DIEGO MARTIN CRESPO A20432558

CS 584-04: Machine Learning

Fall 2018 Midterm Test

Question 1 (50 points)

Anyone can use Chicago's 311 Service Request to report street potholes. After a request has been received, the Department of Transportation will first assess the severity of the pothole, and then schedule road crew to fill up the pothole. After the pothole is filled, the service request will be closed.

You are provided with this CSV file **ChicagoCompletedPotHole.csv** for analyzing the city's efforts to fill up street potholes. The data contains 17,912 observations. Each observation represents a completed request which was created between December 1, 2017 and March 31, 2018 and was completed between December 4, 2017 and September 12, 2018. The data has the following seven variables:

Name	Level	Description
1) CASE_SEQUENCE	Nominal	A unique index for identifying an observation
2) WARD	Nominal	The Chicago's ward number from 1 to 50
3) CREATION_MONTH	Nominal	Calendar month when the request was created
4) N_POTHOLES_FILLED_ON_BLOCK	Interval	Number of potholes filled on the city block
5) N_DAYS_FOR_COMPLETION	Interval	Number of days between CREATION_DATE and COMPLETION_DATE inclusively
6) LATITUDE	Interval	Latitude of the city block
7) LONGITUDE	Interval	Longitude of the city block

You will first identify clusters in the data, and then use a classification tree to profile the clusters. Here are the specifications for performing the respective analyses.

K-Means Clustering

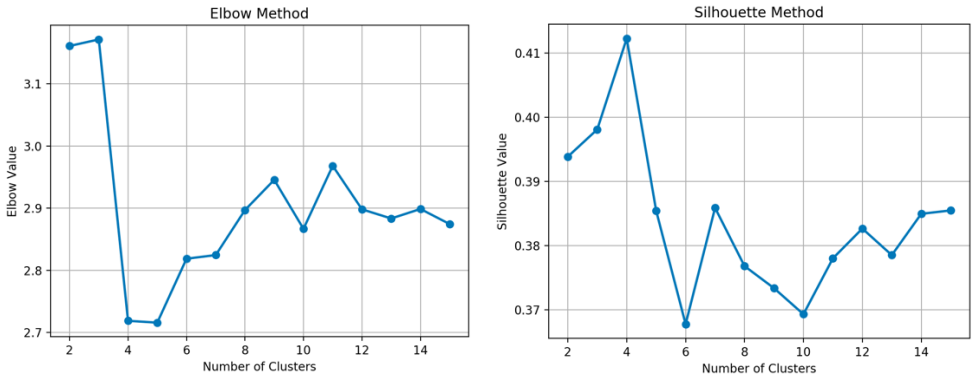
1. Use $\log_e$(N_POTHOLES_FILLED_ON_BLOCK), $\log_e$(1 + N_DAYS_FOR_COMPLETION), LATITUDE, and LONGITUDE (i.e., you need to perform the transformations before clustering)
2. The maximum number of clusters is 15 and the minimum number of clusters is 2
3. The random seed is 20181010
4. Use both the Elbow and the Silhouette methods to determine the number of clusters

Classification Tree

1. The target variable is the Cluster ID
2. Use N_POTHOLES_FILLED_ON_BLOCK, N_DAYS_FOR_COMPLETION, LATITUDE, and LONGITUDE (without any transformations) as the predictors
3. The maximum number of branches is 2
4. The maximum depth is 2
5. The random seed is 20181010.
6. The grow criterion is Entropy

Please answer the following questions.

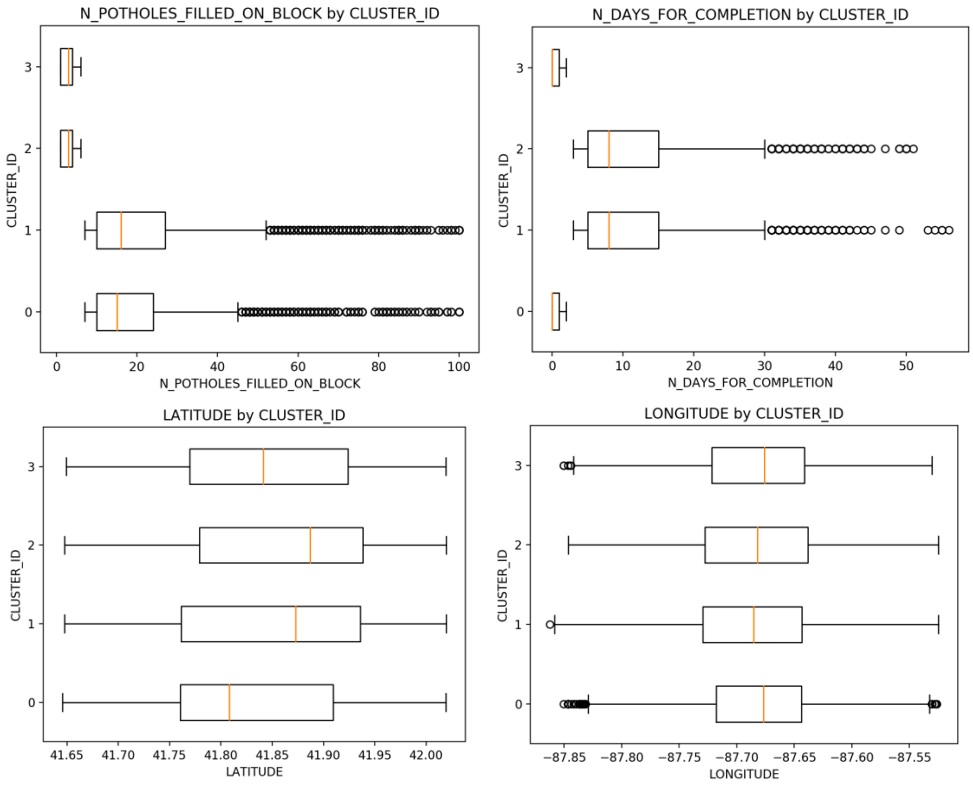
a) (10 points) How many clusters did you determine? Please provide the Elbow and the Silhouette charts and state your arguments. The charts must be properly labeled.
"Exam_Q1_A.py"



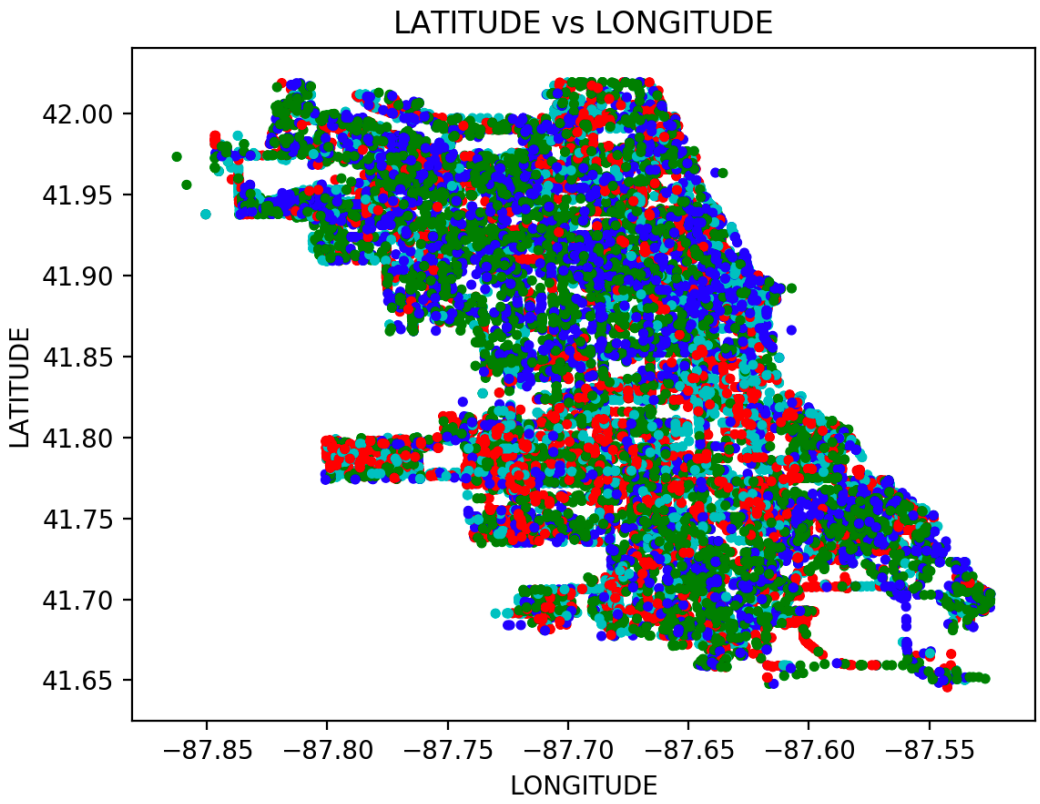
- The Elbow method suggests a number between 4 and 5 (values at the graph knee).
- The silhouette method suggests a number of clusters of 4 (highest value of the graph).

Therefore, we determine that 4 is the best number of clusters.

b) (5 points) Create a box-plot for each of these four variables: N_POTHOLES_FILLED_ON_BLOCK, N_DAYS_FOR_COMPLETION, LATITUDE, and LONGITUDE, grouped by the Cluster ID.
Using 4 clusters:
"Exam_Q1_B-C.py"



c) (5 points) Generate a scatterplot of LATITUDE (y-axis) versus LONGITUDE (x-axis) using the Cluster ID as the color response variable. You may need to adjust the marker size and set the aspect ratio to one in order to make the scatterplot more readable.
"Exam_Q1_B-C.py"



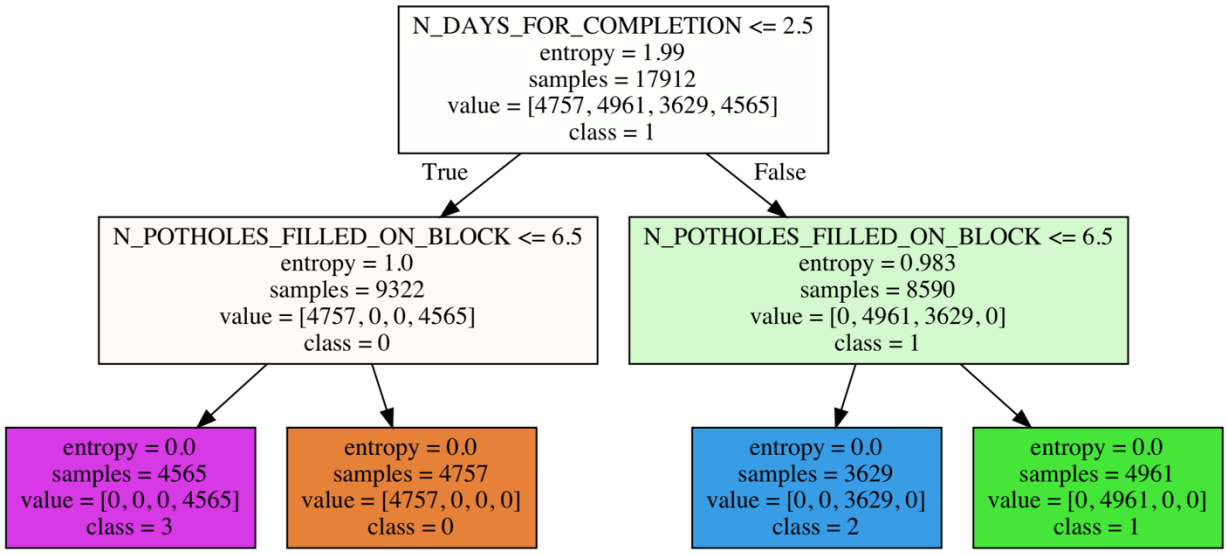
- Red: CUSTER_ID= 0
- Green: CUSTER_ID= 1
- Blue: CUSTER_ID= 2
- Cian: CUSTER_ID= 3

d) (5 points) Comment your scatterplot in (c). In particular, how effective or ineffective do you think the clustering analysis in dividing up the observations according to their geographical locations?
The clustering in 4 groups dividing the observations according to their geographical location is not effective. As it can be inferred in the previous image all the different clusters (each one corresponding to one different color) do not seem to follow some kind of ordered or grouped by colors. They seem to be all mixed, therefore it is not useful to use this method for dividing up the observations according their geographical location (latitude, longitude).

e) (10 points) How many leaves did you have in your classification tree? Please attach the tree diagram which must be properly labeled.

"Exam_Q1_tree.py"

There are 4 leaves, as it can be inferred in the following image:



f) (5 points) Calculate the Misclassification Rate and the Root Mean Squared Error of your classification tree.

"Exam_Q1_tree.py"

- The Misclassification or error rate is defined as: $Misclassification = 1 - Accuracy$. The obtained accuracy is 1. Therefore, Misclassification is equal to 0.
- RMSE obtained is 0.

g) (10 points) Based on your classification tree, how would you describe the profiles of the clusters?

There are 4 types of profiles:

- Class 0
  - N_DAYS_FOR_COMPLETION <= 2.5 & N_POTHOLES_FILLED_ON_BLOCK > 6.5
  - 4757 samples
- Class 1
  - N_DAYS_FOR_COMPLETION > 2.5 & N_POTHOLES_FILLED_ON_BLOCK > 6.5
  - 4961 samples
- Class 2
  - N_DAYS_FOR_COMPLETION > 2.5 & N_POTHOLES_FILLED_ON_BLOCK <= 6.5
  - 3629 samples
- Class 3
  - N_DAYS_FOR_COMPLETION <= 2.5 & N_POTHOLES_FILLED_ON_BLOCK <= 6.5
  - 4565 samples

They are pure clusters

Question 2 (50 points)

In the automobile industry, a common question is how likely a policy-holder will file a claim during the coverage period. Your task is to build several models. After evaluating and comparing the models, you will recommend the model that performs better. In order to avoid discriminating policy-holders, we will use predictors that can be verified and are related to the risk exposures of the policy-holders. The CSV file policy_2001.csv contains data about 617 policy-holders. We will use only the following variables.

Target Variable

- CLAIM_FLAG: Claim Indicator (1 = Claim Filed, 0 = Otherwise) and 1 is the event value.

Nominal Predictor

- CREDIT_SCORE_BAND: Credit Score Tier ('450 – 619', '620 – 659', '660 – 749', and '750 +')

Interval Predictors

- BLUEBOOK_1000: Blue Book Value in Thousands of Dollars (min. = 1.5, max. = 39.54)
- CUST_LOYALTY: Number of Years with Company Before Policy Date (min. = 0, max. ≈ 21)
- MVR_PTS: Motor Vehicle Record Points (min. = 0, max. = 10)
- TIF: Time-in-Force (min. = 101, max. = 107)
- TRAVTIME: Number of Miles Distance Commute to Work (min. = 5, max. ≈ 93)

Since the tools may not take the nominal predictor as is, you will first derive the dummy indicators from the nominal predictors and then use the dummy indicators in building the models. You will build the three models according to the following specifications.

Nearest Neighbors Model

- The number of neighbors is 3
- The distance metric is the standard Euclidean distance
- The search algorithm is the brute-force method

Classification Tree Model

- The maximum number of depths is 10
- The splitting criterion is Entropy
- The random seed is 20181010

Logistic Model

- The optimization algorithm is the Newton-Raphson method
- The maximum number of iterations is 100
- The relative error in parameter estimates acceptable for convergence is 1E-8
- The Intercept term must be included in the model

You will divide the data into the Training and the Testing partitions. You will build and evaluate the three models using the Training partition. Later, you will recommend one model based on the evaluation and the comparison results from the Testing partition.

Data Partition

- The Training partition consists of 70% of the original observations, the remaining 30% goes to the Testing partition.
- The claim rates (i.e., the fraction of observations whose CLAIM_FLAG is 1) must be the same in both partitions.
- The random seed is 20181010.

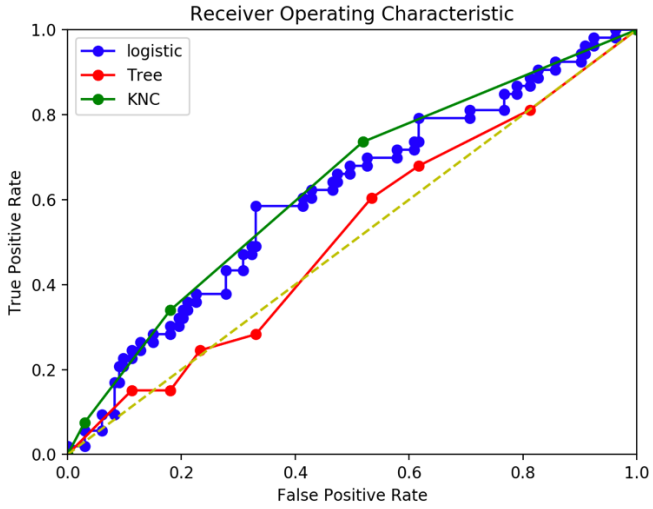
Please answer the following questions.

a) (5 points) How many observations are in the Training and the Testing partitions?
"Exam_Q2_A-B.py"
   - Total of 617 observations
     - 431 observations for the Training partition.
     - 186 observations for the Testing partition.

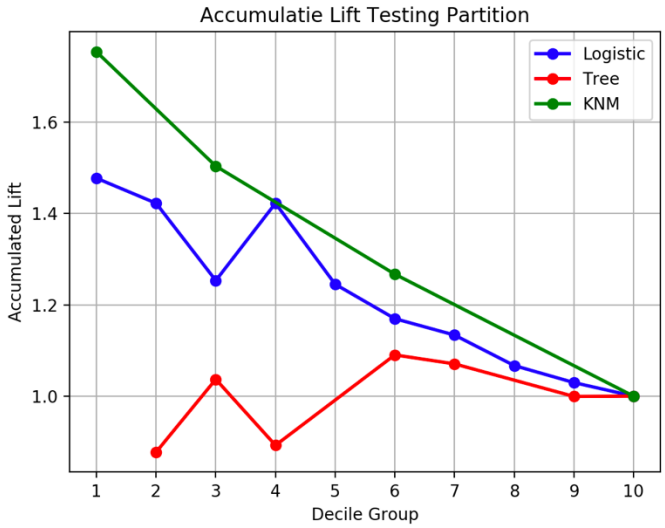
b) (5 points) What are the claim rates in the Training and the Testing partitions?
"Exam_Q2_A-B.py"
   - Claim rate for training partition: 0.287703016.
   - Claim rate for testing partition: 0.284946236.

c) (10 points) Use **the claim rate in the Training partition** as the probability threshold in the misclassification rate calculation. A claim is predicted if the predicted probability of filing a claim is greater than or equal to the probability threshold. Calculate the Area Under Curve metric, the Root Mean Squared Error metric, and the Misclassification Rate for all three models **using the Testing partition**. List the metrics as the rows and the models as the columns in a table.
   - **Nearest Neighbors Model:** "Exam_Q2_KNM.py"
     i. Area Under Curve metric = 0.6319336
     ii. RMSE = 0.5632089
     iii. Misclassification Rate = 0.44623656
   - **ii. Classification Tree Model:** "Exam_Q2_tree.py"
     i. Area Under Curve metric= 0.5139736
     ii. RMSE= 0.3709677
     iii. Misclassification Rate = 0.38172043
   - **iii. Logistic Model:** "Exam_Q2_logistic.py"
     i. Area Under Curve metric = 0.60959
     ii. RMSE = 0.54870326
     iii. Misclassification Rate = 0.36021505

d) (10 points) Calculate (but no need to display) the coordinates of the Receiver Operating Characteristic curve for each of the three models using the Testing partition. Plot all three curves in the same chart but use a different color for each curve. The chart (including the axes, the title, and the curve legends) must be properly labeled.
"Exam_Q2_ROC-LIFT.py"



e) (10 points) Calculate (but no need to display) the coordinates of the Accumulated Lift chart for each of the three models using the Testing partition. Plot all three accumulated lift curves in the same chart but use a different color for each curve. The chart (including the axes, the title, and the curve legends) must be properly labeled.
"Exam_Q2_ROC-LIFT.py"



f) (10 points) Based on the evaluation and the comparison results in (c), (d), and (e), which single model will you recommend? Please state your reasons for your recommendation.

When we compare the results, we should look at the following metrics (only using the test partition):

- Higher AUC: the KNM seems to have the greatest.
- Higher lift in the first few deciles: the KNM also has the greatest values.
- Lower RMSE: the tree model has the lowest.
- Lower Misclassification: the logistic model has the lowest.

We conclude that the KNM seems to behave better than the other two models.