Agnostic Bayesian Learning of Ensembles

Alexandre Lacoste[*] ALEXANDRE.LACOSTE.1@ULAVAL.CA
Département d'informatique et de génie logiciel, Université Laval, Québec, Canada, G1K-7P4

Hugo Larochelle HUGO.LAROCHELLE@USHERBROOKE.CA
Département d'informatique, Université de Sherbrooke, Québec, Canada, J1K-2R1

Mario Marchand MARIO.MARCHAND@IFT.ULAVAL.CA
Département d'informatique et de génie logiciel, Université Laval, Québec, Canada, G1K-7P4

François Laviolette FRANCOIS.LAVIOLETTE@IFT.ULAVAL.CA
Département d'informatique et de génie logiciel, Université Laval, Québec, Canada, G1K-7P4

Abstract

We propose a method for producing ensembles of predictors based on holdout estimations of their generalization performances. This approach uses a prior directly on the performance of predictors taken from a finite set of candidates and attempts to infer which one is best. Using Bayesian inference, we can thus obtain a posterior that represents our uncertainty about that choice and construct a weighted ensemble of predictors accordingly. This approach has the advantage of not requiring that the predictors be probabilistic themselves, can deal with arbitrary measures of performance and does not assume that the data was actually generated from any of the predictors in the ensemble. Since the problem of finding the *best* (as opposed to the true) predictor among a class is known as agnostic PAC-learning, we refer to our method as *agnostic Bayesian learning*. We also propose a method to address the case where the performance estimate is obtained from k-fold cross validation. While being efficient and easily adjustable to any loss function, our experiments confirm that the agnostic Bayes approach is state of the art compared to common baselines such as model selection based on k-fold cross-validation or a learned linear combination of predictor outputs.

1. Introduction

When designing a machine learning system that relies on a trained predictor, one is usually faced with the problem of choosing this predictor from a finite class of models. In practice, the class of models might correspond to different learning algorithms or to different choices of hyperparameters for a specific learning algorithm. A common approach to this problem is to estimate the generalization performance of each predictor on a holdout dataset (through a training/validation set split or using k-fold cross-validation) and use the predictor with the best performance. However, this approach is invariably noisy and overfitting can become a problem. A more successful procedure is to construct an ensemble of many different learned predictors. Many machine learning contests are won this way (Guyon et al., 2010). For instance, the winning team of the Netflix's contest relied on a final predictor trained on the output of the learned models (Bell et al., 2007). Great care must be taken however to avoid overfitting, e.g. by carefully tuning the predictor's own regularization hyperparameters. The choice of the final predictor is likely to influence the end result as well.

At the heart of this selection problem is our inability to know for sure which predictor is the best among our model class. One natural way to reason about such uncertainty would be to formulate it in probabilistic terms. In this paper, we propose to follow this paradigm by formulating priors about the expected performance of each predictor in our chosen class of models. We then use the observed loss measurements on each held-out example as evidence for updating our posterior over the identity of the best predictor in the model class. At test time, we can use this posterior to weight the contribution of each predictor in the ensemble that performs the final prediction.

Proceedings of the 31st *International Conference on Machine Learning*, Beijing, China, 2014. JMLR: W&CP volume 32. Copyright 2014 by the author(s).

We explore different ways of expressing priors over predictor performances and discuss how to perform Bayesian inference. As we will see, this simple paradigm naturally takes into account the correlation between the predictor's output so as to leverage diversity among the ensemble, which is another desiderata for ensemble learning and model averaging methods.

Unlike Bayesian model averaging (Hoeting et al., 1999), our approach does not require that the predictors be themselves probabilistic. It can also deal with arbitrary performance measures. More crucially, this approach does not assume that the observed data has been generated by a predictor from the model class. In other words, we are not looking for the predictor that best explains the observed data, assuming it was generated by a predictor coming from our model class. Instead, at the centre of our approach, we want to find the *best* predictor in terms of a task's performance measure and among all available predictors, while reasoning about our uncertainty around this problem in a Bayesian way.

The non-reliance on the assumption that the true underlying data generating function belongs to our model class is also at the center of agnostic PAC-learning. For this reason, we refer to the proposed framework as *agnostic Bayesian learning*.

Section 2 formally describes the agnostic Bayes approach. We then propose a few methods for obtaining a posterior distribution over a set of predictors. Section 4 presents an adaptation to k-fold cross-validation estimation of the losses. Finally, several experimental results are presented in Section 6.

2. Theoretical Setup

Throughout this paper, we use the inductive learning paradigm and make the usual assumptions of PAC learning theory (Kearns et al., 1994; Valiant, 1984). Thus, a task D corresponds to a probability distribution over the input-output space $\mathcal{X} \times \mathcal{Y}$. Given a training set $S \sim D^m$, the objective is to find, among a set $\mathcal{H}$, the *best* function $h^\star : \mathcal{X} \to \mathcal{Y}$. In general, $\mathcal{H}$ could be any set. However, this work will focus on the case where $\mathcal{H}$ is a finite set of predictors obtained from one or many learning algorithms, with various hyperparameters. We will refer to a member of $\mathcal{H}$ as an hypothesis.

To assess the quality of an hypothesis, we use a loss function $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ that quantifies the penalty incurred when h predicts $h(x)$ while the true answer is y. Then, we can define the risk $R_D(h)$ as being the expected loss of h on task D, i.e. $R_D(h) \overset{\text{def}}{=} \underset{x,y\sim D}{\mathbf{E}} \mathcal{L}(h(x), y)$. Finally, the *best*[1] function is simply the one minimizing the risk, i.e. $h^\star \overset{\text{def}}{=} \underset{h\in\mathcal{H}}{\operatorname{argmin}} R_D(h)$.

Since we do not observe D, it is not generally possible to find $h^\star$ with certainty. For this reason, we are interested in inferring $h^\star$ while modeling our uncertainty about it, using a posterior probability distribution $p(h^\star{=}h|S)$. Then, after marginalizing $h^\star$, we obtain a probabilistic prediction

$$p(y^\star{=}y|x, S) = \sum_{h\in\mathcal{H}} p(h^\star{=}h|S)p(y^\star{=}y|x, h),$$

where $y^\star$ stands for the prediction made by $h^\star$ for a given x. We note that the uncertainty in this prediction solely comes from our lack of knowledge about $h^\star$.

In order to perform a final prediction $\hat{y}$ for a given x it is tempting to use the optimal Bayes decision theory

$$\hat{y} = \underset{y'\in\mathcal{Y}}{\operatorname{argmin}} \sum_{y\in\mathcal{Y}} p(y^o{=}y|x, S)\mathcal{L}(y', y),$$

where y^o is the random variable corresponding to the observed values of y. However, the contrast between $p(y^o{=}y|x, S)$ and $p(y^\star{=}y|x, S)$ prevents us from using this approach. To this end, we use:

$$\hat{y} = \underset{y\in\mathcal{Y}}{\operatorname{argmax}}\, p(y^\star{=}y|x, S),$$

the most probable answer. This yields the following ensemble method:

$$E^\star(x) \overset{\text{def}}{=} \underset{y\in\mathcal{Y}}{\operatorname{argmax}} \sum_{h\in\mathcal{H}} p(h^\star = h|S)I[h(x) = y] \quad (1)$$

Before going further, we first review the usual Bayesian model averaging approach to highlight the fact that it does not exactly use $p(h^\star{=}h|S)$.

2.1. Standard Bayesian Model Averaging

To address the inductive learning paradigm, a variant of Bayesian model averaging can be used, where we suppose that a deterministic function $h^\rightarrow$, belonging to $\mathcal{H}$, is at the origin of the observed relationship between x and y. To perform inference on $h^\rightarrow$, we treat it as a random variable and assume that the observations in S have been altered by a noise model[2] $p(y^o = y|x, h)$. Using the i.i.d. assumption, $p(S|h) = \prod_{i=1}^m p(y_i|x_i, h)p(x_i)$. Next, by defining a prior distribution over $\mathcal{H}$, we can perform Bayesian inference to compute $p(h^\rightarrow{=}h|S) \propto p(S|h)p(h)$. Finally, after marginalization of h, we obtain

$$p(y^o{=}y|x, S) = \sum_{h\in\mathcal{H}} p(h^\rightarrow{=}h|S)p(y^o{=}y|x, h),$$

[1] The best solution may not be unique.

[2] The noise model could also be inferred. In this work, we use a fixed noise model.

which can be used with the optimal Bayes decision theory, to give the following ensemble decision rule

$$E^{\rightarrow}(x) \stackrel{\text{def}}{=} \underset{y' \in \mathcal{Y}}{\operatorname{argmin}} \sum_{y \in \mathcal{Y}} p(y^o{=}y|x, S)\mathcal{L}\left(y', y\right). \quad (2)$$

This formulation has proven to be very useful. However, if the *true* data-generating hypothesis does not belong to $\mathcal{H}$, the posterior $p(h^{\rightarrow}{=}\,h|S)$ may not converge to a posterior peaked at the best hypothesis $h^{\star}$, as $m \rightarrow \infty$. This misbehavior has been studied by Grünwald and Langford (2007) for the zero-one loss scenario. It was shown that under some reasonable restrictions on the prior, there exists a distribution D where the risk of the Bayes predictor is significantly higher than $R_D(h^{\star})$.

One way to overcome this inconsistency is to commit to a noise model that leverages the loss function, such as $p(y^o = y|h, x) \propto e^{-\beta\mathcal{L}(h(x), y)}$ for some fixed $\beta > 0$. Then, we have that $p(h^{\rightarrow}{=}\,h|S) \propto p(h)e^{-m\beta R_S(h)}$, where $R_S(h)$ is the empirical risk measured on S. As $m \rightarrow \infty$, the exponential part of the posterior ensures that any hypothesis not having a risk as low as $R_D(h^{\star})$ will have a negligible weight. We will examine this ensemble method to show that it is outperformed by the methods we propose in this paper.

2.2. Agnostic Bayes

Our main contribution is to propose a method for obtaining $p(h^{\star} = h|S)$, to be used in our ensemble decision $E^{\star}(x)$. The core idea of our approach is to directly reason about $h^{\star}$ instead of assuming the existence in $\mathcal{H}$ of a data generating $h^{\rightarrow}$ and trying to infer it. Since the observed losses in S suffice to distinguish $h^{\star}$ from other hypotheses in $\mathcal{H}$, we do not have to commit to a particular model for the relationship between x and y, and can limit ourselves to modeling the losses under each hypothesis.

Specifically, we propose to treat the risk $r_h \stackrel{\text{def}}{=} R_D(h)$ of each hypothesis h as a random variable, over which we will be defining a prior distribution. Let $l_{h,i} \stackrel{\text{def}}{=} \mathcal{L}\left(h\left(x_i\right), y_i\right)$ be the observed loss of hypothesis h for a sample $(x_i, y_i) \in S$. We also treat $l_{h,i}$ as random variables, governed by a conditional distribution $p(l_{h,i}|r_h)$. For example, in the zero-one loss $\mathcal{L}(y, y') = I[y \neq y']$ case, a natural choice would be to treat the observed losses $l_{h,i}$ as Bernoulli trials of parameter r_h. Assuming a beta prior over r_h, we could then perform Bayesian inference in order to reason about the uncertainty over r_h given the losses observed from S.

In the case of ensemble learning where we have multiple competing hypotheses, the losses $l_{h,i}$ are dependent across the different hypotheses h for the same example (x_i, y_i). Hence, we need to model the losses $\boldsymbol{l}_i \stackrel{\text{def}}{=} \left(l_{1,i}, l_{2,i}, \ldots, l_{|\mathcal{H}|,i}\right)$ for a given example jointly, given the joint risk for all hypotheses $\mathbf{r} \stackrel{\text{def}}{=} \left(r_1, r_2, \ldots, r_{|\mathcal{H}|}\right)$. Section 3 will discuss different joint priors $p(\mathbf{r})$ and observation models $p(\boldsymbol{l}_i|\mathbf{r})$. For now, we just note that from $p(\boldsymbol{l}_i|\mathbf{r})$, we can derive the likelihood of the set of losses $L \stackrel{\text{def}}{=} \{\boldsymbol{l}_i\}_{i=1}^m$ as $p(L|\mathbf{r}) = \prod_{i=1}^m p(\boldsymbol{l}_i|\mathbf{r})$ and, combined with our prior $p(\mathbf{r})$, perform Bayesian inference to obtain $p(\mathbf{r}|L) \propto p(L|\mathbf{r})p(\mathbf{r})$.

After obtaining $p(\mathbf{r}|L)$, we can now compute the posterior probability that a given hypothesis h is the best hypothesis $h^{\star}$ with the lowest risk among $\mathcal{H}$

$$\begin{aligned}
&Pr\left(\forall g \in \mathcal{H} : r_h \leq r_g \;|L\right) \\
&= \underset{\mathbf{r} \sim p(\cdot|L)}{\mathbf{E}} p\left(r_h \leq r_g,\ \forall g \neq h|\mathbf{r}\right) \\
&= \underset{\mathbf{r} \sim p(\cdot|L)}{\mathbf{E}} \mathrm{I}\left(r_h \leq r_g,\ \forall g \neq h\right).
\end{aligned}$$

We propose to use this posterior as our ensemble posterior in Equation (1). Under this model, L is a sufficient statistic for $\mathbf{r}$ and thus for h, i.e. $p\left(h|S\right) = p\left(h|L\right)$. Hence, to sample from $p(h|S)$, it suffices to sample a joint risk $\mathbf{r}$ from $p\left(\mathbf{r}|L\right)$ and to search for the hypothesis with the smallest risk. With repeated sampling, we can then approximately compute our ensemble decision rule. When $\mathcal{Y}$ is continuous, this approximation can affect $\underset{y \in \mathcal{Y}}{\operatorname{argmax}}\, p(y^{\star} = y|S, x)$. To address this issue, we consider a simple Gaussian model to smooth $p(y^{\star} = y|S, x)$. This yields a weighted average of the predictions: $E^{\star}(x) = \sum_{h \in \mathcal{H}} p(h^{\star} = h|S)h(x)$.

3. Priors Over the Joint Risk

In this section, we propose a few choices for the prior $p(\mathbf{r})$ and observation model $p(\boldsymbol{l}_i|\mathbf{r})$. We also discuss how to perform inference for $p(\mathbf{r}|L)$ under different assumptions of the loss function.

3.1. Dirichlet Distribution

We start with a proposal for the specific case of the zero-one loss. As described in Section 2, the observations $l_{h,i} \in \{0, 1\}$ are correlated and put together in a vector $\boldsymbol{l}_i \in \{0, 1\}^d$, where $d \stackrel{\text{def}}{=} |\mathcal{H}|$. We propose to consider the collection of observations $\{\boldsymbol{l}_i\}_{i=1}^m$ as coming from a categorical distribution of $N \stackrel{\text{def}}{=} 2^d$ possible states (i.e. outcomes). Therefore, the counts of observations $\mathbf{k} \stackrel{\text{def}}{=} (k_1, k_2, \ldots, k_N) \in \mathbb{N}^N$ come from a multinomial distribution of parameters $\mathbf{q}$ and m, where $\mathbf{q}$ is the probability of observing each event and sums to 1. With these assumptions, it is natural to use the Dirichlet distribution of parameter $\boldsymbol{\alpha}$ as the model for the prior over $\mathbf{q}$. The posterior distribution $p\left(\mathbf{q}|\mathbf{k}\right)$ is then a Dirichlet distribution of parameter $\boldsymbol{\alpha}{+}\mathbf{k}$. To convert the sample from $p(\mathbf{q}|L)$ to a sample from $p(\mathbf{r}|L)$, we define the state matrix $G \in \{0, 1\}^{d \times N}$ where the j^{th} column corresponds to the binary representation of j. Then, to obtain a sample from $p(\mathbf{r}|L)$, we sample

$\mathbf{q}$ from $\text{Dir}\left(\boldsymbol{\alpha}+\mathbf{k}\right)$ and use $\mathbf{r} = G\mathbf{q}$. Equivalently, we have $p\left(\mathbf{r}|L\right) = \mathbf{E}_{\,\mathbf{q}\sim\text{Dir}(\boldsymbol{\alpha}+\mathbf{k})}\,\mathrm{I}\left(G\mathbf{q}=\mathbf{r}\right)$.

Naively sampling from this posterior yields an algorithm with computational complexity of $O(d2^d)$. However, using a neutral prior of the form $\boldsymbol{\alpha} = \widetilde{\alpha}\mathbf{1}_N$ and the stick breaking representation of the Dirichlet (see Lemma 3.1 of Sethuraman (1991)), we have the following identity

$$\theta X_{\boldsymbol{\alpha}} + (1-\theta)X_{\mathbf{k}} = X_{\boldsymbol{\alpha}+\mathbf{k}},$$

where $\theta \sim \text{Beta}(\widetilde{\alpha}N, m)$, $X_{\boldsymbol{\alpha}} \sim \text{Dir}(\boldsymbol{\alpha})$, $X_{\mathbf{k}} \sim \text{Dir}(\mathbf{k})$, $X_{\boldsymbol{\alpha}+\mathbf{k}} \sim \text{Dir}(\boldsymbol{\alpha}+\mathbf{k})$. Since most values in $\mathbf{k}$ are zeros, samples from $\text{Dir}(\mathbf{k})$ can be obtained in $O(m)$. Thus, we are left with the task of sampling from $\text{Dir}(\boldsymbol{\alpha})$, which can be approximated efficiently using $10\cdot\widetilde{\alpha}N$ samples from the stick breaking process (Sethuraman, 1991). Since $\widetilde{\alpha}N > m$ yields too much importance to the prior, one can safely assume that $\widetilde{\alpha}N \leq m$ and obtain a sample from the prior with computational complexity O(m) .

3.2. Bootstrap Inference

We point out that the Dirichlet posterior presented in Section 3.1 is a generalization of Rubin's Bayesian bootstrap (Rubin, 1981) and is equivalent in the limit $\widetilde{\alpha} \rightarrow 0$. Also, Rubin showed that the Bayesian bootstrap is statistically tightly related to Efron's bootstrap (Efron, 1979). For these reasons, we also consider the bootstrap as a candidate for a simple and generic method to sample from $p(\mathbf{r}|L)$. This is done by sampling with replacement a set $\{\boldsymbol{l}'_i\}_{i=1}^m$ from $\{\boldsymbol{l}_i\}_{i=1}^m$. To obtain $\mathbf{r}$, we use $r_h \leftarrow \sum_{i=1}^m \frac{1}{m} l'_{h,i}; \forall h \in \mathcal{H}$.

3.3. t Distribution

In this section, we make the assumption that the variables $\boldsymbol{l}_i$ are observations coming from a multivariate normal distribution of dimensionality $|\mathcal{H}| \overset{\text{def}}{=} d$, whose mean parameter corresponds to the true risk $\mathbf{r}$. While the normal assumption is generally not true, it can be justified from the central limit theorem. As we will see, experiments in Section 6 show that this assumption works well in practice even with the zero-one loss function, which is one of the most extreme cases of non Gaussian samples.

Specifically, assuming that $p(\boldsymbol{l}_i|\mathbf{r},\Lambda)$ is normal, the likelihood of $L \overset{\text{def}}{=} \{\boldsymbol{l}\}_{i=1}^m$ is

$$p\left(L|\mathbf{r},\Lambda\right) \;\propto\; |\Lambda|^{\frac{m}{2}} e^{\left(-\frac{1}{2}\sum_{j=1}^m(\boldsymbol{l}_j-\mathbf{r})^T\Lambda(\boldsymbol{l}_j-\mathbf{r})\right)}. \tag{3}$$

We want to favor the use of priors over $\mathbf{r}$ and covariance matrix Λ^{-1} such that the posterior $p(\mathbf{r},\Lambda|L)$ is tractable. This can be achieved using the normal-Wishart distribution (DeGroot, 2005, p. 178)

$$p\left(\mathbf{r},\Lambda\right) \;=\; \mathcal{N}\left(\mathbf{r}\middle|\mathbf{r}_0,\left(\kappa_0\Lambda\right)^{-1}\right)\mathcal{W}\left(\Lambda|T_0,\nu_0\right),$$

where $\mathcal{N}$ and $\mathcal{W}$ are the normal and Wishart distributions respectively, $\mathbf{r}_0$ and T_0 are the mean and covariance prior, while κ_0 and ν_0 are parameters related to the confidence we have in $\mathbf{r}_0$ and T_0 respectively (with restrictions $\kappa_0 > 0$ and $\nu_0 > d-1$). Thanks to conjugacy, after observing L, we have that the posterior $p\left(\mathbf{r},\Lambda|L\right)$ is also a normal-Wishart distribution of parameters κ_m, ν_m, $\mathbf{r}_m$ and T_m as follows:

$$\begin{aligned}
\kappa_m &= \kappa_0 + m \\
\nu_m &= \nu_0 + m \\
\mathbf{r}_m &= \frac{\kappa_0\mathbf{r}_0 + m\bar{\boldsymbol{l}}}{\kappa_m} \\
T_m &= T_0 + mS + m\frac{\kappa_0}{\kappa_m}\left(\mathbf{r}_0-\bar{\boldsymbol{l}}\right)\left(\mathbf{r}_0-\bar{\boldsymbol{l}}\right)^T
\end{aligned} \tag{4}$$

where $\bar{\boldsymbol{l}} \overset{\text{def}}{=} \frac{1}{m}\sum_{i=1}^m \boldsymbol{l}_i$ and $S \overset{\text{def}}{=} \frac{1}{m}\sum_{i=1}^m(\boldsymbol{l}_i-\bar{\boldsymbol{l}})(\boldsymbol{l}_i-\bar{\boldsymbol{l}})^T$. Since our goal is to obtain a posterior distribution over $\mathbf{r}$ only, we have to marginalize out Λ from $p\left(\mathbf{r},\Lambda|L\right)$. By doing so, we obtain the multivariate Student's t distribution with $\widetilde{\nu} \overset{\text{def}}{=} \nu_m - d + 1$ degrees of freedom (DeGroot, 2005, p. 179)

$$p\left(\mathbf{r}|L\right) \;=\; t\left(\mathbf{r}\middle|\widetilde{\nu},\mathbf{r}_m,\tfrac{T_m}{\kappa_m\widetilde{\nu}}\right). \tag{5}$$

Samples from this multivariate t-distribution are done by sampling from the normal distribution $\mathbf{z} \sim \mathcal{N}\left(0,\frac{T_m}{\kappa_m\widetilde{\nu}}\right)$, sampling from the chi-squared distribution $\xi \sim \chi^2(\widetilde{\nu})$ and computing $\mathbf{r}_m + \mathbf{z}\sqrt{\frac{\widetilde{\nu}}{\xi}}$. This gives an overall computational complexity of $O\left(d^2(m+k+d)\right)$ to obtain k samples.

For setting the parameters $\mathbf{r}_0$, T_0, κ_0 and ν_0 of the prior, we chose values that were as neutral as possible and numerically stable: $\mathbf{r}_0 = 0.5 \times \mathbf{1}_d$, $T_0 = 0.25 \times I$, $\kappa_0 = 1$ and $\nu_0 = d$.

3.4. Posterior Behavior with Correlated Hypotheses

One advantage of the agnostic Bayes posterior for constructing an ensemble is that it naturally encourages diversity among the predictors, even in the presence of correlation between the predictors in $\mathcal{H}$. We illustrate this with a simple example, shown in Table 1, comparing an agnostic Bayes ensemble with bootstrap inference ($E_b^\star$) and a Bayesian model averaging ensemble with a loss-based noise model and flat prior over the hypotheses ($E^\rightarrow$). Table 1(top) illustrates the case of three equally good but different hypotheses, based on three observed losses for each predictor. We see that both $E_b^\star$ and $E^\rightarrow$ equally weight the three hypotheses, as expected.

Now, in Table 1(bottom), we include into $\mathcal{H}$ an additional hypothesis h_4, which is identical to h_3. We then observe that $E_b^\star$ naturally maintains diversity within the ensemble, by reducing the mass of the identical hypotheses h_3 and

Table 1. Illustration of the posteriors in an agnostic Bayes ensemble ($E_b^\star$) and in Bayesian model averaging ($E^\rightarrow$). **top:** Uncorrelated predictors. **bottom:** Addition of a correlated predictor.

	$\boldsymbol{l}_1$	$\boldsymbol{l}_2$	$\boldsymbol{l}_3$	$p(h^\star \mid S)$	$p(h^\rightarrow \mid S)$
h_1	1	0	0	0.33	0.33
h_2	0	1	0	0.33	0.33
h_3	0	0	1	0.33	0.33

↓

	$\boldsymbol{l}_1$	$\boldsymbol{l}_2$	$\boldsymbol{l}_3$	$p(h^\star \mid S)$	$p(h^\rightarrow \mid S)$
h_1	1	0	0	0.31	0.25
h_2	0	1	0	0.31	0.25
h_3	0	0	1	0.19	0.25
h_4	0	0	1	0.19	0.25

h_4, compared to $E^\rightarrow$ which still weights all hypotheses equally. Diversity is usually considered to be beneficial when constructing an ensemble of predictors (Roy et al., 2011), motivating the use of agnostic Bayes for this task.

4. Model Averaging for Trained Predictors

As mentioned in Section 2, one natural application for the inference of the best hypothesis is model averaging of trained predictors. Namely, let $\mathcal{A}_\gamma$ be a learning algorithm with a hyperparameter configuration $\gamma \in \Gamma$ and let $h_\gamma = \mathcal{A}_\gamma(T)$ represent the classifier obtained using a training set $T \sim D^n$, disjoint from S. The set $\mathcal{H}$ contains all classifiers obtained from each $\gamma \in \Gamma$, when $\mathcal{A}_\gamma$ is trained on T, i.e. $\mathcal{H} \overset{\text{def}}{=} \{h_\gamma | \gamma \in \Gamma\}$. Finally, to obtain the posterior $p(h_\gamma^\star = h_\gamma | S)$, we rely on the set S. Experiments in Section 6 will show that this approach significantly outperforms the usual method of selecting the hypothesis minimizing $R_S(h_\gamma)$.

Unfortunately, this scenario requires that the hypotheses h_γ be trained on a set of data T separate from S, in a training/validation split fashion, wasting an opportunity to measure the hypotheses performance on T as well. Our next step is thus to adapt our agnostic Bayes approach to the k-fold cross-validation scenario, which more fully uses the available data.

4.1. Adapting to k-fold Cross-Validation

Let $\{V_1, V_2, \ldots, V_k\}$ be a partition of S, and let $h_{\gamma,j} \overset{\text{def}}{=} \mathcal{A}_\gamma\left(S \setminus V_j\right)$. Now, denote the loss of model γ on the example (x_i, y_i) as $\widetilde{l}_{\gamma,i} \overset{\text{def}}{=} \mathcal{L}\left(h_{\gamma,j_i}(x_i), y_i\right)$, where j_i is the unique index j such that $(x_i, y_i) \in V_j$. Finally, let $\widetilde{\boldsymbol{l}}_i \overset{\text{def}}{=} \left(\widetilde{l}_{1,i}, \widetilde{l}_{2,i}, \ldots, \widetilde{l}_{|\Gamma|,i}\right)$. Unlike $\{\boldsymbol{l}\}_{i=1}^m$, it is well known that the set of k-fold generated losses $\{\widetilde{\boldsymbol{l}}\}_{i=1}^m$ contains dependencies across the different examples that are induced by the k-fold procedure (Bengio and Grandvalet, 2004). Since the posteriors described in Section 3 relied on independence across examples, we cannot simply ignore the dependencies induced within this process and must adapt our approach.

Specifically, we make the simplifying assumption that these dependencies only affect the effective number of samples. Intuitively, since samples are correlated, there may not be as many as it seems and the estimation of $p(\mathbf{r}|L)$ may be overly confident. We thus propose to add an extra parameter ρ, *the effective sample size ratio*, to compensate for these dependencies. While this parameter requires calibration, we describe in Section 4.2 an efficient method for automatically adjusting its value.

To include the effective sample size ratio in the methods described in Section 3, we will effectively act as if the collection $\{\boldsymbol{l}\}_{i=1}^m$ had been generated by artificially replicating a set of m original samples b times each, to give a new set of $bm' \overset{\text{def}}{=} m$ samples. Thus, the effective number of samples would be $m' = m/b$. Now, supposing that we know $\rho = m'/m$, we want to adapt the posterior's parameters in such a way that the posterior's distribution remains the same, on average, as before the "corruption".

Bootstrap: This is probably the simplest method to adapt. Out of the m observed events, we sample with replacement m' events instead, where $m' = \lceil \rho m \rceil$.

Dirichlet: In this case, each observed event is made to count for ρ instead of 1. After observing m events, the vector of counts $\mathbf{k}' \overset{\text{def}}{=} (k_1', k_2', \ldots, k_N')$ will now sum to m' instead of m.

t-Distribution: In this case, we adapt the quantities described in Equation (4) as follows: $\nu_{m'} = \nu_0 + m'$, $\nu_{m'} = \nu_0 + m'$, $\mathbf{r}_{m'} = \frac{\kappa_0 \mathbf{r}_0 + m' \bar{\boldsymbol{l}}}{\kappa_{m'}}$ and $T_{m'} = T_0 + m'S + m' \frac{\kappa_0}{\kappa_{m'}} \left(\mathbf{r}_0 - \bar{\boldsymbol{l}}\right) \left(\mathbf{r}_0 - \bar{\boldsymbol{l}}\right)^T$.

4.2. Tuning Parameters

To adjust ρ, we treat it as a parameter and fit it by optimizing the resulting ensemble's performance on S, thereby measuring how well the ensemble's weighting posterior can predict each label y_i in S from the hypotheses $(h_{1,j_i}(x_i), h_{2,j_i}(x_i), .., h_{|\Gamma|,j_i}(x_i))$. We've found this to work well in practice. This procedure is also akin to methods that learn a parameterized linear combination of predictors by training on generated examples $\tilde{S} \overset{\text{def}}{=} \left\{\left((h_{1,j_i}(x_i), h_{2,j_i}(x_i), .., h_{|\Gamma|,j_i}(x_i)), y_i\right)\right\}_{i=1}^m$. The best ρ from a set of 20 values equally spaced from 0.1 to 0.8 is used. We use a similar procedure to tune the prior parameter $\widetilde{\alpha}$ of the ensemble based on a Dirichlet prior.

5. Related Work

To overcome some mentioned weaknesses of Bayesian model averaging (such as the reliance on the existence of a single data-generating hypothesis belonging to $\mathcal{H}$), Kim and Ghahramani (2012) proposed an alternative method for Bayesian combination of classifiers. They suppose that, for a given x, the true label is at the origin of the behavior of each individual classifier. Therefore, by modeling the dependencies between each classifier on a validation set, they can perform inference of the original label. Unfortunately, it relies on a combination of MCMC and rejection sampling methods and the computational complexity of certain dependency models grows exponentially with $|\mathcal{H}|$. Thus, this approach is viable only for combining a small set of classifiers. It also only tackles classification tasks and doesn't take into account the loss related to the task at hand, as we do here.

Alternatively, ensemble pruning is an important approach to ensemble methods. Zhang et al. (2006) used semidefinite programming for solving a heuristic based on the covariance of the predictors. Interestingly, the core of their idea is highly related to the covariance matrix used in our t-distribution approach. Unfortunately, they can only address an approximation of their heuristic and it is limited to the zero-one loss.

6. Experiments

We performed experiments to assess the performance of the agnostic Bayes ensemble approach and compared with a few commonly used methods:

ArgMin (AMin): This method represents the common approach of selecting the model h_γ with the best estimated holdout risk $r_\gamma \overset{\text{def}}{=} \frac{1}{m}\sum_{i=1}^{m} l_{\gamma,i}$. When the minimum is not unique, we select one at random.

SoftMin (SMin): We use the Gibbs distribution with parameter β to produce a posterior distribution over the collection of h_γ from r_γ. *i.e.*, $p(h_\gamma|S) \propto e^{-\beta r_\gamma}$ and β is selected with the method described in Section 4.2. This represents the alternative Bayesian model averaging approach described in Section 2.1.

$E_b^\star, E_D^\star, E_B^\star, E_t^\star$: The different agnostic Bayes ensemble decision methods based on Equation (1) and using posterior inference based on the bootstrap, the Dirichlet distribution, the Bayesian bootstrap and the t-distribution respectively. Effective sample size ratio ρ and Dirichlet prior parameter $\widetilde{\alpha}$ are adjusted according to Section 4.2, while the t-distribution prior parameters are fixed to the values specified in Section 3.3. We use 1000 samples from $p(\mathbf{r}|L)$ to estimate $p(h|S)$.

MetaSVM (MSVM): We use MetaSVM to represent the state of the art approach *i.e.*, methods that learn a linear model over the set of models as a final predictor. This is done by using the collection $\tilde{S}$ described in Section 4.2 as a training set for the linear SVM. Traditional cross validation is used to select the best soft margin parameter over 20 candidates values ranging from 10^{-3} to 10^{0} on a logarithmic scale.

Meta Ridge Regression (MRR): When performing experiments on regression tasks, we use ridge regression as a substitution for MetaSVM. The regularization parameter is selected by the leave one out method over 30 candidates ranging from 10^{-4} to 10^{4} on a logarithmic scale.

6.1. Comparing Learning Algorithms On Multiple Datasets

The different model selection methods presented in the previous section are generic and are meant to work across different tasks. It is thus crucial that we test them on several datasets. For that, we have to rely on methods that do not assume commensurability across tasks, such as the sign test, the Wilcoxon signed rank test (WSR) (Demšar, 2006) and the Poisson binomial test (PB test) (Lacoste et al., 2012). The PB test is a Bayesian analogue of the sign test meant for comparing learning algorithms on a collection of tasks, called a context. More precisely, it provides a probabilistic answer to the question "*Does algorithm $\mathcal{A}$ have a higher probability of producing a better predictor than algorithm $\mathcal{B}$ in the given context?*", denoted by $p(\mathcal{A} \succ \mathcal{B}|\mathcal{W})$, where $\mathcal{W}$ represents the context.

To build a substantial collection of datasets, we used the AYSU collection (Ulaş et al., 2009) coming from the UCI and the Delve repositories and we added the MNIST dataset. We also converted the multiclass datasets to binary classification by either merging classes or selecting pairs of classes. The resulting context contains 38 datasets. We have also collected 22 regression datasets from the Louis Torgo collection.[3] to perform experiments using different loss functions.

The set Γ of models used in this experiment is a combination of SVMs, Artificial Neural Networks (ANN), random forests, extra randomized trees (Geurts et al., 2006) and gradient tree boosting (Friedman, 2001) with several variants of hyperparameters. Considering the algorithm name as a hyperparameter and a grid search for each algorithm, this yields a set of 692 hyperparameter configurations, all of which are evaluated using 10 folds cross validation. For the experiments on regression datasets, we used a combination of Kernel Ridge Regression (KRR), Support Vec-

[3] These datasets were obtained from the following source : http://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html

tor Regression (SVR), random forests, extra randomized trees and gradient boosted regression, yielding a total of 480 hyperparameter configurations. Except for a custom implementation of ANN and KRR, we used scikit-learn (Pedregosa et al., 2011) for all other implementations. For more details on the choice of hyperparameters, we refer the reader to the supplementary material.

6.2. Result Table Notation

Each conducted experiment compares the generalization performances of a set of M algorithms on a set of N datasets. In order to evaluate if the observed differences are statistically significant, we use the pairwise PB test where each cell of the table represents $p\,(\text{row} \succ \text{column})$. Since the table has a form of symmetry, we have grayed out redundant information and removed the first column. In addition, we also highlight in blue the results having p-values lower than 0.1 according to the one tail sign test. In general, we have observed a strong correlation between the p-values of the sign test and the probabilities obtained from the PB test. Note however that their values may differ and a highlighted cell does not imply a strong PB probability, nor the converse. Finally, we added a column to each table which reports the expected rank of each algorithm across the collection of datasets. The rank of predictor $h_i = \mathcal{A}_i(S_j)$ on test set T_j is defined as

$$\text{Rank}_{h_i,T_j} \stackrel{\text{def}}{=} \sum_{l=1}^{M} I\left[R_{T_j}(h_l) \leq R_{T_j}(h_i)\right].$$

Then, the expected rank is obtained from the empirical average $\mathbf{E}\left[\text{Rank}\right]_{h_i} \stackrel{\text{def}}{=} \frac{1}{N}\sum_{j=1}^{N} \text{Rank}_{h_i,T_j}$.

6.3. Comparison of Ensemble Decision Methods on Classification Tasks

Our first experiment compares the different methods and baselines in the setting where the hypotheses have been trained and validated on a single split of the dataset. In this scenario, the training data generates the set of hypotheses while the validation data provides observations for building an ensemble. Finally, a testing set is used to report the performances. The effective sample size ratio is fixed to 1 in this scenario.

From Table 2, there are no significant differences between our methods except for a slight reduction in generalization performances for $E_B^\star$, which corresponds to $E_D^\star$ with $\widetilde{\alpha}$ fixed to 0. In this experiment, the only adjusted parameter is $\widetilde{\alpha}$ in the method $E_D^\star$. This may explain why it is ranked first according to the expected rank metric. To simplify the result tables, further evaluations only includes $E_b^\star$ and $E_t^\star$.

Table 3 exhibits a clear conclusion : *The agnostic Bayes ensemble generalizes better than AMin.* Next, when comparing against MSVM and Softmin, while the results are note statistically significant, the expected rank is in favor of both agnostic Bayes ensembles. Also, we note that MSVM is not significantly better than AMin.

Table 2. Comparison of the four proposed agnostic model averaging methods, in the single training/validation split experiment (refer to Section 6.2 for notation).

	$E_D^\star$	$E_t^\star$	$E_b^\star$	$E_B^\star$	$\mathbf{E}$[rank]
$E_D^\star$	0.500	0.509	0.524	0.652	2.43 /4
$E_t^\star$	0.491	0.500	0.541	0.662	2.43 /4
$E_b^\star$	0.476	0.459	0.500	0.640	2.46 /4
$E_B^\star$	0.348	0.338	0.360	0.500	2.67 /4

Table 3. Comparison with the baseline models in the single training/validation split experiment (refer to Section 6.2 for notation).

	$E_b^\star$	**MSvm**	**SMin**	**AMin**	$\mathbf{E}$[rank]
$E_t^\star$	0.541	0.613	**0.787**	**0.911**	2.63 /5
$E_b^\star$	0.500	0.592	0.763	**0.905**	2.66 /5
MSvm	0.408	0.500	0.623	0.789	2.92 /5
SMin	0.237	0.377	0.500	0.759	3.19 /5
AMin	0.095	0.211	0.241	0.500	3.57 /5

It is well known that k-fold cross-validation provides a better estimate of the generalization performance of a learning algorithm than a single training/validation fold experiment. We thus performed another comparison for this setting. In this scenario, the agnostic Bayes method must now take into account the effective sample size ratio, as described in Section 4.1. Selected values ranges from 0.1 to 1 and were mainly concentrated between 0.3 and 0.6. The results are expressed in Table 4 and are similar to that of Table 3. Again, agnostic Bayes is significantly better than Argmin while MSVM is not.

Table 4. Comparison with the baseline models in the cross-validation experiment (refer to Section 6.2 for notation).

	$E_t^\star$	**MSvm**	**SMin**	**AMin**	$\mathbf{E}$[rank]
$E_b^\star$	0.507	0.575	0.707	**0.840**	2.70 /5
$E_t^\star$	0.500	0.578	0.720	**0.840**	2.75 /5
MSvm	0.422	0.500	0.577	0.725	2.95 /5
SMin	0.280	0.423	0.500	0.682	3.12 /5
AMin	0.160	0.275	0.318	0.500	3.46 /5

6.4. Changing the Loss Function

The results from the last section clearly demonstrate the advantage of mixing models over selecting a single one. While the agnostic Bayes methods outperform the baselines, we saw that simply using a linear learning algorithm also exhibits good performances. But what happens when the loss function changes? For example, we cannot use

MetaSVM for combining models on a regression task. We can adapt and use ridge regression but, since it minimizes the quadratic loss, it may not perform well if our task is to minimize the expected absolute difference loss (i.e., $\mathcal{L}(y, y') = |y - y'|$). In other words, to perform a linear combination of models, we have to redesign the learning algorithm for every loss functions. Moreover, some loss functions yield a non-convex optimization problem which requires some form of approximation, e.g., SVM uses the hinge loss in place of the zero-one loss. In contrast, the proposed agnostic Bayes approach is designed to work with any loss function.

Table 5. Comparison with the baseline models on regression tasks for the quadratic loss function (refer to Section 6.2 for notation).

	$E_t^\star$	**MRR**	**SMin**	**AMin**	**E**[rank]
$E_b^\star$	0.839	0.547	**0.929**	**0.992**	2.22 /5
$E_t^\star$	0.500	0.468	0.793	**0.986**	2.64 /5
MRR	0.532	0.500	0.554	0.809	2.88 /5
SMin	0.207	0.446	0.500	**0.992**	3.02 /5
AMin	0.014	0.191	0.008	0.500	4.23 /5

Table 6. Comparison with the baseline models on regression tasks for the absolute loss function (refer to Section 6.2 for notation).

	$E_t^\star$	**SMin**	**MRR**	**AMin**	**E**[rank]
$E_b^\star$	0.735	**0.953**	**0.859**	**0.995**	2.10 /5
$E_t^\star$	0.500	**0.932**	**0.821**	**0.995**	2.37 /5
SMin	0.068	0.500	0.769	**0.982**	3.06 /5
MRR	0.179	0.231	0.500	0.485	3.39 /5
AMin	0.005	0.018	0.515	0.500	4.08 /5

To outline the independence to the loss function of the agnostic Bayes methods, we performed experiments on regression tasks using both the quadratic loss and the absolute difference loss. We compared against the same baseline methods except for MetaSVM which was replaced by meta ridge regression (MRR) and its regularization parameter was selected by minimizing the appropriate loss function during cross validation. Table 5 presents the results obtained when using the quadratic loss function. While we worked with a totally different collection of datasets, the conclusions that follow from this experiment are surprisingly similar to the previous one. In this case, AMin is far down in ranking and the statistical significance of the observed differences are even stronger. Also, MRR is still performing relatively well.

Now, let us see what happens when we change the loss function to the absolute difference loss. Table 6 clearly shows an important degradation of MRR while the relative performances of the other methods are almost unchanged. In addition, the agnostic Bayes approach is now significantly better than the linear model. This clearly shows the importance of optimizing the appropriate loss function. Thus, justifying the usage of the agnostic Bayes ensemble.

7. Conclusion

We proposed the *agnostic Bayes* framework, which can be used to tackle the ubiquitous problem of model selection. This framework's central idea is to model the relationship between the hypotheses risks and observed empirical losses, without relying on assumptions about the true data-generating model. For one, this idea provides a new way of reasoning about machine learning problems. Also, the application to model selection has several desirable characteristics.

Generalization: The generalization performance of the agnostic Bayes ensemble is significantly better than just selecting the model minimizing the empirical expected loss. Also, our expected rank is systematically higher than any other evaluated methods on all experiments.

Flexibility: While most existing model selection algorithms is limited to a particular loss function, the agnostic Bayes ensemble can be used with any loss function. Also, our experiments showed how optimizing with the wrong loss function can be detrimental.

Speed: The bootstrap algorithm is simple to implement and has a linear computational complexity in the size of the dataset. When measuring the learning speed, we observed that the bootstrap algorithm can be several thousand times faster than MetaSVM.

Acknowledgement

Thanks to Calcul Québec for providing support and access to Colosse's high performance computer grid. This work was supported by NSERC Discovery Grants 122405 (M. M.) and 262067 (F. L.).

References

Isabelle Guyon, Amir Saffari, Gideon Dror, and Gavin Cawley. Model selection: Beyond the bayesian/frequentist divide. *The Journal of Machine Learning Research*, 11:61–87, 2010.

Robert M Bell, Yehuda Koren, and Chris Volinsky. The bellkor solution to the netflix prize. *KorBell Team's Report to Netflix*, 2007.

Jennifer A Hoeting, David Madigan, Adrian E Raftery, and Chris T Volinsky. Bayesian model averaging: a tutorial. *Statistical science*, pages 382–401, 1999.

M.J. Kearns, R.E. Schapire, and L.M. Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2):115–141, 1994.

Leslie G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

Peter Grünwald and John Langford. Suboptimal behavior of bayes and mdl in classification under misspecification. *Machine Learning*, 66(2-3):119–149, 2007.

Jayaram Sethuraman. A constructive definition of dirichlet priors. Technical report, DTIC Document, 1991.

D.B. Rubin. The bayesian bootstrap. *The annals of statistics*, 9(1):130–134, 1981.

B. Efron. Bootstrap methods: another look at the jackknife. *The annals of Statistics*, 7(1):1–26, 1979.

M.H. DeGroot. *Optimal statistical decisions*, volume 82. Wiley-interscience, 2005.

Jean-Francis Roy, François Laviolette, and Mario Marchand. From pac-bayes bounds to quadratic programs for majority votes. In *ICML*, pages 649–656, 2011.

Y. Bengio and Y. Grandvalet. No unbiased estimator of the variance of k-fold cross-validation. *The Journal of Machine Learning Research*, 5:1089–1105, 2004.

Hyun-Chul Kim and Zoubin Ghahramani. Bayesian classifier combination. *Journal of Machine Learning Research - Proceedings Track*, 22:619–627, 2012.

Yi Zhang, Samuel Burer, and W Nick Street. Ensemble pruning via semi-definite programming. *The Journal of Machine Learning Research*, 7:1315–1338, 2006.

Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

Alexandre Lacoste, François Laviolette, and Mario Marchand. Bayesian comparison of machine learning algorithms on single and multiple datasets. *Journal of Machine Learning Research - Proceedings Track*, 22:665–675, 2012.

Aydın Ulaş, Murat Semerci, Olcay Taner Yıldız, and Ethem Alpaydın. Incremental construction of classifier and discriminant ensembles. *Information Sciences*, 179(9): 1298–1318, April 2009.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, pages 1189–1232, 2001.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *The Journal of Machine Learning Research*, 12:2825–2830, 2011.