Model Selection with Unlabeled Data for Maximum Entropy Models

Xiaojin Zhu Rose Hoberman

December 16, 2000

Abstract

We tried to apply the Pythagorean property on a sequence of increasingly complex maximum entropy models to detect overfitting, and hence select the model with the appropriate level of complexity. It didn't work.

1 Introduction

Traditionally there are two broad categories of model selection methods: cross validation and complexity penalization. Cross validation needs reasonably large amount of labeled training data, while complexity penalization makes assumptions that may or may not be true in practice.

The newly emerged metric-based methods for model selection [1] form the third category. In these methods a *metric* $d(q_1, q_2)$ is defined between any two models q_1, q_2. For example,

$$d_{p(x)}(q_1, q_2) = E_{p(x)}(q_1(x) \neq q_2(x)) \tag{1}$$

can be proven to be a 'classification disagreement' metric on the marginal distribution $p(x)$. Now consider the true distribution p^* and two models q_1, q_2 we trained on labeled examples. Since a metric satisfies the triangle inequality, we have

$$d_{p(x)}(q_1, q_2) \leq d_{p(x)}(q_1, p^*) + d_{p(x)}(p^*, q_2)$$

If we approximate $d_{p(x)}(q_1, p^*)$ and $d_{p(x)}(p^*, q_2)$ with their training set error rates $\hat{d}(q_1, \tilde{p})$ and $\hat{d}(q_2, \tilde{p})$, and approximate $d_{p(x)}(q_1, q_2)$ with the marginal distribution $p_u(x)$ of a large amount of unlabeled data, we get

$$d_{p_u(x)}(q_1, q_2) \leq \hat{d}(q_1, \tilde{p}) + \hat{d}(q_2, \tilde{p})$$

which we can directly verify. If the inequality doesn't hold, at least one of the training errors doesn't reflect the true error. If both training errors are close to zero, we would know at least one model is overfitting. Then we would select the model with less complexity.

We want to apply similar reasoning to the special case where the trained models are maximum entropy models. Specifically, given some labeled training examples $L = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ generated by a true distribution $p*$, and a set of features $f_1, f_2, \ldots$, the *empirical constraints* are $E_{\tilde{p}}[f_i] = \sum_{(x,y) \in L} \tilde{p}(x, y) f_i(x, y)$, for $i = 1, 2, \ldots$. We can train a sequence of maximum entropy models $q_0, q_1, q_2, \ldots$ on L such that q_0 is the uniform model, q_1 contains feature f_1, q_2 contains features f_1, f_2 and so on, such that

$$E_{q_n}[f_i] = \sum_{x \in L} \tilde{p}(x) \sum_y q_n(y|x) f_i(x, y), \forall i \leq n$$

And we hope this increasingly complex sequence approaches the true distribution p^* [2]. However since the empirical constraints in general are not the true constraints, at some point the model q_n is going to overfit L and be farther away from p^*. We will detect overfitting with some unlabeled data $U = \{x_{m+1}, \ldots, x_M\}$ and the so called Pythagorean property for maximum entropy models.

Let $F = \{f_i : (X, Y) \rightarrow \mathcal{R} | i = 1, \ldots, n\}$ be a set of features. Let $c_1, \ldots, c_n$ be some arbitrary constraints (real numbers). Given some marginal distribution $p(x)$, the *feasible set* is defined as

$$\{p^f | \sum_x p(x) \sum_y p^f(y|x) f_i(x, y) = c_i, \forall i = 1, \ldots, n\}$$

i.e. all the conditional distributions satisfying the constraints under the marginal. Meanwhile, the *exponential family* of F consists of all conditional distributions of the form

$$q(y|x) = \frac{1}{Z(x)} exp(\sum_{i=1}^{n} \lambda_i f_i(x, y))$$

Let q^* be the maximum entropy distribution within the feasible set. It can be proven that q^* is also a member of the exponential family, and the following equation is true:

$$D(p^f || q) = D(p^f || q^*) + D(q^* || q) \tag{2}$$

where p^f is a member of the feasible set and q is a member of the exponential family. $D()$ is conditional Kullback-Leibler divergence on marginal $p(x)$:

$$D(p||q) = \sum_x p(x) \sum_y p(y|x) \log \frac{p(y|x)}{q(y|x)}$$

(2) is known as the Pythagorean property.

2 Method I

We train a sequence of maximum entropy models $q_0 \ldots q_n$ on a combination of labeled and unlabeled data:

$$Constraints : E_{\tilde{p}_L}[f_i]$$

$$Marginals : x \in U$$

We know

$$D_U(p^f||q_{n-1}) = D_U(p^f||q_n) + D_U(q_n||q_{n-1})$$

when p^f is any distribution satisfying the constraints.

If we assume the true distribution p^* of the data is in the feasible set (satisfies the constraints derived from the training data), then this equality will hold:

$$D_U(p^*||q_{n-1}) = D_U(p^*||q_n) + D_U(q_n||q_{n-1})$$

$D_U(q_n||q_{n-1})$ is calculated over the unlabeled data directly. We cannot calculate $D_U(p^*||q_{n-1})$ or $D_U(p^*||q_n)$, but we estimate them by $\hat{D}_L(\tilde{p}_L||q_{n-1})$ and $\hat{D}_L(\tilde{p}_L||q_n)$.

The hope is that if two successive models are both close to $\tilde{p_L}$, but far from each other, the equality will break. We know that only one of the two models could truly be close to the true distribution, so one of the estimates $\hat{D}(\tilde{p}_L||q_n)$ or $\hat{D}(\tilde{p}_L||q_{n-1})$ must be bad. We will discard the more complicated of the two models.

There are three problems with this method:

1. If p^* is not in the feasible set then the equality is not guaranteed to hold, and when it breaks we can say nothing about the accuracy of the estimates.

2. We obtain constraints $E_{\hat{p_L}}[f_i]$ from the labeled data. But we train the model on $x \in U$. This may result in inconsistent constraints. We might not be able to find a model.

3. The method is designed to detect overfitting. But by training with the marginals from the unlabeled data we prevent the models from fitting tightly to the labeled data (shown empirically by large $D(\tilde{p}_L||q_n)$). So there is no longer any overfitting to detect.

3 Method II

We train a sequence maximum entropy models $q_0 ... q_n$ on only the labeled data:

$$Constraints : E_{\tilde{p}_L}[f_i]$$

$$Marginals : x \in L$$

Assumption 1: q_n is the true maximum entropy model over the unlabeled data. Under this assumption, the following equality will hold:

$$D_U(p^*||q_{n-1}) = D_U(p^*||q_n) + D_U(q_n||q_{n-1})$$

Again, $D_U(q_n||q_{n-1})$ is calculated over the unlabeled data directly, but we cannot calculate $D_U(p^*||q_{n-1})$ or $D_U(p^*||q_n)$. But we know that

$$D_U(p^*||q) = -H(p^*) - \sum_i \lambda_i E_{p^*}[f_i] + \sum_x p^*(x) \log Z(x)$$

Given assumption 1 we can replace $E_{p^*}[f_i]$ with $E_{\tilde{p}_L}[f_i]$. We still don't know the true marginal, so we have to approximate it with the marginal on unlabeled data

$$p^*(x) \approx \tilde{p}_U(x)$$

We now have:

$$D_U(p^*||q_n) \;\; = -H(p^*) - \sum_i \lambda_i E_{\tilde{p}_L}[f_i] + \sum_x \tilde{p}_U(x) \log Z(x)$$

We still cannot compute $-H(p^*)$, but we don't need that to determine if the equality holds.

Unfortunately, the maximum entropy model trained on the labeled data is generally *not* the maximum entropy model on the unlabeled data. In our

x															y
1	1	1	1	1	1	0	0	0	0	1	0	0	1	1	1
0	1	0	0	0	0	0	1	1	0	1	1	1	1	0	0
0	0	0	0	1	0	1	1	0	1	0	0	0	1	0	0
0	0	1	0	0	0	1	0	0	1	1	0	1	1	1	0
1	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0
0	1	1	1	0	1	0	1	0	0	1	1	0	0	0	1
1	1	1	0	0	0	0	1	1	1	1	0	0	1	0	1
1	1	1	1	0	1	1	1	0	1	0	1	0	0	1	1
0	1	1	1	1	1	1	0	0	1	0	1	1	0	1	1
1	0	1	1	0	1	1	1	1	0	0	0	1	1	0	1

Table 1: Samples of the coin flip data

experiments, the expectation under the model q_n for the unlabeled data did not match the constraints learned from the labeled data. In this case, we have no guarantees that the equality will hold, and can therefore say nothing about the model when the equality breaks.

4 Experiment

We used a synthetic coin flip example adopted from [3] to test our methods. Let $y \in 0, 1$ be the result of a fair coin flip. x is a vector of $a + b$ bits, constructed as follows: the first a bits are 'relevant bits'. With probability $0.5 * 0.9^{(i-1)}, i = 1 \dots a$ bit i equals y, otherwise it is a random bit. The last b bits are all random bits. The motivation of the design is that a good feature selection program will pick the first a bits in that order and exclude the last b bits. Table 1 is a sample of (x, y) pairs for $a = 5, b = 10$.

4.1 The baseline

As a sanity check, we run the traditional feature selection procedure [2] on this problem with $a = 5, b = 10$. The training set size is 50, test set size is 2000. The candidate features are of the form

$$f_i(x, y) = \begin{cases} 1 & \text{if bit } i \text{ of } x \text{ equals } y, \forall i = 1 \dots n \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

We start from a uniform model (with no feature in it), add the candidate feature with the largest heuristic gain with respect to the current model to create a new model, and iterate until we have added all features into the model. Figure 1 shows the results of 20 trials, with error bars of two standard deviations:

- The upper left plot shows, on average, which bits are selected at what iteration. As expected, the first few iterations tend to select relevant bits.

- The upper right plot shows the average heuristic gain of the feature selected at different iterations. Obviously we are selecting 'good' features, i.e. features with large gain, first. This is correct.

- The lower left plot are the loglikelihood of the training (blue) and test (red) set respectively, at different iterations. The training set loglikelihood increases monotonically with the number of features in the model, but the test set loglikelihood drops after an initial increase. This is exactly overfitting: our models do well on the training set but can't generalize.

- The lower right plot shows the classification error rate of the training and test set. Again we see overfitting.

4.2 Results of method I

Encouraged by the initial result, we began to implement method I with an unlabeled data set of size 2000. However we soon found that training on the unlabeled data with constraints from labeled data is not a very good idea:

- Sometimes training on unlabeled data doesn't converge. It is because the constraints, gathered from the labeled data, are not consistent on the unlabeled data.

- The resulting models do not fit the training set very well. Table 2 compares training on labeled with training on unlabeled. Although they start roughly the same, training on the labeled data eventually overfits the labeled set and achieves a high likelihood. But training on the unlabeled data doesn't. It also results in worse performance on the test data.

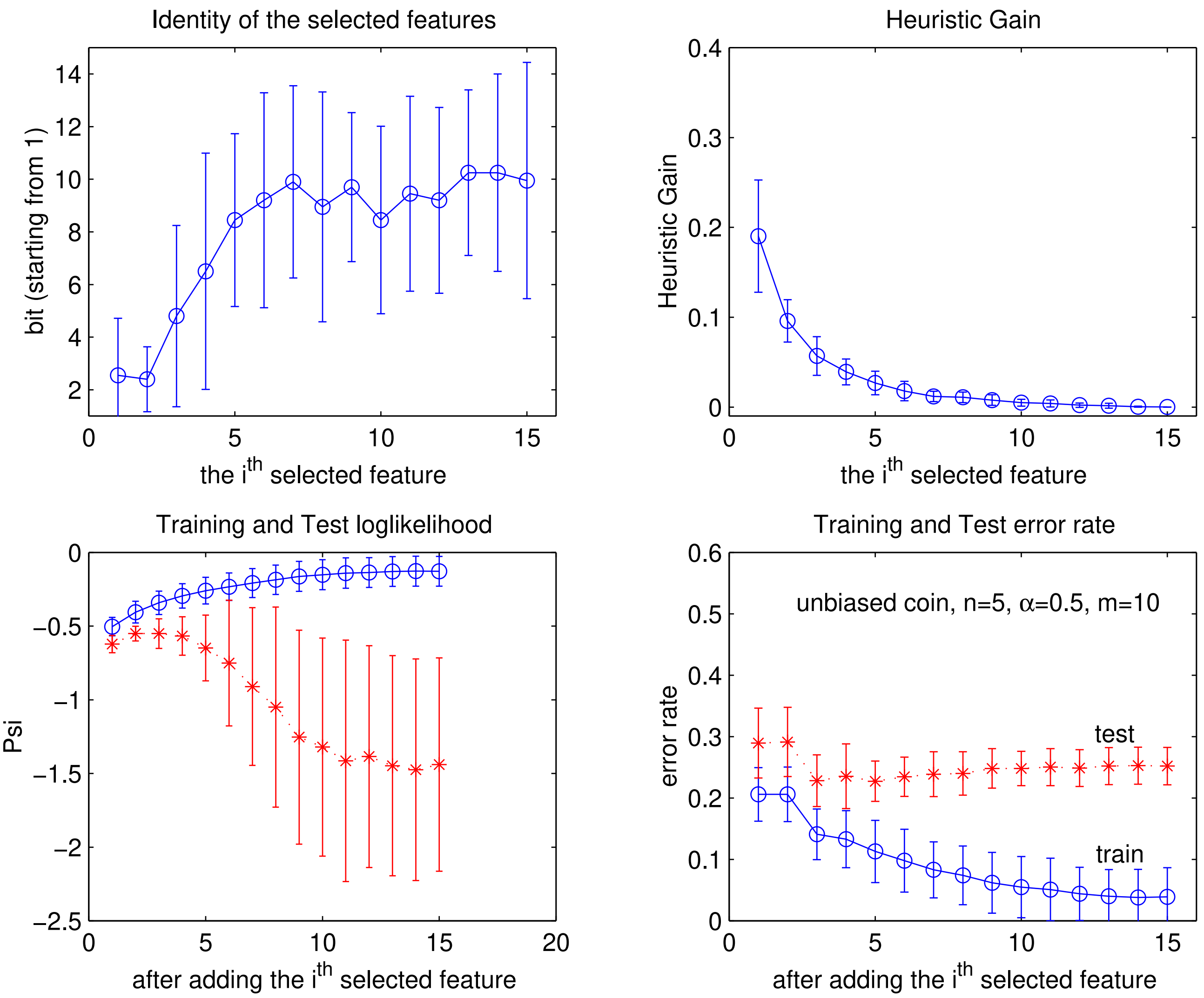

Figure 1: Traditional Feature Selection

strategy	*first feature*	*all features*
train on *labeled* data	-0.5477 ± 0.0723	-0.1063 ± 0.1088
train on *unlabeled* data	-0.5484 ± 0.0748	-1.2530 ± 0.8898

Table 2: The loglikelihood of training set with different training strategies

scheme	baseline	q_n vs. q_{n-1}	q_n vs. q_1	q_n vs. q_0	c.d.m.
threshold	-	0.12	0.37	0.37	-
features	15	10.8	11.1	11.5	9
L LL	-0.01 ± 0.01	-0.04 ± 0.05	-0.04 ± 0.06	-0.03 ± 0.05	-0.03 ± 0.03
L err%	0.0 ± 0.0	1.6 ± 2.3	1.8 ± 2.9	1.3 ± 2.3	1.2 ± 1.2
U LL	-1.06 ± 0.36	-1.05 ± 0.43	-0.82 ± 0.41	-0.86 ± 0.35	-0.93 ± 0.32
U err%	12.1 ± 3.4	14.8 ± 3.6	12.1 ± 3.5	12.0 ± 2.5	12.3 ± 4.1

Table 3: Results of method II

- Because the models do not fit the training set, when the Pythagorean equality breaks, it is almost always because $\hat{D}(\tilde{p}||q_n)$ is large. The fact that the equality does not hold tells us only that $\hat{D}(\tilde{p}||q_n)$ is a bad approximation. But if $\hat{D}(\tilde{p}||q_n)$ is large, knowing that it is a bad approximation is not useful, because the true divergence could actually be smaller.

Because of these findings, we soon abandoned method I and moved on to method II.

4.3 Results of method II

Now we train the models on labeled data only. We run several experiments to compare the following schemes: q_n vs. q_{n-1}, q_n vs. q_1 or q_n vs. q_0. (The reference point is fixed in the latter two schemes.) Since we use approximations, we don't expect the Pythagorean equality to be exact. Instead we specify a predetermined threshold. When the difference between the two sides of the Pythagorean equality exceeds the threshold, we consider the equality to be broken. But we don't stop the model growth all together, rather we throw the current feature away and carry on feature selection until we have exhausted all candidate features.

In addition, we also check the triangle inequality

$$d_{p_u(x)}(q_n, q_{n-1}) \leq \hat{d}_{p_l(x)}(q_n, \tilde{p}) + \hat{d}_{p_l(x)}(\tilde{p}, q_{n-1})$$

based on the classification disagreement metric (1). This is a completely independent test.

Table 3 shows, for each scheme, the threshold used, the average number of features in the final model, the labeled set loglikelihood and error rate, and the unlabeled set loglikelihood and error rate. Unfortunately there is no significant improvement in loglikelihood or error rate, no matter which scheme we use.

5 Discussions

Method 1 didn't work because p^* is not in the feasible set, and the empirical constraints create abnormal distributions on unlabeled data. Method 2 didn't work because the assumption that q_n's are the *true* maximum entropy models is false (we can verify this easily: compute the expectation of features on the unlabeled set with q_n. Most of the time the expectations are different than the empirical constraints).

We feel our failure lies in the inability of the Pythagorean property to establish a connection between the true distribution and the empirical distribution on the labeled set. Recall the Pythagorean property states that

$$D(p^f||q) = D(p^f||q^*) + D(q^*||q)$$

where p^f is any distribution in the feasible set. But the feasible set is defined by a set of empirical constraints, which in general are different from the true constraints. That is to say, in general the true distribution p^* is not in the feasible set. Because of this, the usage of the Pythagorean property is at best a heuristic. (In contrast, in metric-based methods it is always sound to plug in p^*.) If we are only looking for a heuristic, a natural possibility would be to train the models on the labeled data, then compare the divergence between the two models on the labeled data with the divergence on the unlabeled data:

$$D_U(q_n|q_{n-1})\text{vs.}D_L(q_n|q_{n-1})$$

Or if we want a theoretically sound method, we should find a true metric on distributions, or some other equality or inequality that allows us to establish the relation between p^* and other distributions.

Another possible application would be 'early stopping' in maximum entropy model training, i.e. stop training before the empirical constraints are fully satisfied. It has been observed that early stopping results in models that generalize better. This remains to be done in the future.

Acknowledgement

We would like to thank John Lafferty for the original idea and discussions, and Roni Rosenfeld and Tom Mitchell for helpful insights.

References

[1] Dale Schuurmans and Finnegan Southey. Metric-based methods for adaptive model selection and regularization. 2000. Available from `http://www.lpaig.uwaterloo.ca/ dale/papers.html`.

[2] S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing features of random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380–393, April 1997.

[3] Doug Beeferman, Adam Berger, and John Lafferty. Statistical models for text segmentation. *Machine Learning*, 34:177–210, 1999.