Data Mining CS573: Midterm

March 11, 2016

Yu-Chen Chang

Contents

Problem 1

a.

i. Naive Bayes Classifier:

From the naive bayes classifier, we have the formula:
$P(C|\mathbf{X}) = \frac{P(\mathbf{X}|C)P(C)}{P(\mathbf{X})} \propto P(\mathbf{X}|C)P(C)$ (Bayes rule), where C is the class random variable and $\mathbf{X}$ is the attribute random vector.

In NBC, there is an assumption that attributes are conditionally independent given the class, Therefore, we have the naive Bayes classifier:
$P(C|\mathbf{X}) \propto P(\mathbf{X}|C)P(C) \propto \prod_{i=1}^{m} P(X_i|C)P(C)$, where m is the number of attributes and X_i is the i-th attribute random variable.

Because we don't know the distribution of $P(X_i|C)$ and $P(C)$, Therefore, we need likelihood function to determine unknown parameters based on known outcomes. Assume the data D are independently sampled from the same distribution. Let $D = [\mathbf{x}_1, ..., \mathbf{x}_n]$, where n is the number of samples:

$$L(\theta|D) = \prod_{i=1}^{n} P(\mathbf{x}_i, c_i|\theta) \quad \text{(general likelihood)} \tag{1}$$

$$\propto \prod_{i=1}^{n} P(\mathbf{x}_i|c_i, \theta)P(c_i|\theta) \quad \text{(Bayes rule)} \tag{2}$$

$$\propto \prod_{i=1}^{n}\prod_{j=1}^{m} P(x_{ij}|c_i, \theta)P(c_i|\theta) \quad \text{(Naive assumption)} \tag{3}$$

We apply Maximum Likelihood estimation to learn the best parameters by finding the value θ that maximizes likelihood:

$$\hat{\theta}_{MLE} = arg\ \max_{\theta} L(\theta) \tag{4}$$

For Multinomials, Let $A \in \{1, ..., k\}$ be a discrete random variable with k values, where $P(A = j) = \theta_j$. Then P(A) is a multinomial distribution:

$$P(A|\theta) = \prod_{j=1}^{k} \theta_j^{I(A=j)} \text{ , where } I(A = j) \text{ is an indicator function.} \tag{5}$$

The likelihood for a data set D is:

$$P(D|\theta) = \prod_{i=1}^{n}\prod_{j=1}^{k} \theta_j^{I(A=j)} = \prod_{j} \theta_i^{n_j} \tag{6}$$

Therefore, by using Lagrange multipliers, the maximum likelihood estimates for each parameter are:

$$\hat{\theta_j} = \frac{n_j}{n} \tag{7}$$

which means that in multinomial case, MLE can be determined analytically by counting.

For continuous inputs X_i, the common way to represent the distributions $P(X_i|Y)$ to assume that

Problem 1 continued on next page...

for each possible discrete value y_k of Y, the distribution of each continuous X_i is Gaussian, and is defined by a mean and standard deviation specific to X_i and y_k.

$$\mu_{ik} = E[X_i|Y = y_k] \tag{8}$$
$$\sigma_{ik}^2 = E[(X_i - \mu_{ik})^2|Y = y_k] \tag{9}$$

Again, by MLE, we get:

$$\hat{\mu}_{ik} = \frac{1}{\sum_j \delta(Y^j = y_k)} \sum_j X_i^j \delta(Y^j = y_k) \tag{10}$$
$$\hat{\sigma}_{ik}^2 = \frac{1}{\sum_j \delta(Y^j = y_k)} \sum_j (X_i^j - \hat{\mu_{ik}})^2 \delta(Y^j = y_k) \tag{11}$$

Then we can estimate continuous attributes using Gaussian distribution with $\hat{\mu}_{ik}$ and $\hat{\sigma}_{ik}^2$.

In this question, we are given 11 attributes.

1. Record Number
2. Amount Requested
3. Interest Rate Percentage
4. Loan Length in Months
5. Loan Title
6. Loan Purpose
7. Monthly Payment
8. Total Amount Funded
9. Debt-To-Income Ratio Percentage
10. FICO Range
11. Status

The Record Number is used as id and will not be considered as an attribute and Status is the classification goal that we are interested in and used as the class random variable. Therefore, the potential attributes are from the 2 to 10 entry, which forms our attribute random vector.

In the step of classifying out-of-sample items, we will use the above shown formula to calculate the P(C|X) and compare $P(C = c_1|X)$ with $P(C = c_2|X)$ to see whether the out-of-sample with its attributes given in X should belong to c_1 or c_2 class.

ii. From the MLE, we have the formula

$$\hat{\theta_j} = \frac{n_j}{n} \tag{12}$$

The prior is estimated from the dataset by counting the number of each class among the entire dataset. However, if the real value prior is far from the estimated one, it will have significant impacts on the correctness of the prediction. For example, if we have a dataset with half of people with cancer and other half are healthy while in really life the probability that a person has a cancer is nearly 0.01%, then in this situation the prior will be estimated wrong (50%), which should be 0.01% for cancer class and 99.99% for healthy class, and cause large false positive in this prediction. Therefore, we can see that the wrong prior in NBC will cause either false positive or false negative to increase depending on the difference between real prior and the estimated one. That's the reason why prior in NBC is important.

Problem 1 continued on next page...

b. Logistic Regression:

i. In logistic regression, we make the assumption that

$$log\frac{P(\mathbf{x}, y=1)}{P(\mathbf{x}, y=0)} = \mathbf{w}^T\mathbf{x} + w_0 \tag{13}$$

which is equivalent to

$$P(y=1|\mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x}+w_0)}} \tag{14}$$

$$P(y=0|\mathbf{x}) = \frac{e^{-(\mathbf{w}^T\mathbf{x}+w_0)}}{1+e^{-(\mathbf{w}^T\mathbf{x}+w_0)}} \tag{15}$$

Using the canonical representation of the data (adding a dummy feature of value 1 to each input vector), we have

$$P(y=1|\mathbf{x}) = g(\mathbf{x}, \mathbf{w}) = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}}} \tag{16}$$

$$P(y=0|\mathbf{x}) = 1 - g(\mathbf{x}, \mathbf{w}) = \frac{e^{-\mathbf{w}^T\mathbf{x}}}{1+e^{-\mathbf{w}^T\mathbf{x}}} \tag{17}$$

These equations mean that given a training data set $\{(\mathbf{x}_i, y_i) : i = 1, ..., N\}$, and $\mathbf{x}_i \in R^{d+1}$, where N is the total number of training examples and d is the original feature dimension, the learning goal is to find the optimal weight vector $\mathbf{w}$.
The next step is to learn the parameters by using MLE. The log likelihood function is as follows:

$$L(\mathbf{w}) = \sum_{i=1}^{N} log\ p(y_i|\mathbf{x}_i) = \sum_{i=1}^{N} log\ g(\mathbf{x}_i, \mathbf{w})^{y_i}(1 - g(\mathbf{x}_i, \mathbf{w}))^{1-y_i} \tag{18}$$

Taking gradient of L with respect to $\mathbf{w}$, we have

$$\sum_{i=1}^{N}(y_i - g(\mathbf{x}_i, \mathbf{w}))\mathbf{x}_i = \mathbf{\Phi}^T(\mathbf{y} - g(\mathbf{x}, \mathbf{w})) \tag{19}$$

Now we use Newton-Raphson update for gradient descent

$$\mathbf{H} = \sum_{i=1}^{N} g(\mathbf{x}_i, \mathbf{w})(1 - g(\mathbf{x}_i, \mathbf{w}))\mathbf{x}_i\mathbf{x}_i^T \tag{20}$$

we donote it as:

$$\mathbf{H} = \sum_{i=1}^{N} g(\mathbf{x}_i, \mathbf{w})(1 - g(\mathbf{x}_i, \mathbf{w}))\mathbf{x}_i\mathbf{x}_i^T = \mathbf{\Phi}^T\mathbf{R}\mathbf{\Phi} \tag{21}$$

where $R_{nn} = g(\mathbf{x}_i, \mathbf{w})(1 - g(\mathbf{x}_i, \mathbf{w})$
Then the iterative parameter update is

$$\mathbf{w}^{(\text{new})} = \mathbf{w}^{(\text{old})} + (\mathbf{\Phi}^T\mathbf{R}\mathbf{\Phi})^{-1}\mathbf{\Phi}^T(\mathbf{y} - g(\mathbf{x}, \mathbf{w})) \tag{22}$$

$$= (\mathbf{\Phi}^T\mathbf{R}\mathbf{\Phi})^{-1}\mathbf{\Phi}^T\mathbf{R}\mathbf{z} \tag{23}$$

where $\mathbf{z}$ is an N-dimensional vector with elements

$$\mathbf{z} = \mathbf{\Phi}\mathbf{w}^{(\text{old})} - \mathbf{R}^{-1}(g(\mathbf{x}, \mathbf{w}) - \mathbf{y}) \tag{24}$$

In this question, we are given 11 attributes.

1. Record Number
2. Amount Requested
3. Interest Rate Percentage
4. Loan Length in Months
5. Loan Title
6. Loan Purpose
7. Monthly Payment
8. Total Amount Funded
9. Debt-To-Income Ratio Percentage
10. FICO Range
11. Status

The Record Number is used as id and will not be considered as an attribute and Status is the classification goal that we are interested in and used as the class random variable. Therefore, the potential attributes are from the 2 to 10 entry, which forms our attribute random vector. We also feed weight vector $\mathbf{w}$ to the logistic regression to train our model.

Once the model is trained with the $\mathbf{w}$, in the step of classifying out-of-sample items, we will use the above shown formula to calculate the $\mathrm{P}(\mathbf{y}|\mathbf{x})$ and compare $P(y = 0|\mathbf{x})$ with $P(y = 1|\mathbf{x})$ to see whether the out-of-sample with its attributes given in X should belong to $y = 0$ or $y = 1$ class.

ii.

c. i. To find support point for SVM (assuming linearly separable data), Back to our linear model with non-linear features ϕ.

$$y(\mathbf{x}) = \mathbf{w}^T\phi(\mathbf{x}) + b \tag{25}$$

For two classes, if class $t_n \in \{-1, 1\}$ of item $\mathbf{x}_n$ Then $t_n y(\mathbf{x}_n) > 0$ means correctly classified.
Also, the distance to the hyperplane is:

$$\frac{y(\mathbf{x})}{||\mathbf{w}||} \tag{26}$$

Thus, distance of $\mathbf{x}_n$ from decision hyperplane is the maximum minimum distance.

$$\underset{\mathbf{w},b}{arg\ max}\left\{\frac{1}{||\mathbf{w}||}\underset{n}{min}[t_n(\mathbf{w}^T\phi(\mathbf{x}_n) + b)]\right\} \tag{27}$$

However, because the problem is too complicated to compute, so we recast problem into another optimization problem. Then the original problem becomes (as $arg\ max||\mathbf{x}||^{-1} = arg\ min||\mathbf{w}||^2$) a quadratic programming problem.

$$\underset{\mathbf{w},b}{arg\ min}\frac{1}{2}||\mathbf{w}|| \tag{28}$$

s.t.

$$t_n(\mathbf{w}^T\phi(\mathbf{x}_n) + b) \geqslant 1, \quad n = 1, ..., N \tag{29}$$

We can solve constrained optimization problem via Lagrange multipliers $a_n \geqslant 0$

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{n=1}^{N} a_n\left\{t_n(\mathbf{w}^T\phi(\mathbf{x}_n) + b) - 1\right\} \tag{30}$$

Setting derivative of $L(\mathbf{w}, b, \mathbf{a})$ w,r,t $\mathbf{w}$ and b to zero, we get:

$$\mathbf{w} = \sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n) \tag{31}$$

$$0 = \sum_{n=1}^{N} a_n t_n \tag{32}$$

Eliminating $\mathbf{w}$ and b from previous equation using these conditions:

$$\widetilde{L}(a) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{n=1}^{N} a_n a_m t_n t_m k(\mathbf{x}_n, \mathbf{x}_m) \tag{33}$$

s.t.

$$a_n \geqslant 0, \quad n = 1, ..., N \tag{34}$$

$$\sum_{n=1}^{N} a_n t_n = 0 \tag{35}$$

where $k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T\phi(\mathbf{x}')$, and k is the kernel.
For Linear kernels:

$$k(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{x}' \tag{36}$$

Problem 1 continued on next page...

For Gaussian kernels:

$$k(\mathbf{x}, \mathbf{x}') = exp(\frac{1}{2}||\mathbf{x} - \mathbf{x}'||^2) \tag{37}$$

For non-linearly separable data, the

$$a_n \geqslant 0, \quad n = 1, ..., N \tag{38}$$

becomes

$$0 \leqslant a_n \leqslant \mathbf{C}, \quad n = 1, ..., N \tag{39}$$

where $\mathbf{C}$ can be seen as a penalty for misclassification.

In this question, we are given 11 attributes.

1. Record Number
2. Amount Requested
3. Interest Rate Percentage
4. Loan Length in Months
5. Loan Title
6. Loan Purpose
7. Monthly Payment
8. Total Amount Funded
9. Debt-To-Income Ratio Percentage
10. FICO Range
11. Status

The Record Number is used as id and will not be considered as an attribute and Status is the classification goal that we are interested in and used as the class random variable (t_n in the previous formula). Therefore, the potential attributes are from the 2 to 10 entry, which forms our attribute random vector. We also feed weight vector $\mathbf{w}$, kernel type $k(\mathbf{x}, \mathbf{x}')$ and $\mathbf{C}$ to the train our SVM model.

Once the model is trained, in the step of classifying out-of-sample items, we will use the above shown formula($\mathbf{w}^T\phi(\mathbf{x}) + b$) to see its sign to determine which class the out-of-sample with its attributes given in X should belong to $\{-1, 1\}$.

ii.

Problem 2

a.

b.