algorithms compare off training set when all possible functions are equally likely." An "off training set" example x' is one that does not appear in the training set $x' \notin \mathscr{D}_{train}^X$. In our model, this is equivalent to asking how different approximations to the Equations 1-5 compare given a uniform prior $p(f)$. Put into the notation of UBDTM, and understood from a Bayesian perspective [5] the NFL theorem can be re-stated as:

Theorem 4.1 (The No-Free-Lunch Theorem)**:** *Given a uniform prior p(f) over all possible discrete finite functions; a decision that consists in selecting $d \in \mathbb{D}_Y$; a misclassification error rate utility function (also called 0-1 loss): $U(y, d) = 1$ if $d = y$, and 0 else; and an off-training set example x', then all decisions d produce the same expected utility. Since all decisions produce the same expected utility, all "algorithms" for selecting a decision also have the same expected utility.*

The proof can be found in Wolpert's publications [6]. Intuitively, for each function f_1 that predicts one class, there exists another function f_2 that predicts a different class [7]. When all functions are equally likely there is no reason to prefer any classification over another.

The uniform prior over f is sometimes called the "*a priori*" case, or the "unbiased learning" case, although this is a misnomer, since a uniform prior is neither *a priori* nor unbiased. The uniform prior is a specific prior and a specific bias.

The NFL property that all algorithms have the same expected utility also holds for some utility functions other than misclassification error rate so long as they do not impose a "geometrical structure" over Y. "If the error function induces a geometrical structure over Y then we can have *a priori* distinctions between learning algorithms" [6]. But in such cases, this existence of *a priori* distinctions between learning algorithms is imposed primarily by the utility function, since all classes are equally likely off training set.

However, since the posterior predictive can be computed in a utility independent manner, it is also possible to re-frame the NFL theorem in a utility free way. Formally:

Theorem 4.2 (The Bayesian NFL Theorem)**:** *Given a uniform prior p(f) over all possible discrete finite functions, and given an off-training set example x', then the optimal posterior predictive $y'|x', \mathscr{D}_{train}$ is uniform. Posterior predictives other than uniform can result in sub optimal expected utility performance for some end uses.*

NFL has traditionally been interpreted in terms of what Theorem 4.1 says about uniform expected utilities for decisions given a specific set of utility functions. However, we believe that it is the new property expressed in Theorem 4.2 that is most significant. Intuitively this theorem says that uniform priors lead to uniform posterior predictives off training set for all utility function. For some utility functions, a uniform posterior predictive and Equations 3-5 can lead to some decisions having greater expected utility than others. In those cases the utility function leads to *a priori* distinctions between learning algorithms. Utility functions that allow *a priori* distinctions between learning algorithms are quite common, so it is important to report the accurate posterior predictive, even if that posterior predictive is uniform [5].

The NFL theorems have often been mis-interpreted to mean that there is no "best" supervised learning algorithm. This is not the case. This error is partly due to the confusion between the supervised learning problem (the problem of producing the posterior predictive) and end use (the problem of using that posterior predictive in decision making). There can be a best posterior predictive solution even when there may not be a best decision for a specific end use. All the NFL proofs really show is that uniform priors lead to uniform posteriors off training set, which can lead to all decisions being equally good for *some*, but not all, utility functions.

The well known Bayesian optimality proofs clearly indicate that there is an optimal or "best" supervised learning algorithm for each potential prior $p(f)$. In the case of the uniform prior, the best supervised learning algorithm is the one that returns the uniform posterior predictive as Equations 1 and 2 demand. And these same equations will also return the optimal posterior predictive for any prior. The challenge of supervised learning should no longer be to design the best algorithm (that has already been done) but to select the best prior for that known best algorithm for each set of problem classes that we might want to solve, and to provide useful heuristics when the optimal solution is intractable.

This represents a *very* different way of thinking about the implications of the NFL theorems than is sometimes encountered.Thus, one reason to pay attention to utility functions other than misclassification error rate is that thinking in this way can affect the theoretical foundations of supervised learning in important ways.

5 End Use and No Free Dinner

In this section we will explore exactly what can and can not be done independent of end use assumptions, and discuss the need to report sufficient statistics on the posterior predictive when end use is unknown. In Section 6 we will expand the need for sufficient statistics to Active Learning.

Theorem 5.1 (Distributional Output Utility Independence)**:** $p(y'|\boldsymbol{x}', \mathscr{D}_{test})$ *can be computed without regard to end use, or sample cost.*

This can be seen from Equation 1. Alternatively, according to the independence rules of graphical models Y' is independent of O, D, and U, when O, D, and U are unobserved. If one of those values was observed, then this observed value would have an impact on Y'. This rather obvious theorem indicates that, when training a classifier, the sample cost and end use, are irrelevant. What is necessary is a training set and a prior over f. Naturally, generating the training set could involve cost considerations, and we will deal with that in greater detail in the Section 6.

Theorem 5.2 (Insufficient Statistics Assume End Use)**:** *There exists a utility function U^s, for which the expected utility will be sub-optimal if a classifier reports a non-sufficient statistic for the posterior predictive $p(y'|\mathbf{x}', \mathscr{D}_{test})$.*

Discussion. Theorem 5.1 indicates that end use concerns are irrelevant for computing the posterior predictive distribution for y', but makes no such guarantee when reporting a point estimate or other insufficient statistic on the posterior predictive. Theorem 5.2 indicates that it is impossible to construct a Bayes optimal decision for all utility functions using an insufficient statistic for the posterior predictive. This is mathematically unsurprising, but has important implications for supervised learning theory. This means that summaries of the posterior predictive cannot safely be made independent of some utility assumptions. Accurately reporting uncertainty is essential for cases when the utility is unknown. Even in the case where the function f is known to be a deterministic mapping between $\mathbf{x}' \rightarrow y'$, uncertainty about F should still lead to a probabilistic output over y'. Failure to take this into account can lead to many problems including overfit [5]. The most common summary of the posterior predictive is to report the most likely class. This summary assumes a utility function of 0-1 loss, which is commonly not the actual utility function that the end user will employ when using a supervised learning algorithm. If we want to produce algorithms that are truly flexible, and which can be used in multiple end use scenarios, it is essential to report full posterior predictives. This theoretical result will hopefully lead to increased interest in classifiers that accurately report their uncertainty.

Theorem 5.3 (Insufficient Statistical Summaries Given End Use)**:** *Optimality can be maintained with decisions or with some other insufficient statistics of the posterior predictive given end use, or a distribution over possible end uses.*

Discussion. If the end use of the algorithm is known, then useful summaries can often be made. For example, decisions are themselves actually summary statistics on the posterior predictive [8]. And for most distributions they will be insufficient statistics. Yet these decisions can still be optimal for the given utility function if they were made according to Equation 5. It is also possible to select an optimal decision or summary statistic given a distribution over possible end uses. For example, if the end use utility function is unknown, but a distribution over possible utility functions exists $p(f_u)$, then the optimal decision can be computed by expecting over the unknown utility function:

$$E[U|d] = \int_{f_u \in \mathbb{D}_U} \int_{y' \in \mathbb{D}_Y} \int_{o \in \mathbb{D}_O} f_u(o) p(o|y', d) p(y'|\mathbf{x}', \mathbf{x}, y) p(f_u) do\, dy' df_u. \tag{6}$$

Similar approaches can be taken when other elements of end use (decision space, outcome space, outcome distribution etc) are unknown, so long as a distribution over possible end uses exists.

Again, this result is unsurprising from a mathematical perspective, but its implications for machine learning have been under-appreciated. It is well known in statistics that summaries can depend on the loss function (statisticians are pessimistic creatures and tend to think in terms of loss, the inverse of the more intuitive utility). For example, using the mean can be justified by the mean squared error loss function, while using the median can be justified by the absolute-difference loss function. However, in machine learning, we have been slow to recognize that not reporting full posterior predictives constitutes implicit end use assumptions which are often not met in practice.

The next natural question is, what happens when we want to summarize or approximate the posterior predictive when the end use is unknown. One approach, similar to that taken by the No-Free-Lunch theorems, would be to assume that all possible end uses are equally likely.

Theorem 5.4 (No Free Dinner)**:** *If the distribution over possible utility functions is uniform, then all decisions have the same expected utility, and the decision is independent of the posterior predictive and so, independent of any statistic or summary of the posterior predictive.*

Discussion. The No-Free-Lunch Theorem (Theorem 4.1) says that for a uniform distribution $p(f)$, and a specific end use, all decisions d have the same expected utility off training set. The Bayesian No-Free-Lunch Theorem (Theorem 4.2) says that for a uniform distribution $p(f)$, regardless of end use, the posterior predictive is uniform off training set. The No Free Dinner theorem is much stronger than the previous two. It says that for a uniform distribution over possible end uses, all decisions d have the same expected utility regardless of the prior $p(f)$ (machine learning bias), regardless of the posterior predictive $p(y'|\mathbf{x}', \mathscr{D}_{train})$, and regardless of whether $\mathbf{x}'$ is on or off training set. This means that if all utility functions are equally likely, then the expected utility of all classification techniques is the same. In this case, there really is no "best" supervised learner, something that wasn't really true for traditional NFL. However, even in this case, if the utility function will become available at the time of use, then a best classification technique exists, and that technique is to return the full posterior predictive. The posterior predictive can be used after the utility function has become available to find a decision that maintains optimality.

Supervised learning research often involves creating algorithms that will be used in many different ways, and the end use is often unknown at the time of algorithm design. Theorem 5.2 and 5.4 together indicate that summaries of the posterior predictive should not be made before end use is known. This has broad implications for the many machine learning approaches that report MAP estimates, ML estimates, or any other insufficient statistic of the posterior predictive. It means that they are either making some implicit utility assumptions, or they are producing sub optimal approximations to the supervised learning problem. Although heuristic approximations are often necessary, heuristics that report full posterior predictives are more likely to be closer to optimal and provide flexibility that can be useful later when the end use becomes known. Such approximations are therefore more desirable than heuristics that report point estimate summaries. Reporting full posterior predictives is also important in active learning as we will show in Section 6.

6 Sample Complexity, Utility and Active Learning

Sample complexity is usually defined as the number of samples needed to learn a function to within some epsilon of accuracy.[9][10] Such a definition restricts current sample complexity approaches to a single end use and sample cost scenario. More flexible approaches are needed. We propose that sample complexity should be defined in terms of end use and sample cost.

Observation 6.1: *Sample complexity should be defined in terms of end use and sample cost.*

Discussion. Sample complexity is not just a product of the function to be learned but is a product of the use to which the function will be put. When end use differs from misclassification error rate, the sample complexity should change to match. For example, the sample complexity of learning in high assurance systems[11] should be higher than the complexity of learning the same function for less critical purposes. Furthermore, in practice, the number of samples needed is only an accurate measure when all samples have uniform cost. If some samples cost more than others, then analyzing the cost required to gather the right samples becomes important. Selecting the "best" samples is the job of active learning.

Active learning attempts to reduce sample complexity through selective sampling. We will evaluate the impact of end use and sample cost on active learning. There are several active learning scenarios, including: pedagogical, pool, generative, or stream; with discrete or continuous functions; with inductive or transductive evaluation; and with on-line vs. off-line sampling. Space prevents discussing all the above scenarios, however, a more detailed discussion can be found in [1]. Similar end use and sampling cost assumptions need to be made in all of the above situations, so without loss of generality we will focus on one of the more common active learning cases: the pool-based, discrete, transductive, off-line selective sampling scenario. The goal of pool based transductive active learning is to select examples from the pool set $\mathscr{D}_{pool}$ in order to generalize only over specific known feature examples in the test set $\mathscr{D}^X_{test}$.

Most so called "statistically optimal" selective sampling, is actually only greedy optimal [12, 13]. To date, the authors have never seen the full optimal active learning equations actually written down in their entirety. The fully optimal technique for active learning in the above scenario is to select the sample location that satisfies the following rather complex equations:

The value of sampling a set of data S is:

$$VOI(S) = \nu \sum_{x_k \in \mathscr{D}^X_{test}} p(x_k) \Bigg[\max_{d_j \in \mathbb{D}_D} \sum_{d_i \in \mathbb{D}_Y} p(y_i|\mathbf{x}_k, \mathscr{D}_{train} \cup S) \sum_{o_h \in \mathbb{D}_O} p(o_h|y_i, d_j) U(o_h) - \max_{d_j \in \mathbb{D}_D} \sum_{y_i \in \mathbb{D}_Y} p(y_i|\mathbf{x}_k, \mathscr{D}_{train}) \sum_{o_h \in \mathbb{D}_O} p(o_h|y_i, d_j) U(o_h) \Bigg], \quad (7)$$

where ν is the number of times that we expect to be able to exploit the information.

Let the cost of obtaining a set of data S is defined as $C(S)$. Normally this is just the sum of the cost of all the individual elements in S although more complex cases can be envisioned.

$$\begin{aligned} EVSI(\mathbf{x}_{t_1},\mu) = & \sum_{c_1\in\mathbb{D}_Y} p(y_{c_1}|\mathbf{x}_{t_1},\mathscr{D}_{train}) \max_{t_2\in\mathscr{D}^X_{pool}} \sum_{c_2\in\mathbb{D}_Y} p(y_{c_2}|\mathbf{x}_{t_2},\mathscr{D}_{train}\cup\{(\mathbf{x}_{t_1},y_{c_1})\}) \\ & ... \max_{t_\mu\in\mathscr{D}^X_{pool}} \sum_{c_\mu} p(y_{c_\mu}|\mathbf{x}_{t_\mu},\mathscr{D}_{train}\cup\{(\mathbf{x}_{t_1},y_{c_1})...(\mathbf{x}_{t_\mu},y_{c_\mu})\}) \\ & [VOI(\{(\mathbf{x}_{t_1},y_{c_1})...(\mathbf{x}_{t_\mu},y_{c_\mu})\}) - C(\{(\mathbf{x}_{t_1},y_{c_1})...(\mathbf{x}_{t_\mu},y_{c_\mu})\})], \end{aligned} \tag{8}$$

where x_t represents possible sample locations and μ is the number of samples that will be taken. The optimal first sample to get is:

$$\hat{\mathbf{x}} = \operatorname*{argmax}_{\mathbf{x}_{t_1}\in\mathscr{D}^X_{pool}} EVSI(\mathbf{x}_{t_1},\mu).$$

These equations define the active learning problem in our scenario, and very similar equations can be given for the other possible active learning scenarios [1]. For example, inductive rather than transductive learning simply replace the sums over $\mathbf{x}\in\mathscr{D}^X_{test}$ with weighted sums over the domain of X, $\mathbf{x}\in\mathbb{D}_X$ weighted by $p(x)$.

There are several pieces of information that are necessary in order to solve the above equations. These include: the number of times that the information will be exploited ν; the number of samples that will be taken μ; a distribution over $p(\mathbf{x}')$; a prior $p(f)$; the full posterior predictive $p(y|\mathbf{x},\mathscr{D}_{train})$; knowledge about end use, the domains of U and O, the distribution $p(o_h|y_i,d_j)$, and the utility function $U(o_h)$; and finally knowledge about the sample cost for gathering a set of data $C(S)$. This list indicates exactly what issues can theoretically affect active learning. We will now deal with each of these items in tern:

Observation 6.2: *The importance of ν and μ are often overlooked.*

Discussion. Doubling ν effectively doubles the value of information. Information that might have appeared too expensive with low values for ν can be worth sampling with higher values for ν.

The expected value of getting the first sample can change depending on the number of subsequent samples that will be taken, μ. A simple example can make the reasons for this more clear. If no single sample is sufficient to change the maximum expected utility decision, then it can be shown that $EVSI(\mathbf{x},\mu=1)=0-C(x),\forall x$. However, if two samples together have the potential to change the maximum expected utility decision, then it is possible to have a positive $EVSI(\mathbf{x},\mu=2)$. This is one reason that greedy active learning approaches can sometimes perform poorly.

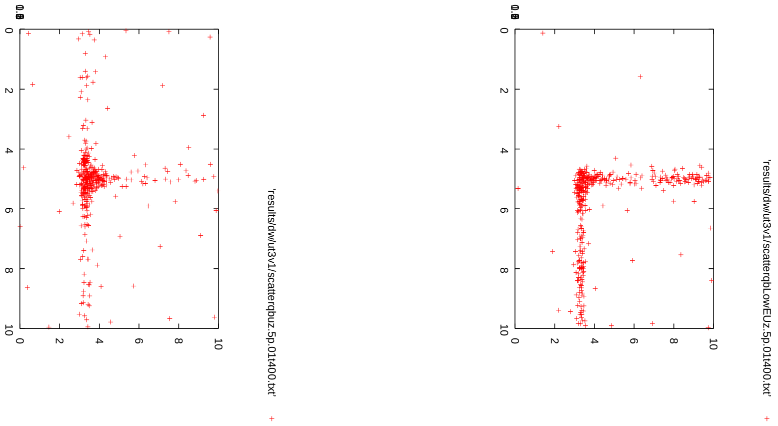
Observation 6.3: *Information about the distribution $p(\mathbf{x}')$ is important for active learning.*

Discussion. When performing simple supervised learning $\mathbf{X}$ and $\mathbf{X}'$ are usually observed and modeling these random variables is less important. For transductive active learning, the features in the test set are given, and these samples provide necessary information concerning $p(\mathbf{x}')$. For inductive active learning the sums over $\mathbf{x}'\in\mathscr{D}^X_{test}$ are replaced with a weighted sum over the domain of X', $\mathbf{x}'\in\mathbb{D}'_X$ weighted by the likelihood that $\mathbf{x}'$ is in the test set $p(\mathbf{x}')$, requiring an explicit model of $p(\mathbf{x}')$.

Theorem 6.4 (The Active Learning No-Free-Lunch Theorem)**:** *Given a uniform prior $p(f)$ over all possible discrete finite functions, a uniform sample cost function C, a test set that is off training set $\mathscr{D}_{train}\cap\mathscr{D}_{test}=\emptyset$, and a pool set that is off test set $\mathscr{D}_{pool}\cap\mathscr{D}_{test}=\emptyset$, then all active learning approaches perform the same.*

Discussion. This means that uniform priors can be more disruptive for active learning than they were for supervised learning. With uniform priors there is no reason to prefer one active learning approach over another, since there is no value of sample information in any off training set location.

Observation 6.5: *A technique for estimating full posterior predictives is necessary for active learning for arbitrary utility function. Classifiers that do not report full posterior predictives can be used, but will only be correct for certain end uses.*



(a) (b)

Figure 2: a) QBU scatter plot. b) QBLowEU scatter plot.

Discussion. VOI is computed using the difference between the maximum expected utility given the sample data minus the maximum expected utility without the sample data. Full posterior predictives (or sufficient statistics thereof) are necessary in order to compute the maximum expected utility for arbitrary utility functions (See Theorem 5.2). However, if the end use is known beforehand, it is sometimes possible to produce an insufficient statistic that will still allow the computation of the maximum expected utility, but only for that specific end use.

Example 6.1: Many approximate techniques exist for active learning, and some do not use classifiers that report full posterior predictives. Query-by-Committee is one such example. However, according to Observation 6.5, such algorithms must have some technique for approximating the uncertainty in the posterior predictive. In the case of Query-by-Committee, the committee can be seen as producing a monte carlo approximation to the uncertainty about the maximum decision for whatever utility assumptions are made by the classifiers that constitute the committee [14].

Observation 6.6: *Active learning depends intimately upon end use.*

Discussion. As in Observation 6.5, since the VOI computation requires a computation of the maximum expected utility, active learning can not be performed without some indication of end use.

Example 6.2: An example involves a simple two dimensional classification task with three different classes. If the utility function involves misclassification error rate, then all the decision boundaries are important, and an active learner should sample in locations surrounding all decision boundaries (See Figure 2(a)). On the other hand, if we change the utility function so that misclassifications between two of the classes are irrelevant, then it would be un-necessary for the active learner to sample along that decision boundary (See Figure 2(b)). Uncertainty Sampling, (sometimes called Query by Uncertainty, or QBU) and Query by Lowest Expected Utility (QBLowEU) perform differently because they each have different utility assumptions. QBU assumes misclassification error rate, and thus samples along the un-necessary decision boundary (Figure 2(a)) while QBLowEU performs better by ignoring this boundary (Figure 2(b)).

Observation 6.7: *Active learning depends upon the sample cost, and different sample costs should lead to different active learning solutions.*

Example 6.3: One active learning project involved annotating parts of speech. Active learning was applied to maximize the value of the expensive human annotation. However, it was unclear how the human annotators would be paid. Annotators could be paid either by the sentence, by the word, or by the hour, and the decision on which payment technique to use had not yet been made. Not only did the best active learning technique change with the way annotators were paid, but the technique that was best in one situation actually performed worse than random in the other situation. The complete results are not presented here for space considerations, but have been previously published elsewhere [15]. It was thus impossible to select the best active learning technique until the mechanism for paying annotators was determined. Despite the importance of these issues, several published studies on active learning for part of speech tagging, have all ignored how the annotators were paid.

7 Conclusions

Utility impacts supervised learning in at least two ways, through end use and sample costs. Much of Machine Learning literature has attempted to analyze the Supervised Learning problem independent of end use and sample costs.

The traditional No-Free-Lunch Theorems depended on a specific end use utility assumptions. We have recast the traditional No-Free-Lunch Theorem in Bayesian decision theoretical terms, and have proposed a new Bayesian No-Free-Lunch Theorem that is more fundamental than the traditional No-Free-Lunch Theorem, because it is independent of end use utility assumptions. This new Theorem indicates that uniform priors lead to uniform posteriors. Furthermore, it can be shown that *a priori* distinctions between learning algorithms exist for some end use utility assumptions.

We separate the concepts of classification (producing a probability distribution over the class output), and decision making (determining the best decision given the probabilistic classification). We show that the classification task itself can be performed independently of any end use assumptions. However, classification algorithms that return a point estimate or other insufficient statistic can not be used to make optimal decisions for arbitrary end use situations. Thus classifiers that return insufficient statistics (such as the most likely class) are all making some implicit end use assumptions that may or may not be met. When all end use utility functions are equally likely all decisions will have the same expected utility and all classification techniques perform equally well.

Utility from end use and sample costs both impact sample complexity and active learning. There are several elements that all impact the active learning problem: the number of times that the information will be exploited ν; the number of samples that will be taken μ; a distribution over $p(\mathbf{x})$; a prior $p(f)$; the full posterior predictive; end use; and the sample cost for gathering a set of data $C(S)$.

References

[1] James L. Carroll. *A Bayesian Decision Theoretical Approach to Supervised Learning, Selective Sampling, and Empirical Function Optimization*. PhD thesis, Brigham Young University, March 2010.

[2] Robbie Haertel, Eric Ringger, Kevin Seppi, James Carroll, and Peter McClanahan. Assessing the costs of sampling methods in active learning for annotation. In *Proceedings of the Conference of the Association of Computational Linguistics (ACL-NAACL: HLT 2008)*, 2008.

[3] Robbie A. Haertel, Kevin D. Seppi, Eric K. Ringger, and James L. Carroll. Return on investment for active learning. In *NIPS Workshop on Cost-Sensitive Machine Learning*, Whistler, British Columbia, Canada, 2008.

[4] Eric Ringger, Marc Carmen, Robbie Haertel, Kevin Seppi, Deryle Lonsdale, Peter McClanahan, James Carroll, and Noel Ellison. Assessing the costs of machine-assisted corpus annotation through a user study. In *The Proceedings of the Language Resources and Evaluation Conference (LREC)*, Morocco, 2008.

[5] James L. Carroll and Kevin D. Seppi. No-free-lunch and bayesian optimality. *Meta-Learning IJCNN Workshop*, 2007.

[6] David H. Wolpert. The supervised learning no-free-lunch theorems. Technical Report 269-1, NASA Ames Research Center, 2001.

[7] Tom M. Mitchell. The need for biases in learning generalizations. Technical Report CBM-TR-117, Rutgers Computer Science, New Brunswick, New Jersey, 1980.

[8] James O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag New York, Inc., New York and Berlin and Heidelberg, 1980.

[9] L. G. Valiant. A theory of the learnable. *ACM*, 27(11):1134–1142, 1984.

[10] A Blumer, A Ehrenfeucht, D Haussler, and MK Warmuth. Learnability and the vapnik-chervonenkis dimension. *Journal of the Association for Computing Machinery*, 1989:929–965, 1989.

[11] James L. Carroll, Christopher K. Monson, and Kevin D. Seppi. A bayesian CMAC for high assurance learning. *Applications of Neural Networks in High-Assurance Systems, NASA-IJCNN Workshop*, 2007.

[12] Nicholas Roy and Andrew Mccallum. Toward optimal active learning through sampling estimation of error reduction. In *Procedings of the 18th International Conference on Machine Learning*, pages 441–448. Morgan Kaufmann, 2001.

[13] David A. Cohn, Zoubin Ghahramani, and Michael I. Jordan. Active learning with statistical models. In *Journal of Artificial Intelligence Research*, volume 4, pages 129–145. AI Access Foundation and Morgan Kaufmann Publishers, 1996.

[14] I. Dagan and S. Argamon-Engelson. Committee-based sample selection for probabilistic classifiers. *Journal of Artificial Intelligence Research*, 11:335–360, 1999.

[15] James L. Carroll, Robbie Haertel, Peter McClanahan, Eric Ringger, and Kevin Seppi. Modeling the annotation process for ancient corpus creation. *Proceedings of the International Conference of Electronic Corpora of Ancient Languages (ECAL), or Chatressar*, November 2007.