HOMEWORK 3

>>Anudeep Kumar<<
>>9084607069<<

Instructions: Use this latex file as a template to develop your homework. Submit your homework on time as a single pdf file to Canvas. Late submissions may not be accepted. Please wrap your code and upload to a public GitHub repo, then attach the link below the instructions so that we can access it. You can choose any programming language (i.e. python, R, or MATLAB). Please check Piazza for updates about the homework.
Please find the link below: GiHub Link Homework 3

1 Questions (50 pts)

1. (9 pts) Explain whether each scenario is a classification or regression problem. And, provide the number of data points (n) and the number of features (p).

   (a) (3 pts) We collect a set of data on the top 500 firms in the US. For each firm we record profit, number of employees, industry and the CEO salary. We are interested in predicting CEO salary with given factors.

   As we can observe the CEO salary has to be mapped using continuous data profit and number of employees. The salary depends on the above two features and hence it is **regression**.

   (b) (3 pts) We are considering launching a new product and wish to know whether it will be a success or a failure. We collect data on 20 similar products that were previously launched. For each product we have recorded whether it was a success or failure, price charged for the product, marketing budget, competition price, and ten other variables.

   In this case the output can be a success or failure and given a new input we predict either success or failure hence we can safely conclude its a classification problem

   (c) (3 pts) We are interesting in predicting the % change in the US dollar in relation to the weekly changes in the world stock markets. Hence we collect weekly data for all of 2012. For each week we record the % change in the dollar, the % change in the US market, the % change in the British market, and the % change in the German market.

   Here the features are change in dollar , the change in US Market, British Market and German Market. Hence to predict % Change in US dollar we need to use regression as the output is continuous data.

2. (6 pts) The table below provides a training data set containing six observations, three predictors, and one qualitative response variable.

X_1	X_2	X_3	Y
0	3	0	Red
2	0	0	Red
0	1	3	Red
0	1	2	Green
-1	0	1	Green
1	1	1	Red

   Suppose we wish to use this data set to make a prediction for Y when $X_1 = X_2 = X_3 = 0$ using K-nearest neighbors.

   (a) (2 pts) Compute the Euclidean distance between each observation and the test point, $X_1 = X_2 = X_3 = 0$.

X_1	X_2	X_3	Euclidean Distance
0	3	0	3
2	0	0	2
0	1	3	3.162
0	1	2	2.236
-1	0	1	1.414
1	1	1	1.732

.

(b) (2 pts) What is our prediction with $K = 1$? Why?

We know that for K=1 we see the closest point and assign the label hence the point $(-1, 0, 1)$ is closest and hence the Prediction= Green

(c) (2 pts) What is our prediction with $K = 3$? Why?

For K=3 we see the 3 nearest points. These are $(-1, 0, 1)$, $(1, 1, 1)$ & $(2, 0, 0)$ Here the majority vote gives the Prediction = Red

3. (12 pts) When the number of features p is large, there tends to be a deterioration in the performance of KNN and other local approaches that perform prediction using only observations that are near the test observation for which a prediction must be made. This phenomenon is known as the curse of dimensionality, and it ties into the fact that non-parametric approaches often perform poorly when p is large.

(a) (2pts) Suppose that we have a set of observations, each with measurements on $p = 1$ feature, X. We assume that X is uniformly (evenly) distributed on [0, 1]. Associated with each observation is a response value. Suppose that we wish to predict a test observation's response using only observations that are within 10% of the range of X closest to that test observation. For instance, in order to predict the response for a test observation with $X = 0.6$, we will use observations in the range [0.55, 0.65]. On average, what fraction of the available observations will we use to make the prediction?

We can observe that there are three regions here $[0, 0.5]$, $[0.5, 0.95]$ & $[0.95, 1]$.
For $x \in [0, 0.5]$ the observations will lie in the interval$[0, x + 0.05]$ and represents $x + 0.05$
similarly $x \in [0.05, 0.95]$ observation will lie in the interval$[x - 0.05, x + 0.05]$ and represents 0.1 and finally for $x \in [0.95, 1]$ will have observation in the interval$[x - 0.05, 1]$ and represent $1.05 - x$. Now to find the average we find the expectation and being uniform, the probability is same.

$$\int_0^{0.05} (x + 0.05)\,\mathrm{d}x + \int_{0.05}^{0.95} (0.1)\,\mathrm{d}x + \int_{0.95}^{1} (1.05 - x)\,\mathrm{d}x = 0.00375 + 0.09 + 0.00375 = 0.09375$$

(b) (2pts) Now suppose that we have a set of observations, each with measurements on $p = 2$ features, $X1$ and $X2$. We assume that predict a test observation's response using only observations that $(X1, X2)$ are uniformly distributed on [0, 1] × [0, 1]. We wish to are within 10% of the range of $X1$ and within 10% of the range of $X2$ closest to that test observation. For instance, in order to predict the response for a test observation with $X1 = 0.6$ and $X2 = 0.35$, we will use observations in the range [0.55, 0.65] for $X1$ and in the range [0.3, 0.4] for $X2$. On average, what fraction of the available observations will we use to make the prediction?

Assuming X1 and X2 are independent we can use the result above to multiply the expectation as we know $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]$ So in that case the average fraction is $0.09375^2 = 0.0087890625$

(c) (2pts) Now suppose that we have a set of observations on $p = 100$ features. Again the observations are uniformly distributed on each feature, and again each feature ranges in value from 0 to 1. We wish to predict a test observation's response using observations within the 10% of each feature's range that is closest to that test observation. What fraction of the available observations will we use to make the prediction?

So in this case the average fraction is $0.09375^{100} = 1.574446 * 10^{-103}$

(d) (3pts) Using your answers to parts (a)–(c), argue that a drawback of KNN when p is large is that there are very few training observations "near" any given test observation.

As we can observe from the above results as p increases the number of training observations tends to 0.

(e) (3pts) Now suppose that we wish to make a prediction for a test observation by creating a p-dimensional hypercube centered around the test observation that contains, on average, 10% of the training observations. For p =1, 2, and 100, what is the length of each side of the hypercube? Comment on your answer.

4. (6 pts) Supoose you trained a classifier for a spam detection system. The prediction result on the test set is summarized in the following table.

		Predicted class	
		Spam	not Spam
Actual class	Spam	8	2
	not Spam	16	974

Calculate

(a) (2 pts) Accuracy Accuracy is $\frac{TP+TN}{TP+TN+FN+FP} = \frac{8+974}{1000} = 0.982$

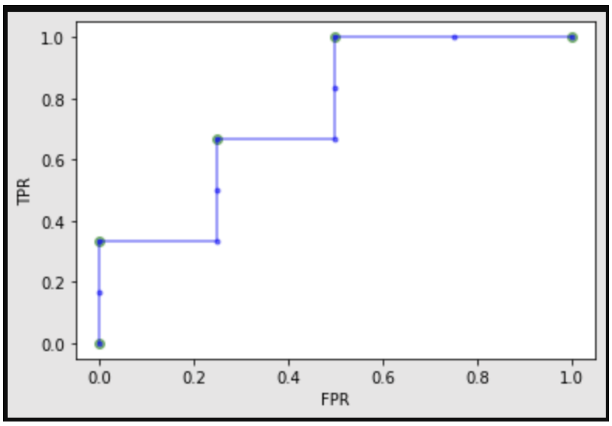
(b) (2 pts) Precision Precision is $\frac{TP}{TP+FP} = \frac{8}{8+16} = 0.0.333$

(c) (2 pts) Recall Recall is $\frac{TP}{TP+FN} = \frac{8}{8+2} = 0.8$.

5. (9pts) Again, suppose you trained a classifier for a spam filter. The prediction result on the test set is summarized in the following table. Here, "+" represents spam, and "-" means not spam.

Confidence positive	Correct class
0.95	+
0.85	+
0.8	-
0.7	+
0.55	+
0.45	-
0.4	+
0.3	+
0.2	-
0.1	-

(a) (6pts) Draw a ROC curve based on the above table.
The ROC Curve is defined as the TPR vs FPR where $TPR = \frac{TP}{TP+FN}$ and $FPR = \frac{FP}{FP+TN}$ i.e the Actual Probability of + and - . Below is the curve:



(b) (3pts) (Real-world open question) Suppose you want to choose a threshold parameter so that mails with confidence positives above the threshold can be classified as spam. Which value will you choose? Justify your answer based on the ROC curve.

In the real world we aim to reducing the number of spam mails being classsified as important and vice versa.Hence we aim to minimize false positive ratio and we can manage with a trade off in TPR. Therefore, the point $(0.0, 0.5)$ which corresponds to 0.8 confidence ratio is the optimum point to keep as the threshold

6. (8 pts) In this problem, we will walk through a single step of the gradient descent algorithm for logistic regression. As a reminder,

$$f(x;\theta) = \sigma(\theta^\top x)$$

$$\text{Cross entropy loss } L(\hat{y}, y) = -[y \log \hat{y} + (1-y)\log(1-\hat{y})]$$

$$\text{The single update step } \theta^{t+1} = \theta^t - \eta \nabla_\theta L(f(x;\theta), y)$$

(a) (4 pts) Compute the first gradient $\nabla_\theta L(f(x;\theta), y)$.

Inserting the function

$$f(x;\theta) = \sigma(\theta^\top x)$$

in the loss function we get

$$L(f(x;\theta), y) = -[y \log \sigma(\theta^\top x) + (1-y)\log(1-\sigma(\theta^\top x))]$$

Substituting value of $\sigma(\theta^\top x) = \frac{1}{1+e^{-\theta^\top x}}$ and Differentiating with respect to θ

$$\nabla_\theta L(f(x;\theta), y) = -[y(1+e^{-\theta^\top x})\frac{e^{-\theta^\top x}}{(1+e^{-\theta^\top x})^2}x + (1-y)\frac{1+e^{-\theta^\top x}}{e^{-\theta^\top x}}\frac{e^{-\theta^\top x}}{(1+e^{-\theta^\top x})^2}x$$

$$\nabla_\theta L(f(x;\theta), y) = -yx\frac{e^{-\theta^\top}}{(1+e^{-\theta^\top x})} + (x)\frac{1}{1+e^{-\theta^\top x}} + yx\frac{1}{(1+e^{-\theta^\top x})}$$

$$\nabla_\theta L(f(x;\theta), y) = -yx + \frac{x}{1+e^{-\theta^\top x}}$$

$$\nabla_\theta L(f(x;\theta), y) = x[\sigma(\theta^\top x) - y]$$

(b) (4 pts) Now assume a two dimensional input. After including a bias parameter for the first dimension, we will have $\theta \in \mathbb{R}^3$.

$$\text{Initial parameters : } \theta^0 = [0, 0, 0]$$

$$\text{Learning rate } \eta = 0.1$$

$$\text{data example : } x = [1, 3, 2], y = 1$$

Compute the updated parameter vector θ^1 from the single update step.

$$\theta^\top x = [0, 0, 0]^T[1, 3, 2] = [0]$$

$$\theta^1 = \theta^0 - \eta x[-y + \sigma(\theta^\top x)]$$

$$\theta^1 = [0, 0, 0] - 0.1[1, 3, 2][-1 + \frac{1}{1+e^0}]$$

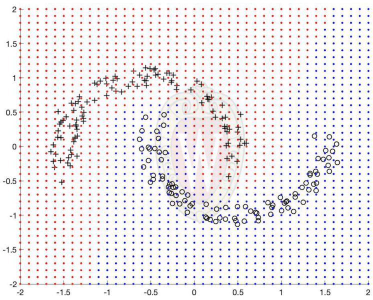
$$\theta^1 = [0, 0, 0] + \frac{1}{20}[1, 3, 2]$$

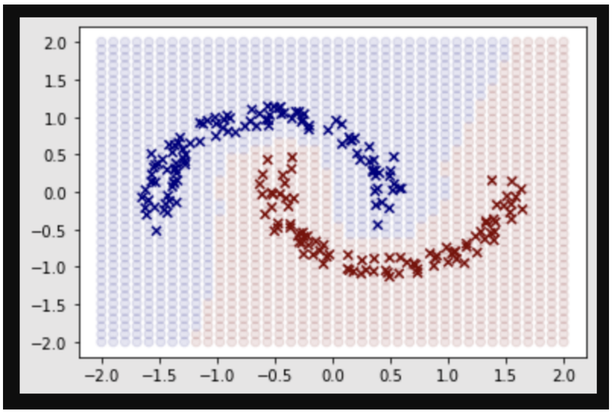
$$\theta^1 = [1/20, 3/20, 1/10]$$

2 Programming (50 pts)

1. (10 pts) Use the whole D2z.txt as training set. Use Euclidean distance (i.e. $A = I$). Visualize the predictions of 1NN on a 2D grid $[-2 : 0.1 : 2]^2$. That is, you should produce test points whose first feature goes over $-2, -1.9, -1.8, \ldots, 1.9, 2$, so does the second feature independent of the first feature. You should overlay the training set in the plot, just make sure we can tell which points are training, which are grid.

   The expected figure looks like this. Below is the result obtained





Spam filter Now, we will use 'emails.csv' as our dataset. The description is as follows.

Features

Label

Email No.	the	to	ect	and	for	of	a	you	hou	in	...	connevey	jay	valued	lay	infrastructure	military	allowing	ff	dry	Prediction
Email 1	0	0	1	0	0	0	2	0	0	0	...	0	0	0	0	0	0	0	0	0	0
Email 2	8	13	24	6	6	2	102	1	27	18	...	0	0	0	0	0	0	0	1	0	0
Email 3	0	0	1	0	0	0	8	0	0	4	...	0	0	0	0	0	0	0	0	0	0
Email 4	0	5	22	0	5	1	51	2	10	1	...	0	0	0	0	0	0	0	0	0	0
Email 5	7	6	17	1	5	2	57	0	9	3	...	0	0	0	0	0	0	0	1	0	0

- Task: spam detection
- The number of rows: 5000
- The number of features: 3000 (Word frequency in each email)
- The label (y) column name: 'Predictor'
- For a single training/test set split, use Email 1-4000 as the training set, Email 4001-5000 as the test set.

- For 5-fold cross validation, split dataset in the following way.
  - Fold 1, test set: Email 1-1000, training set: the rest (Email 1001-5000)
  - Fold 2, test set: Email 1000-2000, training set: the rest
  - Fold 3, test set: Email 2000-3000, training set: the rest
  - Fold 4, test set: Email 3000-4000, training set: the rest
  - Fold 5, test set: Email 4000-5000, training set: the rest

2. (8 pts) Implement 1NN, Run 5-fold cross validation. Report accuracy, precision, and recall in each fold.

   For each fold below is the reported Accuracy Precision and Recall :

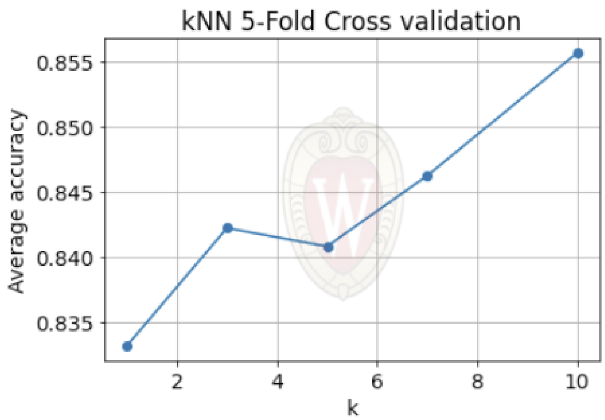
Fold	Accuracy	Precision	Recall
1	0.821	0.6480447	0.8140350
2	0.853	0.6857142	0.8664259
3	0.861	0.7217125	0.8309859
4	0.853	0.7220543	0.812925
5	0.778	0.6105263	0.7581699

3. (12 pts) Implement logistic regression (from scratch). Use gradient descent (refer to question 6 from part 1) to find the optimal parameters. You may need to tune your learning rate to find a good optimum. Run 5-fold cross validation. Report accuracy, precision, and recall in each fold.

   For each fold below is the reported Accuracy Precision and Recall :

Fold	Accuracy(%)	Precision	Recall
1	90.9	0.83916	0.84210
2	88.3	0.88095	0.66787
3	85.8	0.90340	0.55985
4	88.4	0.76969	0.86394
5	52.9	0.39101	0.96732

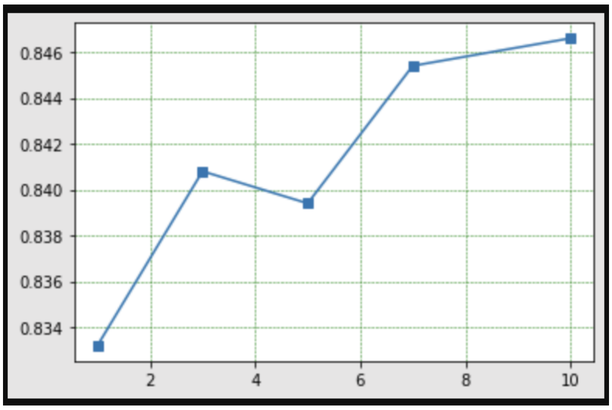
4. (10 pts) Run 5-fold cross validation with kNN varying k (k=1, 3, 5, 7, 10). Plot the average accuracy versus k, and list the average accuracy of each case.
   Expected figure looks like this.

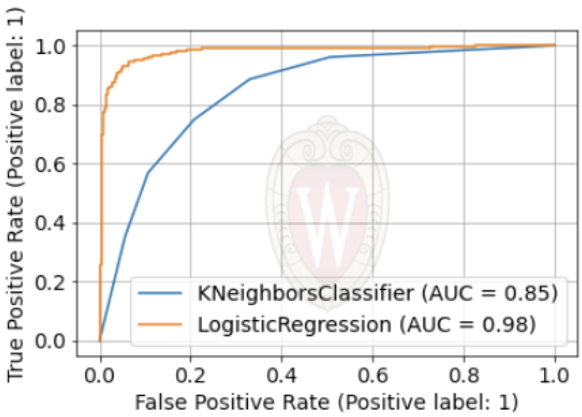


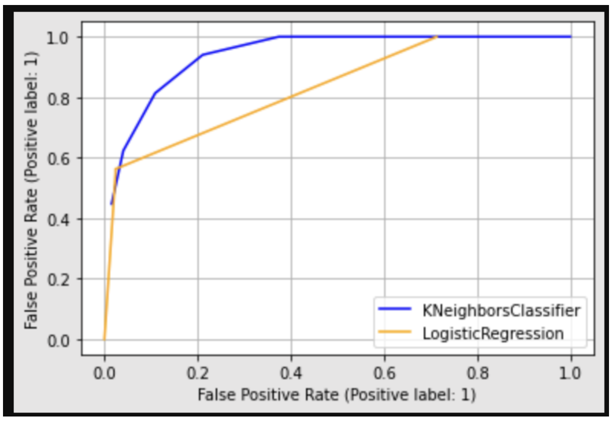
   We obtain the below figure

5. (10 pts) Use a single training/test setting. Train kNN (k=5) and logistic regression on the training set, and draw ROC curves based on the test set.
   Expected figure looks like this. Note that the logistic regression results may differ.

   Below is the figure obtained with KNN=5 and Log regression with alpha=0.1 and iteration 500


0.846
0.844
0.842
0.840
0.838
0.836
0.834
2
4
6
8
10


True Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)
KNeighborsClassifier (AUC = 0.85)
LogisticRegression (AUC = 0.98)
0.0
0.2
0.4
0.6
0.8
1.0


False Positive Rate (Positive label: 1)
False Positive Rate (Positive label: 1)
KNeighborsClassifier
LogisticRegression
0.0
0.2
0.4
0.6
0.8
1.0