CSC411 A3

Shiyuan Lin

December 2017

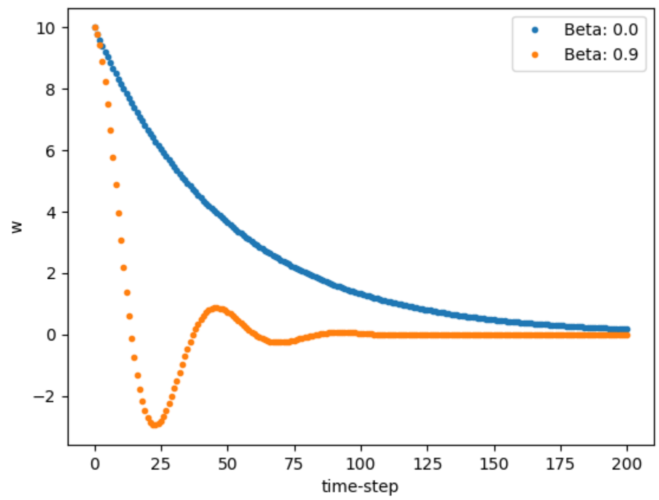
Question 1

```
BernoulliNB baseline train accuracy = 0.5987272405868835
BernoulliNB baseline test accuracy = 0.4579129049389272
Logistic Regression train accuracy = 0.9740144953155383
Logistic Regression test accuracy = 0.6901221455124801
SVM train accuracy = 0.954127629485593
SVM test accuracy = 0.6631704726500266
KNN train accuracy = 0.9737493371044723
KNN test accuracy = 0.11338289962825279
[ 0.51864407  0.65227818  0.62924282  0.63        0.73743017  0.83384615
  0.79283887  0.49517685  0.78516624  0.81746032  0.88265306  0.83532934
  0.61197917  0.7755102   0.72524752  0.64135021  0.60545906  0.8245614
  0.55426357  0.42857143]
[19  7  0 18 16 12  2  3 15  1 14  4 13  8  6  9 17  5 11 10]
2 most confused classes are "talk.religion.misc" and "rec.autos"
```

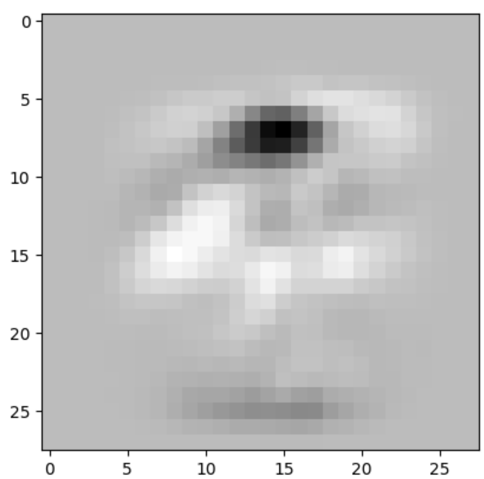
I used "RandomizedSearchCV" in picking the best hyperparameter. RandomizedSearchCV implements a randomized search over parameters, where each setting is sampled from a distribution over possible parameter values.
The reason I picked the logistic regression is that it is the most commonly used model in the world and fast to run. Also I do not need to worry about features being correlated. The general performance for SVM is quite well even without the kernel and SVM performance can be even better with the proper kernel. KNN had a good performance in assignment 2 in low dimensions. I was wondering its performance in high dimensions. The accuracy for logistic regression and SVM is not bad. Both models accuracy is close to 70%. As for KNN, though its accuracy is low, it did not surprise me since the data used in this question is high dimensional.

Question 2

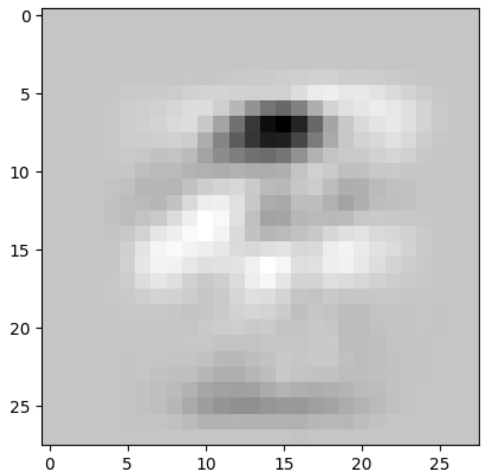
2.1


Beta: 0.0
Beta: 0.9
w
time-step
10
8
6
4
2
0
−2
0
25
50
75
100
125
150
175
200

2.3.5.1


0
5
10
15
20
25
0
5
10
15
20
25

2.3.5.2



Result

```
Data Loaded
Train size: 11025
Test size: 2757
------------------------------
The train loss is 0.3929884616568907 when beta is 0
The test loss is 0.3955727494173091 when beta is 0
The train accuracy is 0.9173696145124717 when beta is 0
The test accuracy is 0.9194776931447225 when beta is 0
The train loss is 0.3622781747932676 when beta is 0.1
The test loss is 0.3494721417690545 when beta is 0.1
The train accuracy is 0.893968253968254 when beta is 0.1
The test accuracy is 0.8951759158505622 when beta is 0.1
```

Question 3

3.1.1 Suppose k is symmetric, then

$$\begin{aligned} k &= V\Lambda V^T \\ &= Vdiag(\lambda_1, ...\lambda_d)V^T \\ &= (Vdiag(\sqrt{\lambda_1}, ...\sqrt{\lambda_d})) * (diag(\sqrt{\lambda_1}, ...\sqrt{\lambda_d})V^T) \\ &= k^{\frac{1}{2}}k^{\frac{1}{2}} \end{aligned}$$

Let $k^{\frac{1}{2}} = x^TU$, *since* $k^{\frac{1}{2}} \cdot k^{\frac{1}{2}T} \geq 0 \Rightarrow (x^TU)(U^Tx) = x^Tkx \geq 0$

Suppose $x^Tkx \geq 0$, *Let* x *be an arbitrary vector. Using the spectral decomposition,* $\Rightarrow x^Tkx = (V^TU)diag(\lambda)(U^TV) = \sum_{i=1}^{U} \lambda_i([V^TU]_i)^2 \geq 0$, *where* U *is a matrix containing the* d *orthogonal eigenvectors of* k. *The expression is non-negative for all* x *iff* $\lambda_i \geq 0 for i$ *= 1, ..., n.So* k *is positive semi-define.*

3.2.1
Let $\phi(x) = a^{0.5}$*, then* $k(x,y) =< \phi(x), \phi(y) >= a^{0.5} \cdot a^{0.5} = a$*, so k(x,y) = a is kernel for* $a \geq 0$

3.2.2
$\phi(x) = f(x) \Rightarrow k(x,y) =< \phi(x), \phi(y) >= f(x) \cdot f(y)$

3.2.3
Let $\phi(x) = [\sqrt{a}\phi^1(x)\sqrt{b}\phi^2(x)]$*, Then*

$$\begin{aligned} k(x,y) &=< \phi(x), \phi(y) > \\ &= a < \phi^{(1)}(x), \phi^{(1)}(y) > +b < \phi^{(2)}(x), \phi^{(2)}(y) > \\ &= ak_1(x,y) + bk_2(x,y) \end{aligned}$$