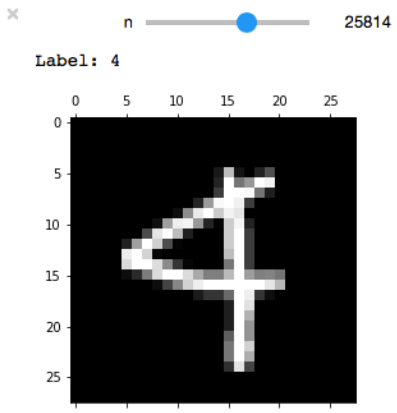
Homework 1 for AML

by Hanfu Liao@ORIE5750, Yuxiang Zhu@CS5785

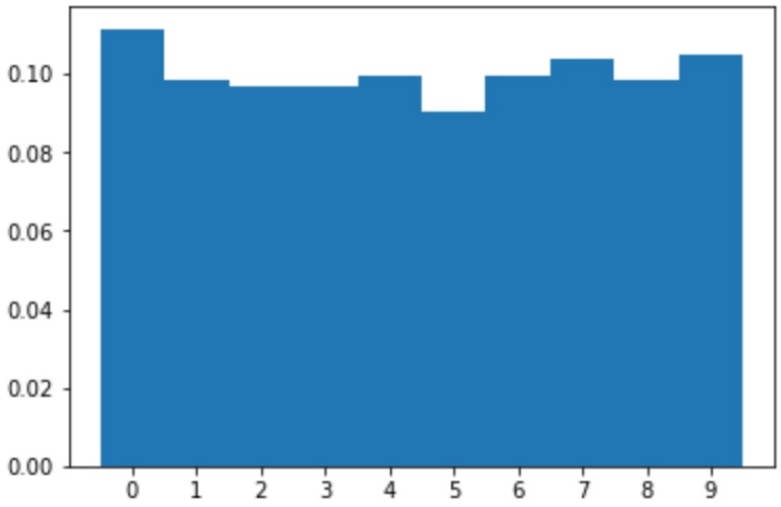
1.DigitRecognizer

1. Joined Kaggle with username Hanfu.
2. Use function `show_image` to show single MNIST image and library `interact` to show all.



3. Use function `show_prior_prob` to display a normalized histogram of digit counts.

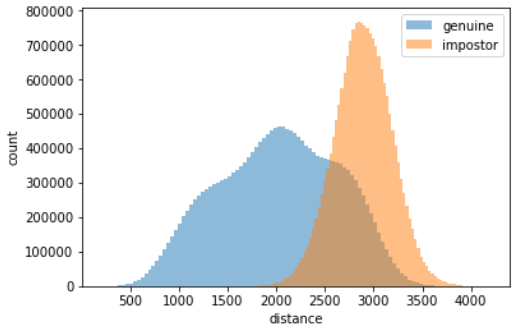
   As we can see from the graph, the prior probabilities of digits are not even.



4. Function `mostNearestNeighbor` takes 1 point, training set returns the index of the nearest target in training set. Function `findUniqDigitsIndex` returns a list of indices of each one of different digits. Function `findNearestNeighborQuestion` gives the input below:

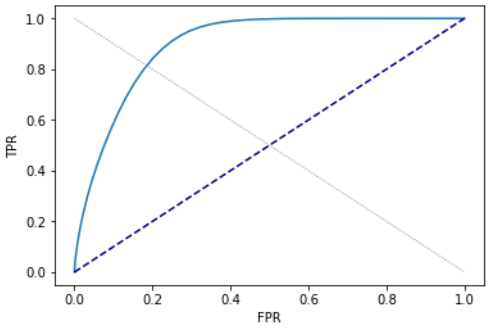
```
Predicted Value :0 Actual value :0
Predicted Value :1 Actual value :1
Predicted Value :2 Actual value :2
Predicted Value :5 Actual value :3
*
Predicted Value :4 Actual value :4
Predicted Value :5 Actual value :5
Predicted Value :6 Actual value :6
Predicted Value :7 Actual value :7
Predicted Value :8 Actual value :8
Predicted Value :9 Actual value :9
```

5. Use function `plotHistogram` to plot the histograms with 100 bin.



6. ROC curve:

   The equal error rate is 0.18



7. The main body of the KNN classifier is named `nearestNeighborKCdist` with some helper function, details in file named Question 1.ipynb.

8. With k= 3, the accuracy rate are 0.9665, 0.965, 0.967785714286 so then the average accuracy rate is 0.966428571429.

```
[[1358    0    7    0    0    3    6    0    5    2]
 [   0 1561   17    2   16    0    1   22   11    3]
 [   2    1 1364    6    0    1    0    2    5    0]
 [   0    2    6 1369    0   20    0    0   20    7]
 [   0    2    2    0 1308    0    3    7    1   13]
 [   2    1    1   12    0 1227    8    0   30    4]
 [   7    4    1    1    5   11 1381    0    8    2]
 [   0    1   22    7    3    2    0 1414    4   23]
 [   1    1    4    9    0    3    0    0 1240    4]
 [   1    2    3    4   36    9    0   19   19 1309]]
success rate:  0.9665

[[1339    0    8    1    2    3    8    0    2    4]
 [   0 1517   18    1   14    0    2   19   10    3]
 [   2    5 1329    9    0    0    0    3    5    1]
 [   0    0    5 1438    0   22    0    2   21   12]
 [   0    1    1    0 1325    1    1    1    8   16]
 [   3    1    0   18    0 1189    9    0   26    3]
 [   4    1    3    1    4   16 1358    0    6    0]
 [   0    6   29   11    3    1    0 1417    2   12]
 [   1    1    4   10    1    2    2    0 1256    5]
 [   1    0    0   10   39   11    0   18   15 1342]]
success rate:  0.965

[[1408    0   14    1    0    5    5    1    5    6]
 [   0 1573    6    6   12    2    2   12   25    3]
 [   0    2 1295   14    0    0    0    3    4    2]
 [   0    0    4 1388    0   20    0    0   16   10]
 [   0    1    0    0 1262    2    2    3    5   10]
 [   1    0    2   14    0 1213    5    0   22    4]
 [   2    0    3    0    7   19 1344    0    6    0]
 [   0    1   25    7    1    1    0 1441    4   26]
 [   0    0    2    7    0    4    0    0 1266    3]
 [   0    0    2    5   34    8    0   17   16 1359]]
success rate:  0.967785714286


average success rate:  0.966428571429
```

9. According to the confusion matrix above, we found following:

   2 is easily to be predicted as 7

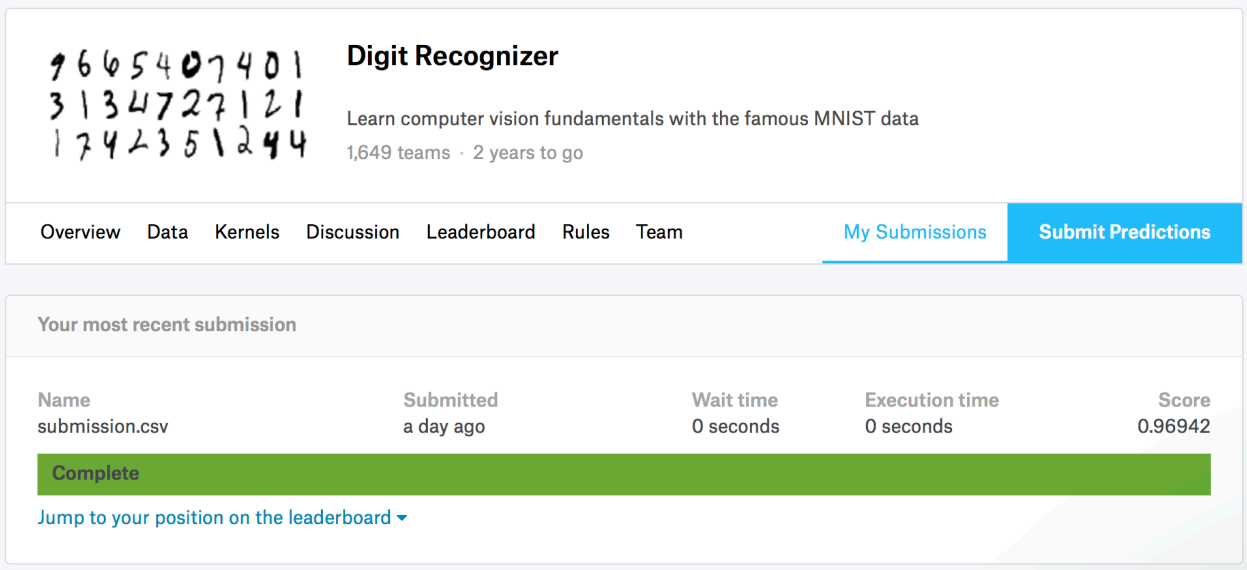
   4 is easily to be predicted as 9

   5 is easily to be predicted as 3

   8 is easily to be predicted as 5

   Particularly, 8 and 9 are very tricky to classify.

10. Kaggle result is here:



2. TheTitanicDisaster

    1. Join the the Kaggle as user named YuxiangZhu.
    2. In the logistic regression model:

        The following features are taken into consideration:

        pclass : the class of the cabin may determine if the lifeboats are closer or not

        sex : males are more likely to give the chance to woman as shown in the movie

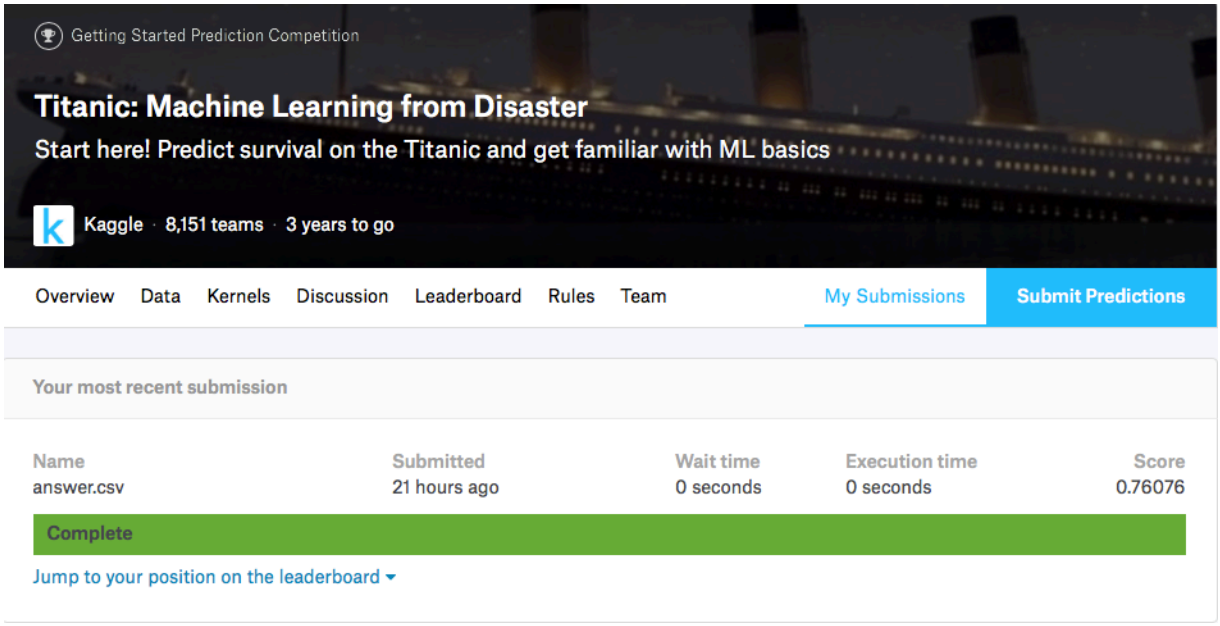
        age : the old will be more likely to give the chance to the young people as they are going to die soon anyway

        sib/spouse: families are running as a group, they don't abandom the others

        parents/children: Similarly as sib/spouse case

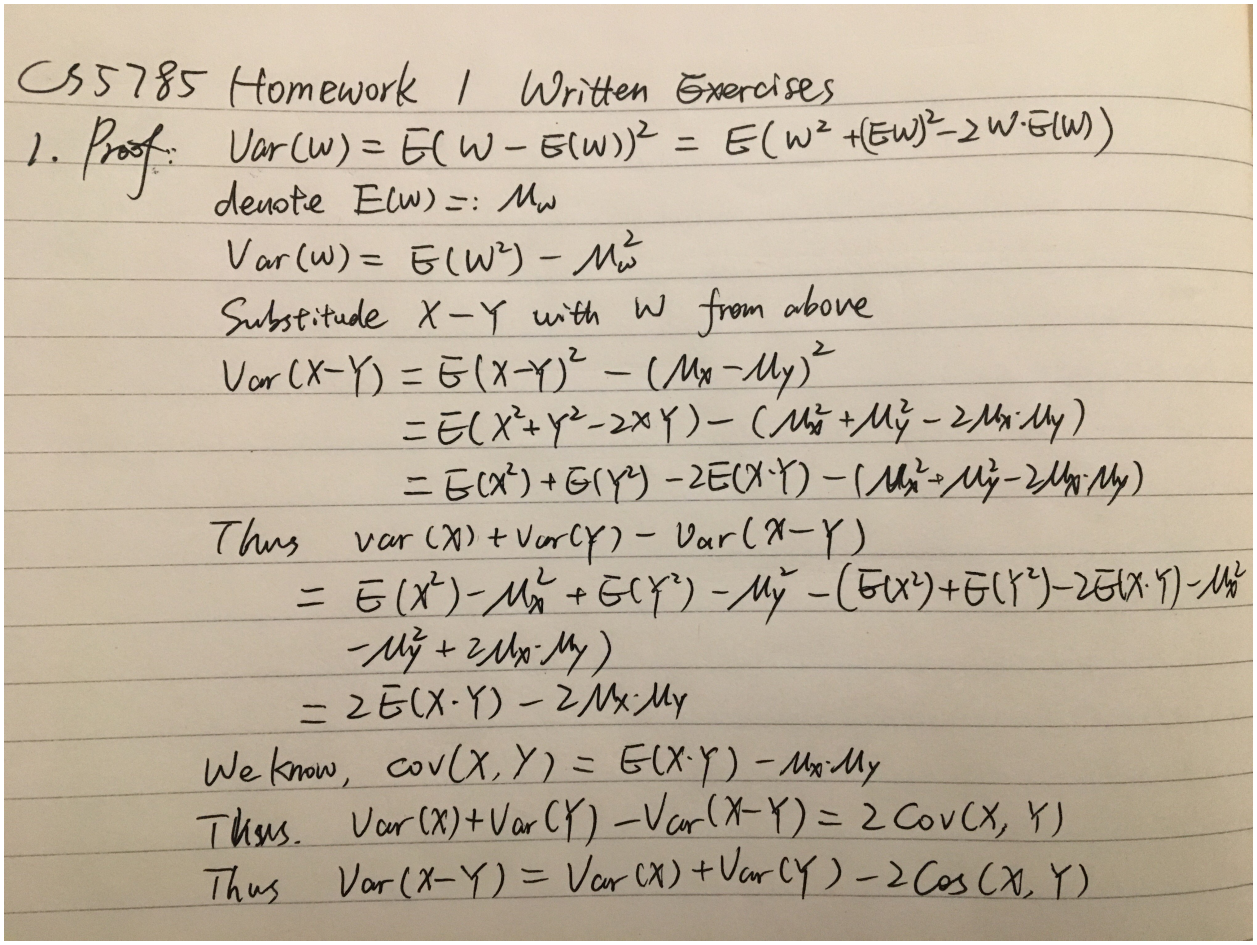
        fare: people who has priviledge or intimate relationship with the system could have lower fare to get on ship as they are more likely to get the lifeboats

        Embarked: different on board location may determine their attitude toward whether to live or give the chance to the others
    3. The classifier ranked 6487 with score of 0.76076.



3. Written Part

1.



CS5785 Homework 1 Written Exercises

1. Proof: $Var(W) = E(W - E(W))^2 = E(W^2 + (EW)^2 - 2W \cdot E(W))$

denote $E(W) =: \mu_W$

$Var(W) = E(W^2) - \mu_W^2$

Substitude $X - Y$ with W from above

$Var(X-Y) = E(X-Y)^2 - (\mu_X - \mu_Y)^2$

$= E(X^2 + Y^2 - 2XY) - (\mu_X^2 + \mu_Y^2 - 2\mu_X \cdot \mu_Y)$

$= E(X^2) + E(Y^2) - 2E(X \cdot Y) - (\mu_X^2 + \mu_Y^2 - 2\mu_X \cdot \mu_Y)$

Thus $Var(X) + Var(Y) - Var(X-Y)$

$= E(X^2) - \mu_X^2 + E(Y^2) - \mu_Y^2 - (E(X^2) + E(Y^2) - 2E(X \cdot Y) - \mu_X^2 - \mu_Y^2 + 2\mu_X \cdot \mu_Y)$

$= 2E(X \cdot Y) - 2\mu_X \cdot \mu_Y$

We know, $cov(X, Y) = E(X \cdot Y) - \mu_X \cdot \mu_Y$

Thus. $Var(X) + Var(Y) - Var(X-Y) = 2Cov(X, Y)$

Thus $Var(X-Y) = Var(X) + Var(Y) - 2Cos(X, Y)$

2.

CS5785 Homework 1 Written Exercise

2. (a) denote D as test result for widget is defective
d as widget is actually defective.

We have $P(D) = P(D|d)\cdot P(d) + P(D|d^c)\cdot P(d^c)$

$P(d) = 1/100000 = 10^{-5}$

$P(D|d) = 0.95 \quad P(D|d^c) = 1 - P(D|d) = 0.05$

The chance that a widget is actually defective given the test result is defective is $P(d|D)$, by Bayes Theorem:

$$P(d|D) = \frac{P(D|d)\cdot P(d)}{P(D|d)\cdot P(d) + P(D|d^c)\cdot P(d^c)}$$

$$= \frac{0.95\times 10^{-5}}{0.95\times 10^{-5} + 0.05\times(1-10^{-5})}$$

$$= 0.00018997$$

(b) denote T as the number of widgets thrown away per year
T(d) as the number of defective widgets in T per year
T(g) as the number of good widgets in T per year
d as the number of actually defective widgets per year

We have $T = P(D)\times 10^7 = 5\times 10^5 + 90$

$T(d) = P(D|d)\cdot P(d)\times 10^7 = 0.95\times 10^{-5}\times 10^7 = 95$

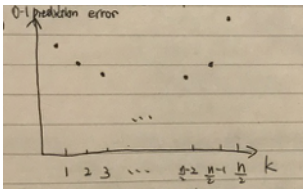
$d = 10^{-5}\times 10^7 = 100 = P(d)\times 10^7$

$T(g) = T - T(d) = 5\times 10^5 - 5 = 499995$

Thus, 499995 good widgets be thrown away each year. The num of bad widgets shipped ~~out~~ each year is $(d - T(d)) = 5$.

3. a. As k decrease from n to 1, the average 0-1 prediction error drops starting at n/2.

Because it start with misclassifying half of the points, as k becomes a reasonable value, the average 0-1 prediction error reaches its minima, however, as k keeps diminishing to 1, it is very likely to train a wrong model, as there are overlapping data between the two class, and in this case, the wrong model will have high average 0-1 prediction error, graphically, the trend oscillates along the expected path. As k = 1, as the training and testing are ovelaped, he average 0-1 prediction error is 0. Overall, the whole graph will looks like a wide U shape with exception of the point where k = 1, it is 0. Since k can only take integer, so it is a discrete graph whose trend is U shape.



b. The graph (above) will look something like this, but this one will be a little bit above the graph in part a, as the average error rate is larger when part of the data are removed. To be more precise, the right end of the U shape will be a little bit higher than the left end, because the classifier is determined by the randomness of the removal action comparing to all the points are taking into consideration in the first case, so the result will be an increase in the variance, that is the reason we are expecting the right hand side of the U shape to be a little bit higher than the right hand side.

c. Considering the validation accuracy, as k decreased from 20 to 10, k-fold cross validation reduces its variance while increase the bias; when k decrease from 5 to 2, the sample size get smaller, so the variance will increase du to the instability of the training set themselves, but overall the variance of the k-fold cross validation should be approximately the same as we repeat doing validation and average out the high variance, however, this would significantly increase the requirement for computational resources. Overall, some magic number in between the two extreme cases will be a great choice. So I will say k=5 is fairly good.

d. We can apply weight on the features in K-NN classifier. To be precise, we weight the closer point higher and weight distant points lower using an linear model. So in this case, a distant data point only make a little effect on training the classifier.

e. When there are too many features, it it hard to quantify the notion of the distance among different features, and adding the up even with “different weight” will be misleading, plus it is really hard to find out a reasonable “different weight” in terms of coefficient when adding the distances up. Because of the reasons above, distance becomes unreliable to train the classifier as the K-NN model is trained on distance and the label, therefore, too many features may make our classifier instead of fitting the training data but being too general to classify the testing data. In addition, adding a feature means more computation, as there are a lot of features, the demand of computation power increases exponentially.