Fall2016 HomeWork2

Hema Venkata Krishna Giri Narra

October 3, 2016

1 Logistic Regression

1.1 Negative log likelihood for binary model

$L(w) = -\log(\prod_{i=1}^{n} P(\frac{Y=y_i}{X=x_i}))$
$\rightarrow -(\sum_{i=1}^{n} log(\sigma(b + w^T x_n)^{y_n}[1 - \sigma(b + w^T x_n]^{1-y_n}))$
$\rightarrow -(\sum_{i=1}^{n} y_n log(\sigma(b + w^T x_n) + (1 - y_n)\log[1 - \sigma(b + w^T x_n)])$

1.2 Gradient Descent and update

$\frac{\partial L}{\partial w} = -(\sum_{i=1}^{n} y_n[1 - \sigma(w^T x_n)]x_n - (1 - y_n)\sigma(w^T x_n)x_n)$
$\rightarrow \sum_{i=1}^{n}(\sigma(w^T x_n) - y_n)x_n$
Updating w
$w^{(t+1)} \leftarrow w^{(t)} - \eta \sum_{i=1}^{n}(\sigma(w^T x_n) - y_n)x_n$
This solution will converge to a global minimum because the Hessian matrix, H, (matrix of second order derivatives) is positive definite.
So the solution is convex and has a global minimum.
$H = \sum_{i=1}^{n} x_n {x_n}^T \sigma(w^T x_n)(1 - \sigma(w^T x_n))$
Since $\sigma\epsilon(0, 1), x_n {x_n}^T$ is positive,H is positive

1.3 Negative log likelihood for multi-class classification

$L(w_1, .., w_K) = -\sum_n \log(P(\frac{y_n}{x_n})$
Let $y_{nk} = 1$ if $y_n = k$ else 0
$\rightarrow L(w_1, .., w_K) = -\sum_n \sum_k y_{nk} \log(P(\frac{C_k}{x_n}))$

Replacing $P(\frac{C_k}{x_n})$ given posterior probability:
$\rightarrow \sum_n \sum_k y_{nk}(\log(1 + \sum_{t=1}^{K-1} exp(w_t^T x_n)) - (w_k^T x_n))$

1.4 Gradient descent and update

$\frac{\partial L(w_1,..,w_K)}{\partial w_i} = \sum_n \sum_k y_{nk} \frac{exp(w_i^T x_n)}{1+\sum_{t=1}^{K-1} exp(w_t^T x_n)} - \sum_n y_{ni} x_n$
$\rightarrow \sum_n \sum_k y_{nk} P(\frac{Y=y_i}{X=x_n}) - \sum_n y_{ni} x_n$
Update rule is:
$w_i^{(t+1)} \leftarrow w_i^{(t)} - \eta(\sum_n \sum_k y_{nk} P(\frac{Y=y_i}{X=x_n}) - \sum_n y_{ni} x_n)$

2 Gaussian discriminant

2.1 MLE

The log likelihood function is:
$\rightarrow \sum_n \log p(x_n, y_n)$
For simplified notation, let class $y_n$be either 1 or 0 instead of 2 and 1. p_0, p_1 be $P(y=0), P(y=1)$
$\rightarrow \sum_n (y_i \log p_1 + y_i \log N(u_2, \sigma_2^2) + (1-y_i) \log p_0 + (1-y_i) \log N(u_1, \sigma_1^2))$
To estimate p_1, Also $p_0 = 1 - p_1$.
Terms in above expression having p_1
The terms are $\sum_n (y_i \log p_1 + (1-y_i) \log 1 - p_1)$
Taking derivative of terms and setting to Zero:
$p_1 = \frac{N_1}{N}$ where N_1 number of samples with $y_i = 1$
Same procedure for p_0. $p_0 = \frac{N_0}{N}$
To estimate u_1 (exact same procedure for u_2)
Taking derivative over terms with u_1 and setting to zero
$u_1 = \frac{\sum_{y_i=1} x_i}{N_1}$
Similar form for u_2
$u_2 = \frac{\sum_{y_i=0} x_i}{N_0}$
To estimate σ_1 (exact same procedure for $sigma_2$)
Taking derivative over terms with σ_1 and setting to zero we can get σ_1

2.2 Posterior probability follows a Logistic function

$P(\frac{y=1}{x}) = \frac{1}{1+\frac{P(x/y=c_2)P(y=c_2)}{P(x/y=c_1)P(y=c_1)}}$
Let p_1, p_2 be $P(y=c_1), P(y=c_2)$

$\rightarrow \frac{1}{1+\frac{N(u_2,\Sigma)p_2}{N(u_1,\Sigma)p_1}}$
$\rightarrow \frac{1}{1+E}$
After substituting Multivariate gaussian distribution forms in E, $\log E$ is:
$\rightarrow \log\frac{p_2}{p_1} - \frac{1}{2}x^T\Sigma^{-1}x + u_2^T\Sigma^{-1}x - \frac{1}{2}u_2^T\Sigma^{-1}u_2 + \frac{1}{2}x^T\Sigma^{-1}x - u_1^T\Sigma^{-1}x + \frac{1}{2}u_1^T\Sigma^{-1}u_1$
$\rightarrow -(u_1 - u_2)^T\Sigma^{-1}x + \frac{1}{2}u_1^T\Sigma^{-1}u_1 - \frac{1}{2}u_2^T\Sigma^{-1}u_2 + \log\frac{p_2}{p_1}$
$\rightarrow -\theta^T x + const$

3 Programming Assignment

3.1 Linear Regression and Ridge Regression

The performance of Linear and Ridge regressions is almost similar.

3.1.1 Linear Regression

MSE for training set: 20.95
MSE for test set : 28.38

3.1.2 Ridge Regression

$\lambda = 0.01$
MSE for training set: 20.95
MSE for test set : 28.38
$\lambda = 0.1$
MSE for training set: 20.95
MSE for test set : 28.39
$\lambda = 1.0$
MSE for training set: 20.95
MSE for test set : 28.49

3.1.3 Cross-Validation

The best λ is obtained by uniformly searching in the [0.0001, 10] in 2000 steps and observing the Average MSE over Cross-Validation sets. To reduce run time, in the submitted program this is set to 10 steps.
It is observed in every program run that λ value of 10 provides the lowest average MSE.
On the training set the MSE for $\lambda = 10$ is 29.64

3.2 Feature Selection

The results for Residue based feature selections is identical to Brute force based featue selection. These two approaches provide lower MSE than the feature selection based on 4 highest correlated features.

3.2.1 Highest 4 correlated features in absolute values

The features with highest absolute correlation in order are: LSTAT, RM, PTRATIO, INDUS
MSE for training set: 26.41
MSE for test set : 31.64

3.2.2 Residue based recursive feature selestion

The features with highest absolute correlation in order are: LSTAT, RM, PTRATIO, CHAS
MSE for training set: 25.11
MSE for test set : 34.67

3.2.3 Brute force

The features with highest absolute correlation in order are: CHAS, RM, PTRATIO, LSTAT
MSE for training set: 25.11
MSE for test set : 34.67

3.3 Polynomial Feature Selection

MSE for training set: 5.06
MSE for test set : 23.79
MSE's for both the training and test set reduced.