Massachusetts Institute of Technology

6.867 Machine Learning, Fall 2006

Problem Set 4: Solutions

1. (a) **(8 points)** We have

$$L(\mathcal{D};\theta) = \prod_{r=1}^{n}\prod_{i=1}^{d} P(x_{r_i}|y_r)P(y_r) \tag{1}$$

where the number of examples is n. We can parameterize $P(y)$ with the parameter ψ, as

$$P(y) = \psi^{\frac{y+1}{2}}(1-\psi)^{\frac{1-y}{2}} \tag{2}$$

This is the same kind of parametrization we have used for $P(x_i|y)$. Eqn 1 thus becomes

$$L(\mathcal{D};\theta,\psi) = \prod_{r=1}^{n}\left[\left(\prod_{i=1}^{d}\theta_{i|y_r}^{\frac{x_{r_i}+1}{2}}(1-\theta_{i|y_r})^{\frac{1-x_{r_i}}{2}}\right)\psi^{\frac{y_r+1}{2}}(1-\psi)^{\frac{1-y_r}{2}}\right] \tag{3}$$

In the above equation, $\theta_{i|+1}$ occurs whenever $x_{r_i}=1$ and $y_r=1$ and $(1-\theta_{i|+1})$ occurs whenever $x_{r_i}=-1$ and $y_r=1$. Similarly, ψ occurs whenver $y_r=1$ and $(1-\psi)$ occurs whenver $y_r=-1$. With this intuition we now have:

$$L(\mathcal{D};\theta,\psi) = \left[\prod_{i=1}^{d}\prod_{y=\{-1,1\}}\theta_{i|y}^{\hat{n}_{iy}(1,y)}(1-\theta_{i|y})^{\hat{n}_{iy}(-1,y)}\right]\left[\psi^{\hat{n}_y(1)}(1-\psi)^{\hat{n}_y(-1)}\right] \tag{4}$$

In the above, we have used the $\hat{n}$ notation used in the lectures; e.g., $\hat{n}_{iy}(1,-1)$ counts the number of examples with $x_{r_i}=1$ and $y_r=-1$. We then have

$$L(\mathcal{D};\theta,\psi)P(\theta,\psi) = L(\mathcal{D};\theta,\psi)P(\theta)P(\psi) \tag{5}$$

We assume a uniform prior on ψ i.e. $P(\psi)=1$. Since $\psi\in[0,1]$, this is already normalized. Then we have:

$$L(\mathcal{D};\theta,\psi)P(\theta,\psi) = \left[\prod_{i=1}^{d}\prod_{y=\{-1,1\}}\theta_{i|y}^{\hat{n}_{iy}(1,y)}(1-\theta_{i|y})^{\hat{n}_{iy}(-1,y)}\right]\left[\psi^{\hat{n}_y(1)}(1-\psi)^{\hat{n}_y(-1)}\right]\times \tag{6}$$

$$\left[\prod_{i=1}^{d}\prod_{y=\{-1,1\}}\frac{1}{B(r^+{+}1,r^-{+}1)}\theta_{i|y}^{r^+}(1-\theta_{i|y})^{r^-}\right] \tag{7}$$

where $B(a,b)=\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(a)}$. We collect the terms togethers

$$L(\mathcal{D};\theta,\psi)P(\theta,\psi) = \frac{1}{Q_r}\left[\prod_{i=1}^{d}\prod_{y=\{-1,1\}}\theta_{i|y}^{\hat{n}_{iy}(1,y)+r^+}(1-\theta_{i|y})^{\hat{n}_{iy}(-1,y)+r^-}\right]\left[\psi^{\hat{n}_y(1)}(1-\psi)^{\hat{n}_y(-1)}\right] \tag{8}$$

where $Q_r=B(r^+{+}1,r^-{+}1)^{2d}$. Thus, $m_{i|y}^+=\hat{n}_{iy}(1,y)+r^+$ and $m_{i|y}^-=\hat{n}_{iy}(-1,y)+r^-$.

(b) **(8 points)** We have

$$P(\theta,\psi|\mathcal{D}) \propto \left[\prod_{i=1}^{d}\prod_{y=\{-1,1\}}\theta_{i|y}^{m^+_{i|y}}(1-\theta_{i|y})^{m^-_{i|y}}\right]\left[\psi^{\hat{n}_y(1)}(1-\psi)^{\hat{n}_y(-1)}\right] \quad (9)$$

The right-hand side (RHS) consists of a product of Beta distributions. To normalize it, we could integrate over each $\theta_{i|y}$ over the range $\theta_{i|y} = [0,1]$, using integration by parts for each such case. However, there's a much simpler method. Since the posterior has the form of a product of Beta distributions, we could directly use the corresponding normalization constant. The normalization constant for the Beta distribution is described in the problem-set. Using it, we have

$$P(\mathcal{D}|\mathcal{F}) = \left[\prod_{i=1}^{d}\prod_{y=\{-1,1\}}B(m^+_{i|y}+1, m^-_{i|y}+1)\right]B(\hat{n}_y(1)+1,\hat{n}_y(-1)+1) \quad (10)$$

i.e.,

$$P(\theta,\psi|\mathcal{D}) = \frac{1}{P(\mathcal{D}|\mathcal{F})}\left[\prod_{i=1}^{d}\prod_{y=\{-1,1\}}\theta_{i|y}^{m^+_{i|y}}(1-\theta_{i|y})^{m^-_{i|y}}\right]\left[\psi^{\hat{n}_y(1)}(1-\psi)^{\hat{n}_y(-1)}\right] \quad (11)$$

If you chose to preserve the constant $1/Q_r$ as part of the initial $P(\theta,\psi|\mathcal{D})$, your answer in Eqn 10 should be multiplied by $1/Q_r$.

(c) **(9 points)** If feature i is included ($\mathcal{F}_2$), the corresponding terms in $P(\mathcal{D}|\mathcal{F})$ will be $B(m^+_{i|1}+1, m^-_{i|1}+1)B(m^+_{i|-1}+1, m^-_{i|-1}+1)$. If it is not included ($\mathcal{F}_1$), there will only be one term θ_i which will combine counts for both $y=1$ and $y=-1$, i.e., the term corresponding to feature i will be $B(\hat{n}_i(1)+r^+ +1, \hat{n}_i(-1)+r^-+1)$.
To choose $\mathcal{F}_1$ over $\mathcal{F}_2$, we need

$$\frac{B(\hat{n}_i(1)+r^+ +1, \hat{n}_i(-1)+r^-+1)}{B(m^+_{i|1}+1, m^-_{i|1}+1)B(m^+_{i|-1}+1, m^-_{i|-1}+1)} > 1 \quad \text{or} \quad (12)$$

or,

$$\log B(\hat{n}_i(1)+r^+ +1, \hat{n}_i(-1)+r^-+1)) \quad (13)$$
$$-\log B(m^+_{i|1}+1, m^-_{i|1}+1) \quad (14)$$
$$-\log B(m^+_{i|-1}+1, m^-_{i|-1}+1) > 0 \quad (15)$$

(d) **(2 points)** At the MLE value, the first derivative is zero (the derivative of a differentiable function is zero at maxima and minima). As such, a first-order expansion will not buy us much— it will only lead to a constant-valued function.

(e) **(6 points)** From previous part, $A_1 = 0$. Let $\Sigma = (-A_2)^{-1}$. Since A_2 is the Hessian (i.e. the matrix of second derivatives) evaluated at a maxima, it is negative definite, so that the negative of its inverse Σ is positive definite. Also, we are given that $|\Sigma| \approx (n^r C(r))^{-1}$. We now have

$$\int L(\mathcal{D};\theta)P(\theta)d\theta = \int L(\mathcal{D};\theta)\cdot 1\cdot d\theta \quad (16)$$
$$= \int \exp(\log L(\mathcal{D};\theta))d\theta \quad (17)$$

$$= \int \exp(\log L(\mathcal{D};\hat{\theta}_{ML}) - \frac{1}{2}(\theta-\hat{\theta}_{ML})^T\Sigma^{-1}(\theta-\hat{\theta}_{ML}))d\theta \quad (18)$$

$$= \left(\int L(\mathcal{D};\hat{\theta}_{ML})d\theta\right)\left(\int \exp(\frac{1}{2}(\theta-\hat{\theta}_{ML})^T\Sigma^{-1}(\theta-\hat{\theta}_{ML}))d\theta\right) \quad (19)$$

$$(20)$$

$L(\mathcal{D};\hat{\theta}_{ML})$ is a constant, i.e., it doesn't depend on θ. Also, the second term looks a lot like a Gaussian distribution. So we have

$$\left(L(\mathcal{D};\hat{\theta}_{ML})\right)\left((2\pi)^{r/2}|\Sigma|^{1/2}\int \frac{1}{(2\pi)^{r/2}|\Sigma|^{1/2}}\exp(\frac{1}{2}(\theta-\hat{\theta}_{ML})^T\Sigma^{-1}(\theta-\hat{\theta}_{ML}))d\theta\right) \quad (21)$$

$$\approx L(\mathcal{D};\hat{\theta}_{ML})\cdot(2\pi)^{r/2}(n^rC(r))^{-1/2}\cdot 1 \quad (22)$$

$$\approx L(\mathcal{D};\hat{\theta}_{ML})\left(\frac{2\pi}{n}\right)^{r/2}C_1(r) \quad (23)$$

(f) **(2 points)** Taking the log of the expression from the previous part, we have:

$$\log P(\mathcal{D};\mathcal{F}_r) \approx \log L(\mathcal{D};\hat{\theta}_{ML}) - \frac{r}{2}\log n + \frac{r}{2}\log 2\pi + C_2(r) \quad (24)$$

As $n\to\infty$, the terms that depend only on r and not on n can be ignored. So that $\lim_{n\to\infty}\log P(\mathcal{D};\mathcal{F}_r)$ becomes

$$\log L(\mathcal{D};\hat{\theta}_{ML}) - \frac{r}{2}\log n \quad (25)$$

2. (a) The likelihood is:

$$\mathcal{L}(D;\Theta) = \prod_{k=0}^{m}\prod_{i=t_{k-1}}^{t_k} N(x_i;\mu_k,\Sigma_k). \quad (26)$$

The log-likelihood is:

$$\ell(D;\Theta) = \sum_{k=0}^{m}\sum_{i=t_{k-1}}^{t_k}\log N(x_i;\mu_k,\Sigma_k); \quad (27)$$

Furthermore, the number of free variables in $m+1$, d-dimensional multivariate Gaussians is $m(d+d(d+1)/2)$. Consequently, the BIC is:

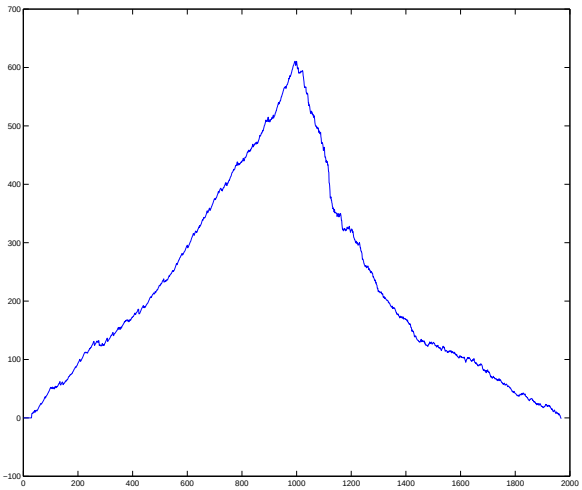
$$\text{BIC} = \sum_{k=0}^{m}\sum_{i=t_{k-1}}^{t_k-1}\log N(x_i;\mu_k,\Sigma_k) - \left(\frac{d+d(d+1)/2}{2}\right)\sum_{k=0}^{m}\log(t_k-t_{k-1}).$$

It is worth noting that:

$$\sum_{i=t_{k-1}}^{t_k-1}\log N(x_i;\mu_k,\Sigma_k) = -\frac{t_k-t_{k-1}}{2}\left(\log|\Sigma| + d\log(2\pi)+1\right). \quad (28)$$

(b)

```
function [bestat,topscore,scores]=SPLIT(X)
·   N = size(X,1);
·   d = size(X,2);
·   λ = 1;
·   s1 = 0 ; s2 = sum(X);
·   S1 = 0 ; S2 = 0;
·   for i = 1 : N
·   ·   S2 = S2 + X(i,:)'X(i,:);
·   end
·   μ = s2/N;
·   Σ = S2/N − μ'μ;
·   topscore = −10;
·   bestat = −1;
·   scores = [];
·   for i = 1 : N − 2
·   ·   s1 = s1 + X(i,:);
·   ·   S1 = S1 + X(i,:)'X(i,:);
·   ·   s2 = s2 − X(i,:);
·   ·   S2 = S2 − X(i,:)'X(i,:);
·   ·   μ1 = s1/i;
·   ·   Σ1 = S1/i − μ1'μ1;
·   ·   μ2 = s2/(N − i);
·   ·   Σ2 = S2/(N − i) − μ2'μ2;
·   ·   if i > 30 and i < N − 30
·   ·   ·   score = N log(det(Σ)) − i log(det(Σ1))
·   ·   ·   ·   − (N − i) log(det(Σ2))
·   ·   ·   ·   − λ/2(d + d(d + 1)/2) log(N);
·   ·   ·   if score > topscore
·   ·   ·   ·   bestat = i;
·   ·   ·   ·   topscore = score;
·   ·   ·   end
·   ·   ·   scores(i,:) = [score, det(Σ), det(Σ1), det(Σ2)];
·   ·   end
·   end
end
≫ LOAD -ascii 'data1'
≫ C = data1;
≫ [bestat,topscore,scores] = SPLIT(C);
```



(c)

$\gg$ **load** -ascii 'cepstra1.mat'
$\gg C =$ cepstra1;
$\gg$ MULTISPLIT(C)
ans =

- 149
- 194
- 291
- 421
- 492
- 556
- 668
- 738
- 1470
- 1587
- 1693
- 1751
- 1840

$\gg$ **load** -ascii 'cepstra2.mat'
$\gg C =$ cepstra2;
$\gg$ MULTISPLIT(C)
ans =
· 32
· 198
· 230
· 285
· 449
· 514
· 684
· 813
· 852
· 897
· 1040
· 1197
· 1229
· 1397
· 1534
· 1683

3. (a) Take the negative derivative of the loss function to get the weights:

$$\frac{e^{-z}}{1+e^{-z}}. \tag{29}$$

The numerator and denominator are both positive and the numerator is less than the denominator. Thus, the quotient is between 0 and 1. Given that the unnormalized weights are bounded, examples that are badly misclassified and those that are just barely misclassified will end up with comparable weights after normalization.

(b) The value of $\hat{\alpha}_1$ is infinite. Increasing α_1 will decrease all the training losses since $y_t h(\mathbf{x}_t; \hat{\theta}_1) > 0$ for all t.

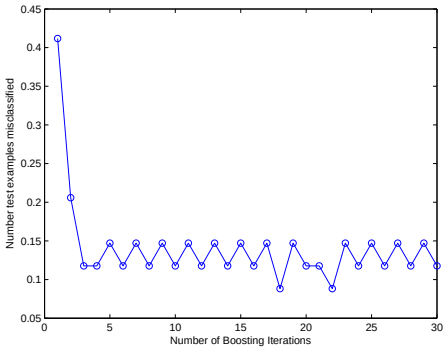
(c) There are two ways to show this. First, by construction when β is fixed. It suffices to provide a set of points such that an ensemble with the radial basis learners can classify them in all possible ways. We will use n base learners, each associated with one training point. The points can be placed far enough apart such that the only relevant contribution to the ensemble output comes from the base learner associated with each point. Since the base learners reproduce the training labels for individual points, so will the ensemble.

Another way is to use the result in problem set 2 that the gram matrix for the radial basis kernel is invertible so that the discriminant function

$$h(\mathbf{x}; \theta) = \sum_{t=1}^{n} \alpha_t y_t \exp(-\beta \|\mathbf{x} - \mathbf{x}_t\|^2) \tag{30}$$

can be chosen to take any values over n-points $\mathbf{x}_1, \ldots, \mathbf{x}_n$. Strictly speaking we'd have to show, in addition, that α_t's in the above expression can be all non-negative. The product $\alpha_t y_t$ is not constrained if we can choose $y_t \in \{-1, 1\}$ for each base learner (y_t here need not be the label we aim to reproduce with $h(\mathbf{x}_t; \theta)$).

(d) It does not overfit; the test error decreases initially, but does not increase again after many iterations as it would if it were overfitting.



(e) Replace lines 9 and 10 in `call_boosting.m` with:

- [y_est,sum_of_alpha]=eval_boost(model(1:k),data.xtrain);
- err(k)=sum(y_est.*data.ytrain/sum_of_alpha≤0.5)/length(data.ytrain);

The margin errors for $\rho = 0.1$ tend to decrease.