CSCI 567–Homework 2

Chengliang Dong (Meng Liu, Kan Wang, Ye Feng and Peehoo Dewan) Friday 10th October, 2014

Notations for all answers of this homework, note that it may conflict with some notations for given in the quesiton. But for consistency of solution, this version of notation is applied instead.

- m : number of samples
- n : number of features
- feature vector: $\mathbf{X}^{(i)} = \begin{pmatrix} 1 \\ x_1^i \\ \vdots \\ x_n^i \end{pmatrix} \in \mathbb{R}^{n+1}$
- outcome vector: $\mathbf{Y} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{pmatrix} \in \mathbb{R}^{m}$
- weight vactor: $\boldsymbol{\beta} = \begin{pmatrix} \beta^{(0)} \\ \beta^{(1)} \\ \vdots \\ \beta^{(n)} \end{pmatrix} \in \mathbb{R}^{n+1}$
- define matrix: $\mathbf{X} = \begin{pmatrix} (\mathbf{X}^{(1)})^T \\ (\mathbf{X}^{(2)})^T \\ \vdots \\ (\mathbf{X}^{(m)})^T \end{pmatrix} \in \mathbb{R}^{m \times (n+1)}$
- hypothesis function: $h_{\boldsymbol{\beta}}(\mathbf{X}) = \mathbf{X}\boldsymbol{\beta}$

1 Linear Regression

1.1 Regression with heterogeneous noise

1.1.1 Log likelihood

Notation, obviously, the error follows multivariate Gaussian $N(0, \Sigma)$, where $\Sigma = \begin{pmatrix} \sigma_1^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_m^2 \end{pmatrix}$.

With the above definition, we have,

$$L(\boldsymbol{\beta}) = log(2\pi)^{-\frac{m}{2}} |\Sigma|^{-\frac{1}{2}} exp(-\frac{1}{2}(\mathbf{Y} - h_{\boldsymbol{\beta}}(\mathbf{X}))^T \Sigma^{-1} (\mathbf{Y} - h_{\boldsymbol{\beta}}(\mathbf{X})))$$

1.1.2 MLE

from 1.1.1, we have,

$$
\begin{aligned}
L(\boldsymbol{\beta}) &= log(2\pi)^{-\frac{m}{2}}|\Sigma|^{-\frac{1}{2}}exp(-\frac{1}{2}(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))^T\Sigma^{-1}(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X})))\\
&= log(2\pi)^{-\frac{m}{2}}|\Sigma|^{-\frac{1}{2}} - \frac{1}{2}(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))^T\Sigma^{-1}(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))\\
\nabla_{\boldsymbol{\beta}}L(\boldsymbol{\beta}) &= -\frac{1}{2}\nabla_{\boldsymbol{\beta}}(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))^T\Sigma^{-1}(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))\\
&= -\frac{1}{2}\nabla_{\boldsymbol{\beta}}tr(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))^T\Sigma^{-1}(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))\\
&= -\frac{1}{2}(\nabla_{\boldsymbol{\beta}}tr\mathbf{Y}^T\Sigma^{-1}\mathbf{Y} - \nabla_{\boldsymbol{\beta}}tr\boldsymbol{\beta}^T\mathbf{X}^T\Sigma^{-1}\mathbf{Y} - \nabla_{\boldsymbol{\beta}}tr\mathbf{Y}^T\Sigma^{-1}\mathbf{X}\boldsymbol{\beta} + \nabla_{\boldsymbol{\beta}}tr\boldsymbol{\beta}^T\mathbf{X}^T\Sigma^{-1}\mathbf{X}\boldsymbol{\beta})\\
&= -\frac{1}{2}(-2\nabla_{\boldsymbol{\beta}}tr\boldsymbol{\beta}\mathbf{Y}^T\Sigma^{-1}\mathbf{X} + \nabla_{\boldsymbol{\beta}}tr\boldsymbol{\beta}\boldsymbol{\beta}^T\mathbf{X}^T\Sigma^{-1}\mathbf{X})\\
&= -\frac{1}{2}(-2\nabla_{\boldsymbol{\beta}}tr\boldsymbol{\beta}\mathbf{Y}^T\Sigma^{-1}\mathbf{X} + \nabla_{\boldsymbol{\beta}}tr\boldsymbol{\beta}\mathbf{I}\boldsymbol{\beta}^T\mathbf{X}^T\Sigma^{-1}\mathbf{X})\\
&= (\mathbf{Y}^T\Sigma^{-1}\mathbf{X})^T - \frac{1}{2}(\mathbf{X}^T\Sigma^{-1}\mathbf{X}\boldsymbol{\beta}\mathbf{I} + (\mathbf{X}^T\Sigma^{-1}\mathbf{X})^T\boldsymbol{\beta}\mathbf{I}^T)\\
&= \mathbf{X}^T\Sigma^{-1}\mathbf{Y} - \mathbf{X}^T\Sigma^{-1}\mathbf{X}\boldsymbol{\beta}\\
&:= 0\\
\boldsymbol{\beta} &= (\mathbf{X}^T\Sigma^{-1}\mathbf{X})^{-1}(\mathbf{X}^T\Sigma^{-1}\mathbf{Y})
\end{aligned}
$$

1.2 Smooth Coefficient

1.2.1 Regularizer

Define

$$
\mathbf{A} = \begin{pmatrix}
1 & -1 & 0 & \cdots & 0 & 0\\
0 & 1 & -1 & \cdots & 0 & 0\\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
0 & 0 & 0 & \cdots & 1 & -1\\
0 & 0 & 0 & \cdots & 0 & 0
\end{pmatrix} \tag{1}
$$

Therefore, we have,

$$
\begin{aligned}
g(h_{\boldsymbol{\beta}}(\mathbf{X})) &= (\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))^T(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X})) + \lambda_1\|\boldsymbol{\beta}\|_2^2 + \lambda_2(\mathbf{A}\boldsymbol{\beta})^T(\mathbf{A}\boldsymbol{\beta})\\
\boldsymbol{\beta} &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, g(h_{\boldsymbol{\beta}}(\mathbf{X}))
\end{aligned}
$$

1.2.2 Solve optimization

$$
\begin{aligned}
\nabla_{\boldsymbol{\beta}}g(h_{\boldsymbol{\beta}}(\mathbf{X})) &= \nabla_{\boldsymbol{\beta}}tr(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X}))^T(\mathbf{Y}-h_{\boldsymbol{\beta}}(\mathbf{X})) + \nabla_{\boldsymbol{\beta}}\lambda_1 tr\boldsymbol{\beta}^T\boldsymbol{\beta} + \nabla_{\boldsymbol{\beta}}\lambda_2 tr(\mathbf{A}\boldsymbol{\beta})^T(\mathbf{A}\boldsymbol{\beta})\\
&= \nabla_{\boldsymbol{\beta}}tr\mathbf{Y}^T\mathbf{Y} - 2\nabla_{\boldsymbol{\beta}}tr\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta} + \nabla_{\boldsymbol{\beta}}\lambda_1 tr\boldsymbol{\beta}^T\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} + \lambda_1 2\boldsymbol{\beta} + \nabla_{\boldsymbol{\beta}}\lambda_2 tr\boldsymbol{\beta}\mathbf{I}\boldsymbol{\beta}^T\mathbf{A}^T\mathbf{A}\\
&= -2\mathbf{X}^T\mathbf{Y} + 2\lambda_1\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} + 2\lambda_1\boldsymbol{\beta} + 2\lambda_2\mathbf{A}^T\mathbf{A}\boldsymbol{\beta}\\
&:= 0\\
\boldsymbol{\beta} &= (\mathbf{X}^T\mathbf{X} + \lambda_1 + \lambda_2\mathbf{A}^T\mathbf{A})^{-1}\mathbf{X}^T\mathbf{Y}
\end{aligned}
$$

1.3 Linearly Constrained Linear Regression

Define,

$$L(\boldsymbol{\beta}, \lambda) = g(h_{\boldsymbol{\beta}}(\mathbf{X})) + \lambda(\mathbf{A}\boldsymbol{\beta} - \mathbf{b})$$

Then,

$$\begin{aligned}\boldsymbol{\beta} &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, L(\boldsymbol{\beta}, \lambda) \\ &= \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, g(h_{\boldsymbol{\beta}}(\mathbf{X})) + \lambda(\mathbf{A}\boldsymbol{\beta} - \mathbf{b})\end{aligned}$$

Differentiate with respective to $\boldsymbol{\beta}$, we get,

$$\begin{aligned}\nabla_{\boldsymbol{\beta}} L(\boldsymbol{\beta}, \lambda) &= \nabla_{\boldsymbol{\beta}}(g(h_{\boldsymbol{\beta}}(\mathbf{X})) + \lambda(\mathbf{A}\boldsymbol{\beta} - \mathbf{b})) \\ &= -2\mathbf{X}^T\mathbf{Y} + 2\mathbf{X}^T\mathbf{X}\boldsymbol{\beta} + \mathbf{A}^T\boldsymbol{\lambda} \\ &:= 0\end{aligned} \tag{2}$$

Differentiate with respective to $\boldsymbol{\lambda}$, we get,

$$\begin{aligned}\nabla_{\boldsymbol{\lambda}} L(\boldsymbol{\beta}, \lambda) &= \mathbf{A}\boldsymbol{\beta} - \mathbf{b} \\ &:= 0\end{aligned} \tag{3}$$

Put (1) and (2) together, we get,

$$\mathbf{MN} = \mathbf{K} \tag{4}$$

Where,

$$\begin{aligned}\mathbf{M} &= \begin{pmatrix} 2\mathbf{X}^T\mathbf{X} & \mathbf{A}^T \\ \mathbf{A} & 0 \end{pmatrix} \\ \mathbf{N} &= \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\lambda} \end{pmatrix} \\ \mathbf{K} &= \begin{pmatrix} 2\mathbf{X}^T\mathbf{Y} \\ \mathbf{b} \end{pmatrix}\end{aligned} \tag{5}$$

If $\mathbf{M}$ is invertible, then,

$$\begin{aligned}\mathbf{N} &= \mathbf{M}^{-1}\mathbf{K} \\ \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\lambda} \end{pmatrix} &= \begin{pmatrix} 2\mathbf{X}^T\mathbf{X} & \mathbf{A}^T \\ \mathbf{A} & 0 \end{pmatrix}^{-1} \begin{pmatrix} 2\mathbf{X}^T\mathbf{Y} \\ \mathbf{b} \end{pmatrix} \\ &= \begin{vmatrix} 2\mathbf{X}^T\mathbf{X} & \mathbf{A}^T \\ \mathbf{A} & 0 \end{vmatrix}^{-1} \begin{pmatrix} 0 & -\mathbf{A}^T \\ -\mathbf{A} & 2\mathbf{X}^T\mathbf{X} \end{pmatrix} \begin{pmatrix} 2\mathbf{X}^T\mathbf{Y} \\ \mathbf{b} \end{pmatrix} \\ &= (-\mathbf{A}^T\mathbf{A})^{-1} \begin{pmatrix} 0 & -\mathbf{A}^T \\ -\mathbf{A} & 2\mathbf{X}^T\mathbf{X} \end{pmatrix} \begin{pmatrix} 2\mathbf{X}^T\mathbf{Y} \\ \mathbf{b} \end{pmatrix}\end{aligned}$$

If $\mathbf{X}^T\mathbf{X}$ is invertible, then from (2) and (3) we have,

$$\begin{aligned}\boldsymbol{\beta} &= (2\mathbf{X}^T\mathbf{X})^{-1}(2\mathbf{X}^T\mathbf{Y} - \mathbf{A}^T\lambda) \\ \boldsymbol{\beta} &= \mathbf{A}^{-1}\mathbf{b}\end{aligned}$$

Thus, we get,

$$\begin{aligned}
\lambda &= 2(\mathbf{A}^T)^{-1}(\mathbf{X}^T\mathbf{X})\mathbf{A}^{-1}(\mathbf{A}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} - \mathbf{b}) \\
&= 2(\mathbf{A}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}^T)^{-1}(\mathbf{A}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} - \mathbf{b}) \\
\boldsymbol{\beta} &= (2\mathbf{X}^T\mathbf{X})^{-1}(2\mathbf{X}^T\mathbf{Y} - 2\mathbf{A}^T(\mathbf{A}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}^T)^{-1}(\mathbf{A}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} - \mathbf{b})) \\
&= (\mathbf{X}^T\mathbf{X})^{-1}(\mathbf{X}^T\mathbf{Y} - \mathbf{A}^T(\mathbf{A}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{A}^T)^{-1}(\mathbf{A}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} - \mathbf{b}))
\end{aligned}$$

2 Online Learning

Define Lagrange multipler as follows,

$$\begin{aligned}
L(\mathbf{w}_{i+1}, \lambda) &= \|\mathbf{w}_{i+1} - \mathbf{w}_i\|_2^2 + \lambda(\mathbf{w}_{i+1}^T\mathbf{x}_i)y_i \\
&= (\mathbf{w}_{i+1} - \mathbf{w}_i)^T(\mathbf{w}_{i+1} - \mathbf{w}_i) + \lambda(\mathbf{w}_{i+1}^T\mathbf{x}_i)y_i \\
\nabla_{\mathbf{w_{i+1}}} L(\mathbf{w}_{i+1}, \lambda) &= \nabla_{\mathbf{w_{i+1}}}(\mathbf{w}_{i+1} - \mathbf{w}_i)^T(\mathbf{w}_{i+1} - \mathbf{w}_i) + \lambda(\mathbf{w}_{i+1}^T\mathbf{x}_i)y_i \\
&= \nabla_{\mathbf{w_{i+1}}} tr((\mathbf{w}_{i+1} - \mathbf{w}_i)^T(\mathbf{w}_{i+1} - \mathbf{w}_i) + \lambda(\mathbf{w}_{i+1}^T\mathbf{x}_i)y_i) \\
&= \nabla_{\mathbf{w_{i+1}}} tr\mathbf{w}_{i+1}^T\mathbf{w}_{i+1} - \nabla_{\mathbf{w_{i+1}}} tr2\mathbf{w}_i^T\mathbf{w}_{i+1} + \nabla_{\mathbf{w_{i+1}}} tr\lambda y_i(\mathbf{w}_{i+1}^T\mathbf{x}_i) \\
&= 2\mathbf{w}_{i+1} - 2\mathbf{w}_i + \lambda y_i\mathbf{x_i} \\
&:= 0 \\
\nabla_\lambda L(\mathbf{w}_{i+1}, \lambda) &= (\mathbf{w}_{i+1}^T\mathbf{x}_i)y_i \\
&:= 0
\end{aligned}$$

Solve for these two equations, we get,

$$\mathbf{w}_{i+1} = \mathbf{w}_i - \frac{1}{2}\lambda\mathbf{x}_i$$

Where,

$$\mathbf{w}_{i+1}^T\mathbf{x}_i y_i = 0$$

3 Kernels

3.1 K_3

Since k_1 and k_2 are valid kernel functions, K_1 and K_2 are positive semi-definite. Then we have,

$$\begin{aligned}
\mathbf{V}^T\mathbf{K}_3\mathbf{V} &= a_1\mathbf{V}^T\mathbf{K}_1\mathbf{V} + a_2\mathbf{V}^T\mathbf{K}_2\mathbf{V}, \forall a_1, a_2 > 0, \forall\mathbf{V} \\
&\geq 0
\end{aligned}$$

So K_3 is positive semi-definite.

3.2 K_4

From the question, we get,

$$\begin{aligned}\mathbf{K}_4 &= \begin{pmatrix} f(\mathbf{x}_1)^2 & f(\mathbf{x}_1)f(\mathbf{x}_2) & \cdots & f(\mathbf{x}_1)f(\mathbf{x}_m) \\ f(\mathbf{x}_1)f(\mathbf{x}_2) & f(\mathbf{x}_2)^2 & \cdots & f(\mathbf{x}_2\mathbf{x}_m) \\ \vdots & \vdots & \ddots & \vdots \\ f(\mathbf{x}_1)f(\mathbf{x}_m) & f(\mathbf{x}_1)f(\mathbf{x}_{m-1}) & \cdots & f(\mathbf{x}_m)^2 \end{pmatrix} \\ &= F(\mathbf{X})F(\mathbf{X})^T\end{aligned}$$

Where,

$$F(\mathbf{X}) = \begin{pmatrix} f(\mathbf{x}_1) \\ f(\mathbf{x}_2) \\ \vdots \\ f(\mathbf{x}_m) \end{pmatrix}$$

Therefore,

$$\begin{aligned}\mathbf{V}^T\mathbf{K}_4\mathbf{V} &= \mathbf{V}^T F(\mathbf{X})F(\mathbf{X})^T\mathbf{V}, \forall \mathbf{V} \\ &= (F(\mathbf{X})^T\mathbf{V})^T(F(\mathbf{X})^T\mathbf{V}) \\ &\geq 0\end{aligned}$$

Therefore, K_4 is positive semi-definite.

3.3 K_5

Obviously, K_5 is the Hadamard product of K_1 and K_2. Since K_1 and K_2 are positive semi-definite matrices, according to the properties of Hadamard product, we have K_5 is a positive semi-definte matrix.

4 Variance Bias Tradeoff

4.1 Closed form solution

As defined above, we can rewrite the optimization formation as follows,

$$\begin{aligned}g(\boldsymbol{\beta}) &= \frac{1}{n}(\mathbf{Y} - h_{\boldsymbol{\beta}}(\mathbf{X}))^T(\mathbf{Y} - h_{\boldsymbol{\beta}}(\mathbf{X})) + \lambda\boldsymbol{\beta}^T\boldsymbol{\beta} \\ \nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}) &= \nabla_{\boldsymbol{\beta}}(\frac{1}{n}(tr\mathbf{Y}^T\mathbf{Y} + tr\boldsymbol{\beta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\beta} - 2tr\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta}) + tr\lambda\boldsymbol{\beta}^T\boldsymbol{\beta}) \\ &= \frac{2}{n}\boldsymbol{\beta}\mathbf{X}^T\mathbf{X} - 2\mathbf{X}^T\mathbf{Y} + 2\lambda\boldsymbol{\beta} \\ &:= 0 \\ \hat{\boldsymbol{\beta}} &= (\frac{1}{n}\mathbf{X}^T\mathbf{X} + \lambda)^{-1}\mathbf{X}^T\mathbf{Y}\end{aligned}$$

Since,

$$\epsilon \sim N(0, \sigma^2\mathbf{I})$$

So,

$$\mathbf{Y} \sim N(\mathbf{X}\boldsymbol{\beta}^*, \sigma^2\mathbf{I})$$

$$\hat{\boldsymbol{\beta}} \sim N((\frac{1}{n}\mathbf{X}^T\mathbf{X}+\lambda)^{-1}\mathbf{X}^T\mathbf{X}\boldsymbol{\beta}^*, \sigma^2(\frac{1}{n}\mathbf{X}^T\mathbf{X}+\lambda)^{-1}\mathbf{X}^T((\frac{1}{n}\mathbf{X}^T\mathbf{X}+\lambda)^{-1}\mathbf{X}^T)^T)$$

4.2 Bias

$$\begin{aligned} E[\mathbf{X}^{(i)T}\hat{\boldsymbol{\beta}}] - \mathbf{X}^{(i)T}\boldsymbol{\beta}^* &= \mathbf{X}^{(i)T}E(\hat{\boldsymbol{\beta}}) - \mathbf{X}^{(i)T}\boldsymbol{\beta}^* \\ &= \mathbf{X}^{(i)T}(\frac{1}{n}\mathbf{X}^T\mathbf{X}+\lambda)^{-1}\mathbf{X}^T\mathbf{X}\boldsymbol{\beta}^* - \mathbf{X}^{(i)T}\boldsymbol{\beta}^* \end{aligned}$$

4.3 Variance

$$\begin{aligned} Var[\mathbf{X}^{(i)T}\hat{\boldsymbol{\beta}}] &= \mathbf{X}^{(i)T}\Sigma_{\hat{\boldsymbol{\beta}}}\mathbf{X}^{(i)} \\ &= \sigma^2\mathbf{X}^{(i)T}(\frac{1}{n}\mathbf{X}^T\mathbf{X}+\lambda)^{-1}\mathbf{X}^T((\frac{1}{n}\mathbf{X}^T\mathbf{X}+\lambda)^{-1}\mathbf{X}^T)^T\mathbf{X}^{(i)} \end{aligned}$$

4.4 Bias-variance

$MSE = Bias^2 + Variance$. Obviously, from equations from 4.2 an 4.3, we can see that when λ goes up, variance goes down and bias goes up. When λ goes down, varinace goes up and bias goes down.

5 Programming

5.1 Top three words

Word	Count
enron	600
will	351
please	291

5.2 Batch gradient descent

5.2.1 Formula with no regularization

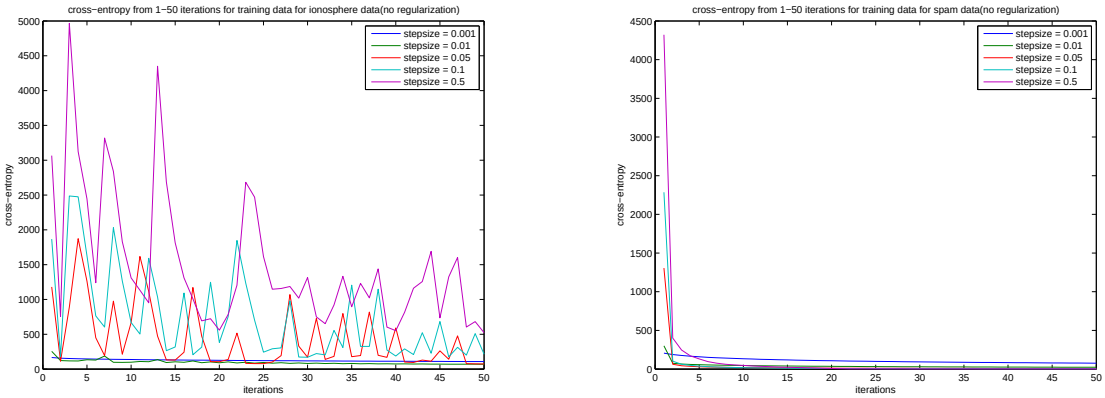
$$\begin{aligned} \begin{pmatrix} \mathbf{b}^{i+1} \\ \mathbf{w}^{i+1} \end{pmatrix} &:= \begin{pmatrix} \mathbf{b}^{i} \\ \mathbf{w}^{i} \end{pmatrix} - \eta\mathbf{X}^T(\mathbf{P}-\mathbf{Y}) \\ \mathbf{P} &:= \begin{pmatrix} \frac{1}{1+exp(\mathbf{X}^{(1)})^T\mathbf{w}^i)} \\ \frac{1}{1+exp(\mathbf{X}^{(2)})^T\mathbf{w}^i)} \\ \vdots \\ \frac{1}{1+exp(\mathbf{X}^{(m)})^T\mathbf{w}^i)} \end{pmatrix} \end{aligned} \tag{5}$$

5.2.2 Formula with regularization

$$\begin{pmatrix} \mathbf{b}^{i+1} \\ \mathbf{w}^{i+1} \end{pmatrix} := \begin{pmatrix} \mathbf{b}^{i} \\ \mathbf{w}^{i} \end{pmatrix} - \eta(\mathbf{X}^T(\mathbf{P}-\mathbf{Y}) + \begin{pmatrix} 0 \\ 2\lambda\mathbf{w}^i \end{pmatrix})$$
$$\mathbf{P} := \begin{pmatrix} \frac{1}{1+exp(-\mathbf{X}^{(1)})^T\mathbf{w}^i)} \\ \frac{1}{1+exp(-\mathbf{X}^{(2)})^T\mathbf{w}^i)} \\ \vdots \\ \frac{1}{1+exp(-\mathbf{X}^{(m)})^T\mathbf{w}^i)} \end{pmatrix} \tag{6}$$

5.2.3 Cross Entropy without regularization plot

Figure 1



(a) Ionosphere data

(b) Spam data

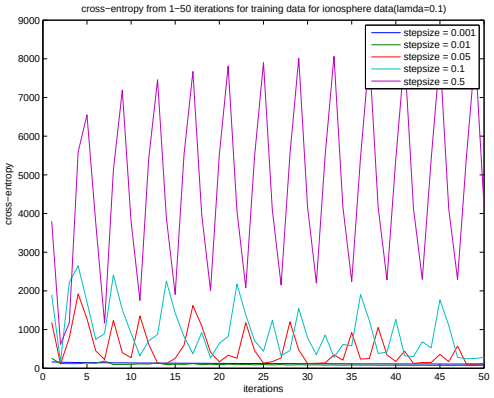
Figure 1: Cross entropy at 1-50 iterations at different stepsizes using gradient descent algorithm (without regularization)

5.2.4 L_2 norm without regularization

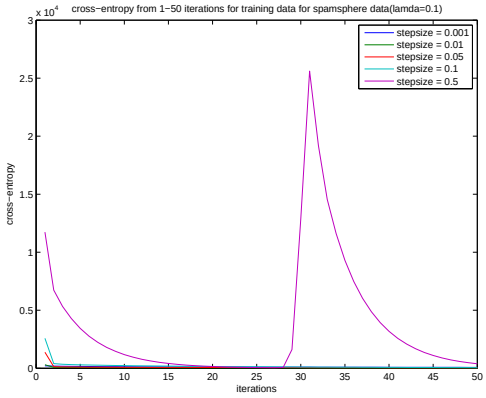
norm	0.001	0.01	0.05	0.1	0.5
Ionosphere	1.4946	4.6553	18.6196	37.2186	190.2706
Spam	2.5933	7.9696	28.4815	55.6097	275.7817

5.2.5 Cross Entropy with regularization plot

Figure 2



(a) Ionosphere data



(b) Spam data

Figure 2: Cross entropy at 1-50 iterations at different stepsizes using gradient descent algorithm (with regularization)

5.2.6 L_2 norm with regularization ($\eta = 0.01$)

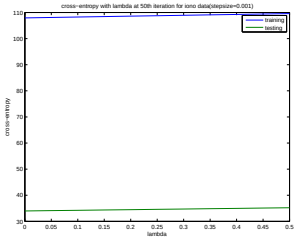
norm	0	0.05	0.10	0.15	0.20	0.25	0.30
Ionosphere	4.6553	4.5754	4.4988	4.4253	4.3548	4.2869	4.2216
Spam	7.9696	7.7183	7.4806	7.2558	7.0432	6.8422	6.6522

norm	0.35	0.40	0.45	0.50
Ionosphere	4.1586	4.0977	4.0387	3.9816
Spam	6.4726	6.3029	6.1426	5.9911

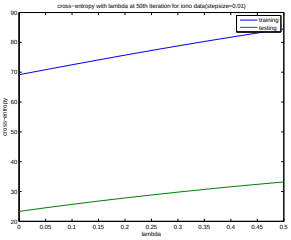
5.3 Cross Entropy at different regularization coefficient

Figure 3

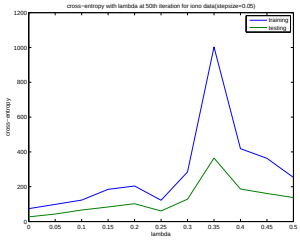
Figure 4



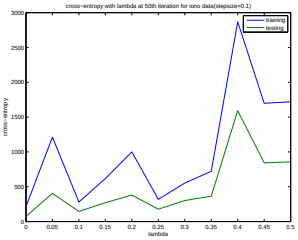
(a) Ionosphere data (stepsize = 0.001)



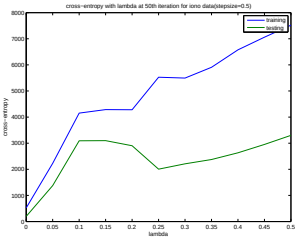
(b) Ionosphere data (stepsize = 0.01)



(c) Ionosphere data (stepsize = 0.05)



(d) Ionosphere data (stepsize = 0.1)



(e) Ionosphere data (stepsize = 0.5)

Figure 3: Cross entropy at 50th iteration at different stepsizes using newton algorithm (without regularization)

5.4 Newton's method

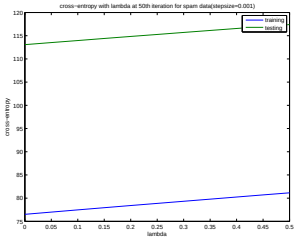
5.4.1 Formula with no regularization

$$\begin{pmatrix} \mathbf{b}^{i+1} \\ \mathbf{w}^{i+1} \end{pmatrix} := \begin{pmatrix} \mathbf{b}^{i} \\ \mathbf{w}^{i} \end{pmatrix} - (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{P}-\mathbf{Y})$$

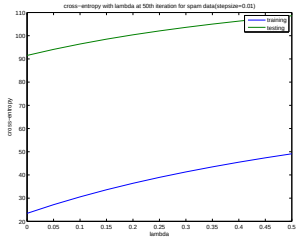
$$\mathbf{P} := \begin{pmatrix} \frac{1}{1+exp(\mathbf{X}^{(1)})^T\mathbf{w}^i)} \\ \frac{1}{1+exp(\mathbf{X}^{(2)})^T\mathbf{w}^i)} \\ \vdots \\ \frac{1}{1+exp(\mathbf{X}^{(m)})^T\mathbf{w}^i)} \end{pmatrix} \tag{7}$$

$$\mathbf{W} := diag(\mathbf{P}.*(\mathbf{I}-\mathbf{P}))$$

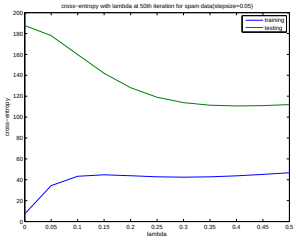
where $diag$ means create a diagonal of matrix based on the value of the vector. $.*$ means elementwise multiplication



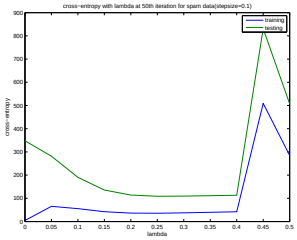
(a) Spam data (stepsize = 0.001)



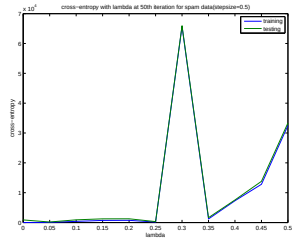
(b) Spam data (stepsize = 0.01)



(c) Spam data (stepsize = 0.05)



(d) Spam data (stepsize = 0.1)



(e) Spam data (stepsize = 0.5)

Figure 4: Cross entropy at 50th iteration at different stepsizes using newton algorithm (without regularization)

5.4.2 Formula with regularization

$$\begin{pmatrix} \mathbf{b}^{i+1} \\ \mathbf{w}^{i+1} \end{pmatrix} := \begin{pmatrix} \mathbf{b}^{i} \\ \mathbf{w}^{i} \end{pmatrix} - (\mathbf{X}^T\mathbf{W}\mathbf{X} + 2 * \lambda\mathbf{I})^{-1}\mathbf{X}^T(\mathbf{P} - \mathbf{Y})$$

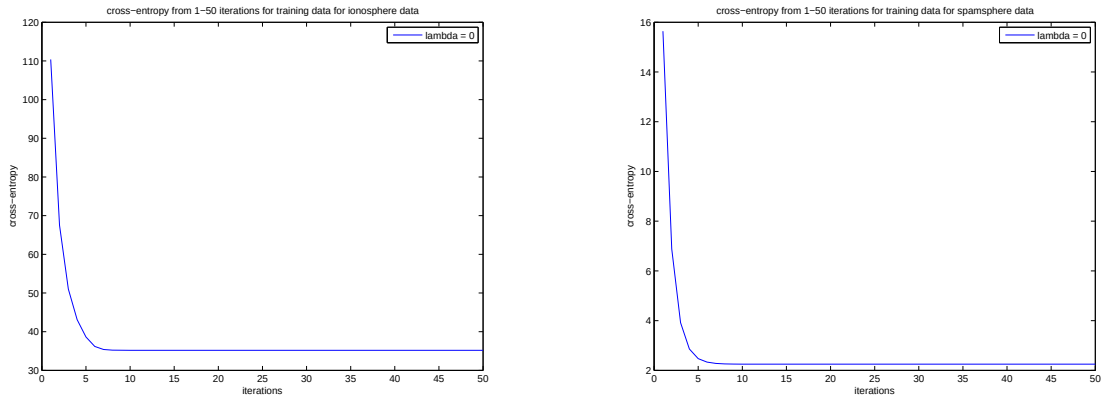
$$\mathbf{P} := \begin{pmatrix} \frac{1}{1+exp(\mathbf{X}^{(1)})^T\mathbf{w}^i)} \\ \frac{1}{1+exp(\mathbf{X}^{(2)})^T\mathbf{w}^i)} \\ \vdots \\ \frac{1}{1+exp(\mathbf{X}^{(m)})^T\mathbf{w}^i)} \end{pmatrix} \tag{8}$$

$$\mathbf{W} := diag(\mathbf{P}. * (\mathbf{I} - \mathbf{P}))$$

where $diag$ means create a diagonal of matrix based on the value of the vector. $.*$ means elementwise multiplication

5.4.3 Entropy function without regularization plot

Figure 5



(a) Ionosphere data

(b) Spam data

Figure 5: Cross entropy at 1-50 iterations using newton algorithm (without regularization)

5.4.4 L_2 norm without regularization

Ionosphere data = 67.615

Ionosphere data = 297.92

5.4.5 Cross entropy without regularization for test data

Ionosphere data = 79.8198

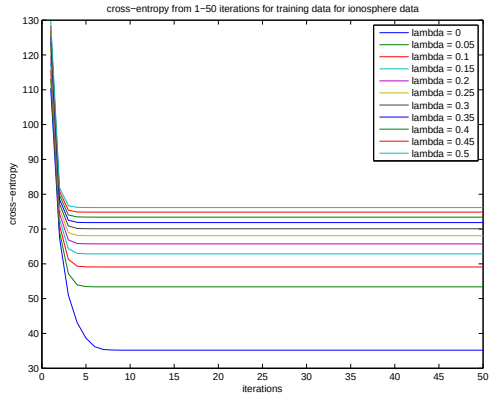
Ionosphere data = 1727

5.4.6 Cross entropy with regularization plot

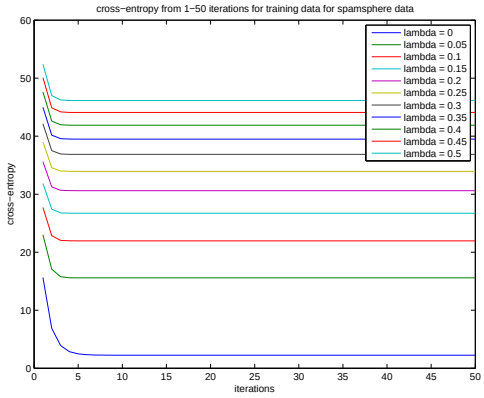
Figure 6

5.4.7 L_2 norm with regularization

norm	0	0.05	0.10	0.15	0.20	0.25	0.30
Ionosphere	67.6150	12.3324	9.4225	8.0300	7.1672	6.5655	6.1146
Spam	297.9283	12.4677	10.3636	9.2211	8.4543	7.8860	7.4395



(a) Ionosphere data



(b) Spam data

Figure 6: Cross entropy at 1-50 iterations using newton algorithm (regularization)

norm	0.35	0.40	0.45	0.50
Ionosphere	5.75981	5.4704	5.2282	5.0211
Spam	7.0748	6.7685	6.5059	6.2770

5.4.8 Cross entropy with regularization

norm	0	0.05	0.10	0.15	0.20	0.25	0.30
Ionosphere	79.8198	40.0769	36.3005	35.1174	34.8151	34.89689	35.1557
Spam	1727	1224	1161	1139	1131	1128	1129

norm	0.35	0.40	0.45	0.50
Ionosphere	35.4951	35.8681	36.2510	36.6319
Spam	1132	1135	1140	1145

5.5 Conclusion for 8

At the same level of regularization, stability of convergence is negatively related to stepsize. At the same level of stepsize, stability of convergence is negatively related to level of regularization.

5.6 Conclusion for 9

Generally, Newton method is less robust than gradient decent. However, with initial points properly chosen, it, steadily and rapidly, converges.