ML for DS. HW2

Abhay Pawar
asp2197

Problem 1

$$\hat{\pi}, \hat{\theta}_y^{(1)}, \hat{\theta}_y^{(2)} = \arg\max_{\pi, \theta_y^{(1)}, \theta_y^{(2)}} \sum_{i=1}^{n} \ln p(y_i|\pi) + \sum_{i=1}^{n} \ln p(x_{i1}|\theta_{y_i}^{(1)}) + \sum_{i=1}^{n} \ln p(x_{i2}|\theta_{y_i}^{(2)})$$

$\hat{\pi} \rightarrow$

Ignoring last two terms

$$\hat{\pi} = \arg\max_{\pi} \sum_{i=1}^{n} \ln p(y_i|\pi)$$

Let $\sum \ln p(y_i|\pi)$ be P_1

$$\frac{\partial P_1}{\partial \pi} = \frac{\partial}{\partial \pi}\left[\sum \ln [\pi^{y_i}(1-\pi)^{1-y_i}]\right]$$

$$= \frac{\partial}{\partial \pi}\left[\sum y_i \ln \pi + \sum (1-y_i)\ln(1-\pi)\right]$$

$$= \frac{\sum y_i}{\pi} + \frac{\sum (1-y_i)}{(1-\pi)}(-1) = 0$$

$$\therefore \sum y_i - \pi(\sum y_i) = n\pi - \pi \sum y_i$$

$$\therefore \boxed{\hat{\pi} = \frac{\sum_{i=1}^{n} y_i}{n}}$$

$\theta_y^{(1)} \rightarrow$ Will only depend on second term.

$$\hat{\theta}_y^{(1)} = \arg\max_{\theta_y^{(1)}} \sum \ln p(x_i|\theta_{y_i}^{(1)}) = \arg\max_{\theta_y^{(1)}} (P_2)$$

Since, $\hat{\theta}_{y_i}^{(1)}$ can take two values based on y_i, P_2 can be written as -

$$P_2 = \sum \ln p(x_i|\theta_{y_i=0}^{(1)}) \times y_i + \sum \ln p(x_i|\theta_{y_i=1}^{(1)}) \times (1-y)$$

$$\therefore \frac{\partial P_2}{\partial \theta_{y_i}^{(1)}} = \left(\frac{\sum x_i y_i}{\theta_{y_i=0}^{(1)}} + \frac{\sum (1-x_i) y_i (-1)}{(1-\theta_{y_i=0}^{(1)})}\right) + \left(\frac{\sum x_i(1-y_i)}{\theta_{y_i=1}^{(1)}} + \frac{\sum (1-x_i)(1-y_i)(-1)}{(1-\theta_{y_i=1}^{(1)})}\right)$$

For $y_i = 0$, the first term will be zero & second term needs to be equated to zero for maximising.

Similarly for $y_i = 1$.

Using results from $\hat{\Pi}$,

$$\hat{\theta}^{(1)}_{y_i} = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} y_i} + \frac{\sum_{i=1}^{n} x_i (1-y_i)}{\sum_{i=1}^{n} (1-y_i)}$$

(c) $\hat{\theta}^{(2)}_{y} \rightarrow$

$$P_3 = \sum_{i=1}^{n} \ln p(x_{i2} | \theta^{(2)}_{y_i})$$

Splitting as done in (b) -

$$P_3 = \sum_{i=1}^{n} y_i \ln p(x_{i2} | \theta^{(2)}_{y_i=0}) + \sum_{i=1}^{n} (1-y_i) \ln p(x_{i2} | \theta^{(2)}_{y_i=1})$$

$$P_2 = \sum_{i=1}^{n} y_i [\ln \theta^{(2)}_{y=0} - (\theta^{(2)}_{y=0}+1)\ln(x_{i2})] + \sum (1-y_i) [\ln \theta^{(2)}_{y=1} - (\theta^{(2)}_{y=1}+1) \times \ln(x_{i2})]$$

$$\frac{\partial P_2}{\partial \theta^{(2)}_y} = \sum y_i \left[\frac{1}{\theta^{(2)}_{y=0}} - \theta^{(2)}_{y=0} \ln(x_{i2})\right] + \sum (1-y_i) \left[\frac{1}{\theta^{(2)}_{y=1}} - \theta^{(2)}_{y=1} \ln(x_{i2})\right]$$

For $y_i=0$, First term will be zero & second needs to be set to zero to find $\theta^{(2)}_{y=1}$ and vice versa.

$$\therefore \theta^{(2)}_y = \frac{\sum_{y=1}^{n} y_i}{\sum_{y=1}^{n} \ln(x_{i2}) y_i} + \frac{\sum_{y=1}^{n} (1-y_i)}{\sum_{i=1}^{n} \ln(x_{i2})(1-y_i)}$$

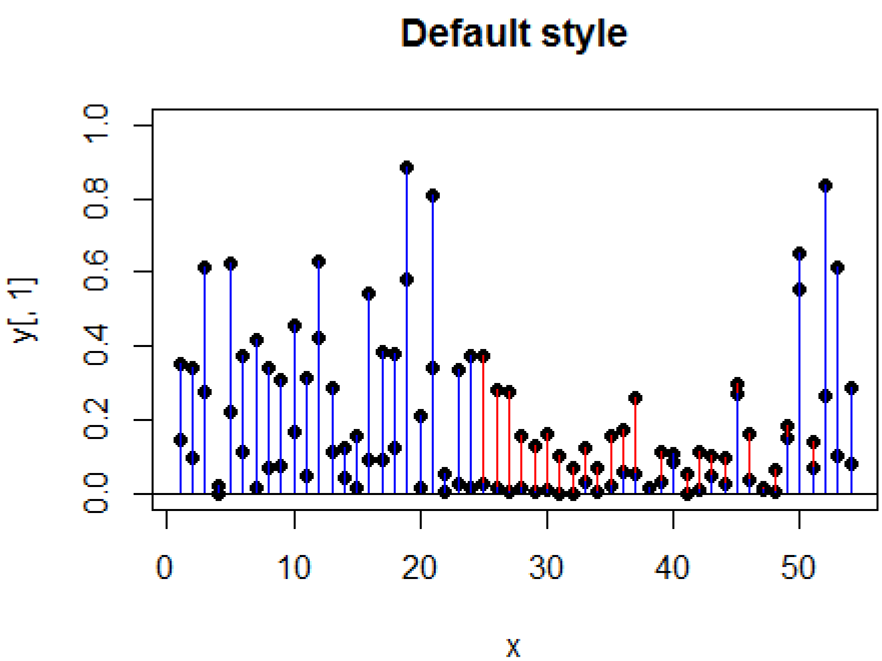
Problem 2

(a)

	Predicted y=0	Predicted y=1
Actual y=0	54	2
Actual y=1	5	32

Accuracy = 92.47%

(b)

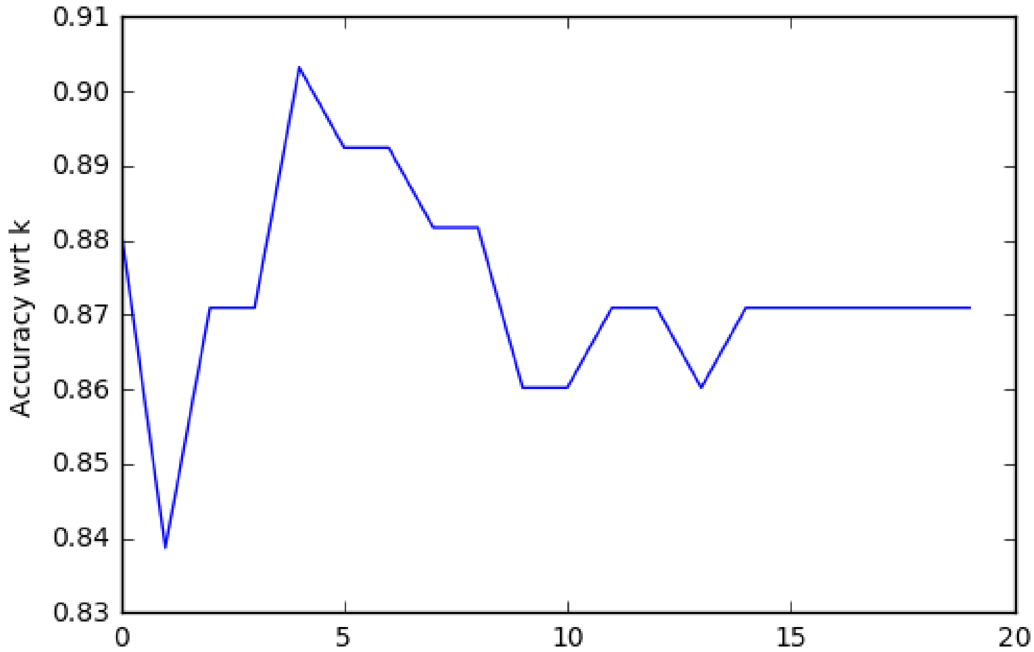


Variables 16 and 52 have the largest difference in values of the two parameters. This implies that these variables can differentiate between y=0 and y=1 well. Variable sixteen tells the presence of word 'free' in the mail. It is obvious why these variable has high predictive power. The word 'free' implies that the mail is trying to sell something to the customer and giving something free. Such mails can be clearly classified as spam. Variable 52 tells if exclamation is present in the mail. Again, spam emails might use more exclamations to convey excitement and to push the customer to buy something.

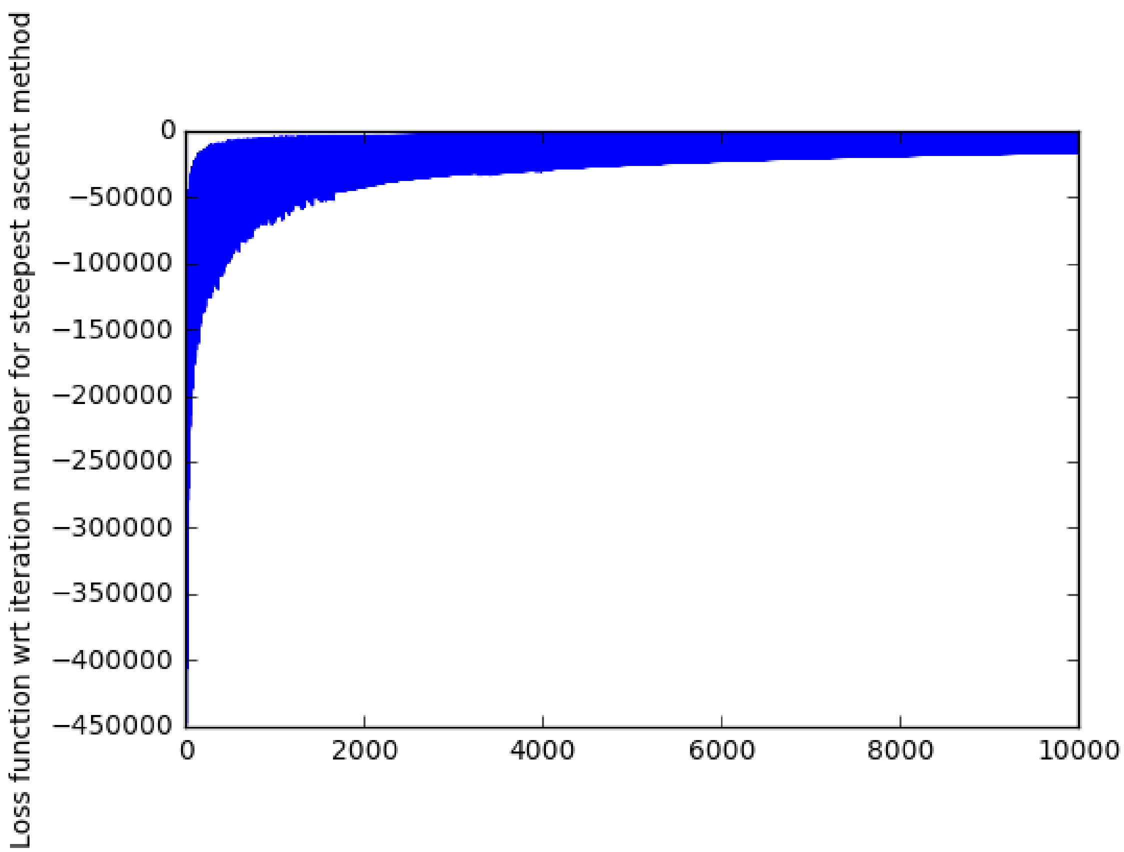
(c)

For even k, if equal number of neighbors have y=1 and y=0, I am assigning y_pred=1. Following are the prediction accuracies starting with k=1:

```
0.8817204 0.8387097 0.8709677 0.8709677 0.9032258 0.8924731 0.8924731 0.88172
0.8817204 0.8602150 0.8602150 0.8709677 0.8709677 0.8602150 0.8709677 0.87096
0.8709677 0.8709677 0.8709677 0.8709677
```


Accuracy wrt k
0.91
0.90
0.89
0.88
0.87
0.86
0.85
0.84
0.83
0
5
10
15
20

(d)


Loss function wrt iteration number for steepest ascent method
0
−50000
−100000
−150000
−200000
−250000
−300000
−350000
−400000
−450000
0
2000
4000
6000
8000
10000

(e)

Accuracy = 91.39%