CSC311 Assignment1

Peiqing Yu

September 30th 2020

1. **Classification with nearest neighbour**
   (a) Please check hw1_q1_code.py
   (b) Please check Figure1 in appendix
   (c) When we consider vector space model for text, the cosine method is better at catching semantic similarity in the text. That's because the cosine metrics measure the angle between two document vector instead of the euclidean metrics which measure the distance of every dimension which is non-zero in either vector.For text difference, it is the orientation of vector matters the most. Also, euclidean metrics is not very good at dealing with high dimensional data.

2. **Regularized Linear Regression**

   (a) The gradient descent update rule for the regularized cost function $\mathcal{J}^{\beta}_{reg}$ is:

$$
\begin{aligned}
w_j &\leftarrow w_j - \alpha(\frac{\partial \mathcal{J}^{\beta}_{reg}}{\partial w_j}) \\
&= w_j - \alpha(\frac{\partial \mathcal{J}}{\partial w_j} + \frac{\partial \mathcal{R}}{\partial w_j}) \\
&= w_j - \alpha(\frac{\partial \mathcal{J}}{\partial w_j} + \beta_j w_j) \\
&= w_j - \alpha(\frac{1}{N}\sum_{i=1}^{N} x_j(y^{(i)} - t^{(i)}) + \beta_j w_j) \\
&= w_j(1-\alpha\beta_j) - \alpha\frac{\partial \mathcal{J}}{\partial w_j} \qquad * \\
b &\leftarrow b - \alpha(\frac{\partial \mathcal{J}^{\beta}_{reg}}{\partial b}) \\
&= b - \alpha(\frac{\partial \mathcal{J}}{\partial b} + \frac{\partial \mathcal{R}}{\partial b}) \\
&= b - \alpha(\frac{1}{N}\sum_{i=1}^{N}(y^{(i)} - t^{(i)}) + 0) \\
&= b - \alpha\frac{\partial \mathcal{J}}{\partial b}
\end{aligned}
$$

   We tend to update the W_j, b in the direction opposite the steepest ascent, the gradient.This form of regularization is sometimes called weight decay because we can see from the update rule that we keep the weight from being too big(*) by multiplying a factor slightly less than 1. Note that the bias b not affect the $\mathcal{J}^{\beta}_{reg}$

(b) We can derive a system of linear equations for $\mathcal{J}_{reg}^{\beta}$

$$\begin{aligned}
\frac{\partial \mathcal{J}_{reg}^{\beta}}{\partial w_j} &= \frac{\partial \mathcal{J}}{\partial w_j} + \frac{\partial \mathcal{R}}{\partial w_j} \\
&= \frac{1}{N}\sum_{i=1}^{N} x_j^{(i)}(\sum_{j'=1}^{D} w_{j'}x_{j'}^{(i)} - t^{(i)}) + \beta_j w_j \\
&= \sum_{j'=1}^{D} \frac{1}{N}(\sum_{i=1}^{N} x_j^{(i)}x_{j'}^{(i)})w_{j'} - \frac{1}{N}\sum_{i=1}^{N} x_j^{(i)}t^{(i)} + \beta_j w_j \\
\mathbf{A}_{jj'} &= \frac{1}{N}(\sum_{i=1}^{N} x_j^{(i)}x_{j'}^{(i)})w_{j'} \\
\mathbf{c}_j &= \frac{1}{N}\sum_{i=1}^{N} x_j^{(i)}t^{(i)} - \beta_j w_j
\end{aligned}$$

(c) According to part(b), we can have that:

$$\begin{aligned}
\mathbf{A} &= \frac{1}{N}\mathbf{X}^T\mathbf{X} \\
\mathbf{c} &= \frac{1}{N}\mathbf{X}^T\mathbf{t} - diag(\beta)\mathbf{W}
\end{aligned}$$

Since we have $\nabla\mathcal{J}_{reg} =$ **Aw-c**, we should let the gradient equals to 0 to find the closed form solution. Assuming **A** is invertible:

$$\begin{aligned}
\mathbf{w} &= N(\mathbf{X}^T\mathbf{X})^{-1}(\frac{1}{N}\mathbf{X}^T\mathbf{t} - diag(\beta)\mathbf{w}) \\
\mathbf{w} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t} - N(\mathbf{X}^T\mathbf{X})^{-1}diag(\beta)\mathbf{w} \\
(\mathbf{I} + N(\mathbf{X}^T\mathbf{X})^{-1}diag(\beta))\mathbf{w} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t} \\
\mathbf{w} &= (\mathbf{I} + N(\mathbf{X}^T\mathbf{X})^{-1}diag(\beta))^{-1}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t} \\
\mathbf{w} &= (\mathbf{I} + (N(\mathbf{X}^T\mathbf{X})^{-1}diag(\beta))^{-1})(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t} \\
\mathbf{w} &= (\mathbf{I} + \frac{1}{N}diag(\beta)^{-1}(\mathbf{X}^T\mathbf{X}))(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t} \\
\mathbf{w} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{t} + \frac{1}{N}diag(\beta)^{-1}\mathbf{X}^T\mathbf{t}
\end{aligned}$$

3. **Loss Function**

We can derive a sequence of vectorized mathematical expression for the gradients of the cost with respect to w and b:

$$\begin{aligned}
\mathbf{y} &= \mathbf{XW} + \mathbf{b} \\
\frac{\partial \mathcal{J}}{\partial \mathbf{y}} &= \frac{1}{N}sum(sin(\mathbf{XW} + \mathbf{b} - \mathbf{t})) \\
\frac{\partial \mathcal{J}}{\partial \mathbf{w}} &= \frac{1}{N}\mathbf{X}^T sin(\mathbf{XW} + \mathbf{b} - \mathbf{t}) \\
\frac{\partial \mathcal{J}}{\partial \mathbf{b}} &= \frac{1}{N}sum(sin(\mathbf{XW} + \mathbf{b} - \mathbf{t}))
\end{aligned}$$

4. **Cross Validation**

(b) Please check hw1_q4_code.py

(c) Please check hw1_q4_code.py

(d) The shape of the test error and 5-fold, 10-fold cross validation error look like a hook. This is because we start with relatively no regularization which results in a large weight thus large error. When we increase the lambda, we penalize the weight more thus reach a point of lowest error. However, when we pass the optimal lambda, we suffer a trade off that the model is too simple.

Please also check appendix Figure2 Figure3.

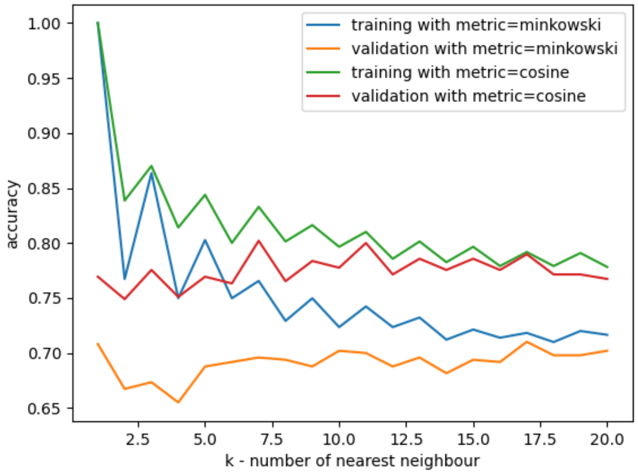


Figure1

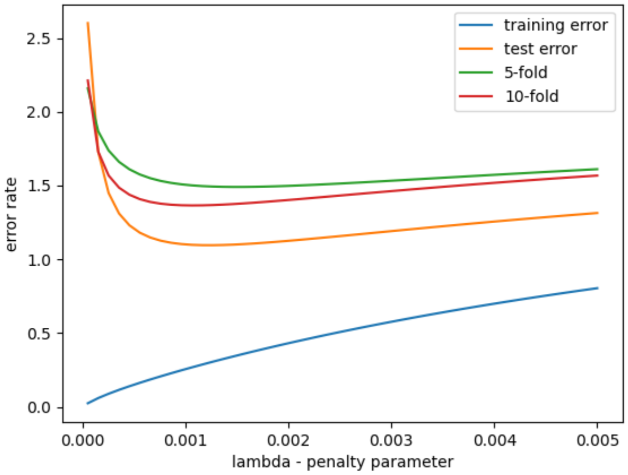
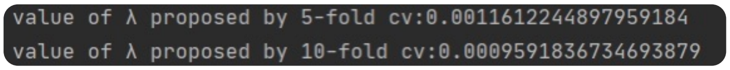


Figure 2



```
value of λ proposed by 5-fold cv:0.0011612244897959184
value of λ proposed by 10-fold cv:0.0009591836734693879
```

Figure 3 Report of lambda choose