INDENG 242: Applications in Data Analysis, Fall 2022

Midterm Exam

October 2022

Instructions: Please refer to the Honor Code that you signed and the Midterm Exam Instructions document posted on bCourses.

1 Short Answer Questions – 40 Points

Instructions: For these questions, it is not sufficient to simply state your answer. Instead, you must state your answer and write one or more sentences explaining the reasoning for your answer. Each question is worth 5 points.

1. Suppose that we train a classification model that has accuracy equal to 0.99 on the test set, and that the test set contains at least one positive observation and at least one negative observation. Then, without any other information, the most definitive statement we can make about the TPR (true positive rate) of that model on the test set is:

   A. The TPR is equal to 0.99

   B. The TPR is equal to 1

   C. The TPR is at least 0.90

   D. The TPR is between 0 and 1

2. Suppose that, conditioned on $Y = 1$, X is normally distributed with mean 4 and variance 1. Similarly, conditioned on $Y = 2$, X is normally distributed with mean 5.5 and variance 1. Now, given a new observation $X = x$, we are interested in predicting whether $Y = 1$ or $Y = 2$. A threshold value of 4.35 is chosen, so that we predict $Y = 2$ if $x \geq 4.35$ and $Y = 1$ if $x < 4.35$. This is represented pictorially in Figure 1.

   Figure 1

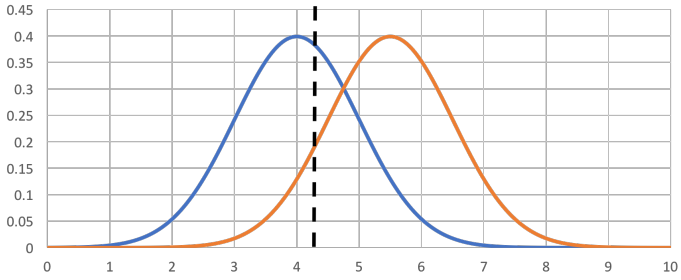


   Figure 2 defines five different shaded regions within Figure 1, and the letters refer also to their respective areas.

   Suppose that $Y = 1$ corresponds to a positive outcome. Then the FPR (false positive rate) is equal to:

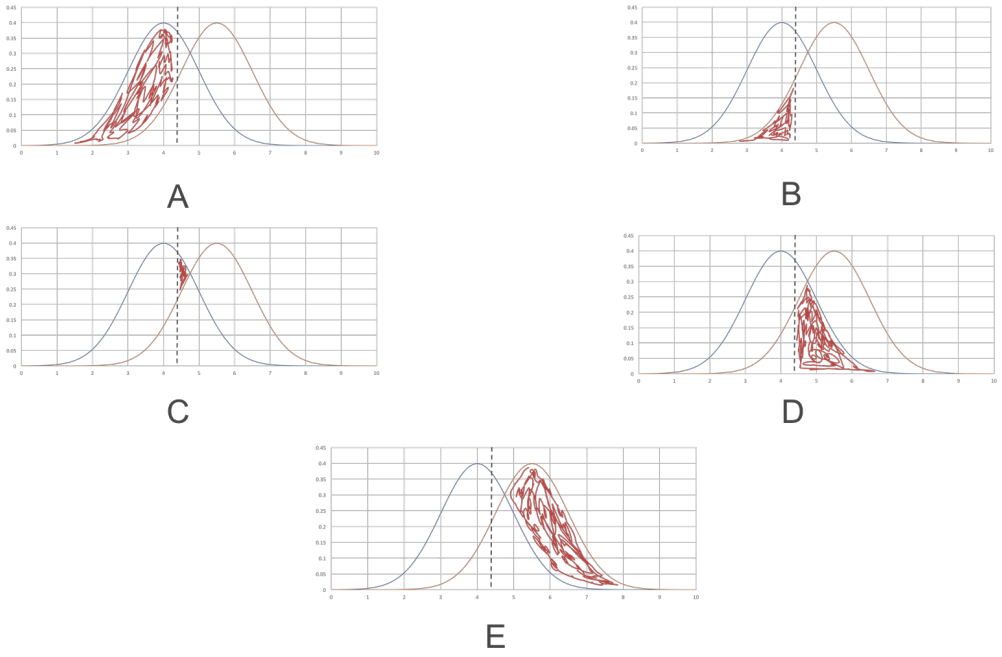
   A. $(C + D)/(A + B + C + D)$

   B. $(C + D)/(A + B)$

   C. $B/(D + E)$

   D. $B/(B + D + E)$

Figure 2



3. Consider a training set with $n = 1000$ data points. Compare the computation time of using 10-fold cross validation and leave-one-out cross validation to select the `cp` parameter in CART.

4. In a binary classification problem, what is an effective method for building a model and generating a confidence interval for its TPR (true positive rate) after splitting the data into training set and testing set? Specifically, in which dataset do you fit the model, and where to construct a bootstrap confident interval?

5. Write down the pseudo-algorithm (and appropriate comments/explications) for random forest (RF) and boosting. What is the advantage of RF computationally over tree-boosting? (Hint: think about how trees are generated in the two algorithms)

6. I collect a set of data (n = 100 observations) containing a single predictor and a quantitative response. I then fit a linear regression model to the data, as well as a separate quadratic regression, i.e. $Y = \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon$.

   (a) Suppose that the true relationship between X and Y is linear, i.e. $Y = \beta_0 + \beta_1 x + \epsilon$. Consider the training residual sum of squares (RSS) for the linear regression, and also the training RSS for the quadratic regression. Would we expect one to be lower than the other, would we expect them to be the same, or is there not enough information to tell? Justify your answer.

   (b) Answer (a) using test rather than training RSS.

   Provide a short example on Jupyter to justify your reasoning, setting `np.random.seed(10)` and using the ϵ `= np.random.normal(0, 1, 100)` to generate the random white noise.

7. Which of the following actions has the least risk of increasing the likelihood of overfitting?

   A. Increasing the number of trees/iterations when training a boosting model.

   B. Introducing new independent variables in a linear regression model that are quadratic functions of the original set of independent variables.

   C. Decreasing the value of m (`max_features` on sklearn) when training a random forests model while leaving the number of trees fixed.

   D. Increasing the number of trees when training a random forests model while leaving the value of m (`max_features` on sklearn) fixed.

8. Suppose that we trained a Random Forest model and a Boosting model. The test set OSR^2 value of the Random Forest model is 0.632, and the test set OSR^2 value of the Boosting model is 0.641. Given this information, can you confidently conclude that one model will have a better OSR^2 value when making future predictions? What can be done to be more confident about which is the better model?

2 Calculation Questions – 30 Points

Instructions: Please provide justification and show your work at all steps. Please also clearly state your final answer. Your grade will depend on the clarity of your response, the reasoning you have used, as well as the correctness of your answer.

1. (9 points) Consider deriving the probability that a given observation is part of a bootstrap sample. Suppose that we obtain a bootstrap sample from a set of n observations.

   Please answer the following questions.

   (a) (3 points) What is the probability that the second bootstrap observation is not the jth observation from the original sample?

   (b) (3 points) Argue that the probability that the jth observation is not in the bootstrap sample is $(1 - \frac{1}{n})^n$.

   (c) (3 points) When $n = 3$, what is the probability that the jth observation is in the bootstrap sample?

2. (6 points) Consider a sample of students in a machine learning class with variables $X_1 =$ `hours studied`, $X_2 =$ `undergrad GPA`, and $Y =$ `receive an A`. Fitting a logistic regression for estimating Y, we get $\hat{\beta}_0 = 5$, $\hat{\beta}_1 = 0.045$, $\hat{\beta}_2 = 1.5$. Please answer the following questions.

   (a) (3 points) Estimate the probability that a student who studies for 30 hour and has an undergrad GPA of 3.0 gets an A in the class.

   (b) (3 points) How many hours would the student in part (a) need to study to have a 50% chance of getting an A in the class

3. (9 points) PGA is the professional golf tour and they have been collecting some statistics to better understand trends in their league. In particular, they have collected the amount won per season (in millions of $) together with some statistics about players performances (average number of strokes per round, average number of putts per hole etc). They ran the linear regression and they obtain the results shown in 3

   (a) (3 points) Based on the output on the tables 3, is there enough evidence to conclude that the true coefficient corresponding to `AverageScore` is not equal to 0? On what have you based your answer? What about `DrivingAccuracy`?

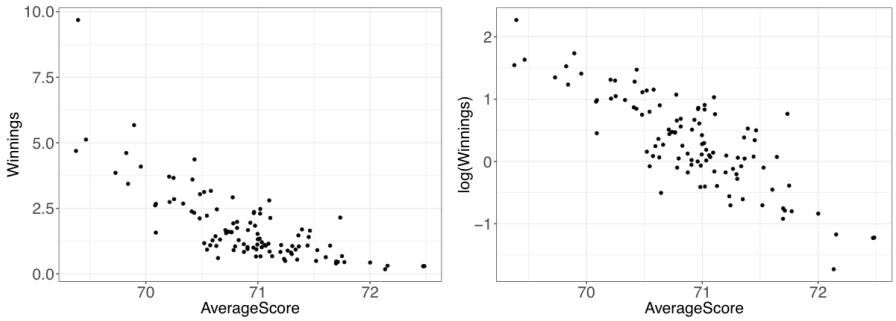
```
Call:
lm(formula = Winnings ~ AverageScore + AveragePutts + AverageDrivingDist +
    DrivingAccuracy, data = golf_train)


Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       114.161145  14.961424   7.630 1.40e-11 ***
AverageScore       -1.745918   0.186456  -9.364 2.46e-15 ***
AveragePutts        2.192717   4.374250   0.501    0.617
AverageDrivingDist  0.026401   0.017066   1.547    0.125
DrivingAccuracy    -0.003636   0.029532  -0.123    0.902
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.89 on 100 degrees of freedom
Multiple R-squared:  0.6087,Adjusted R-squared:  0.5931
F-statistic: 38.89 on 4 and 100 DF,  p-value: < 2.2e-16
```

Figure 3

(b) (3 points) Consider adding a new independent variable to the model called `AveragePuttsPerRound`, which is equal to the average number of putts per 18 hole round. (NB: `AveragePutts` is the average number of putts per hole, and you may assume that each round consists of exactly 18 holes.) Is it possible for this new variable to improve the linear regression model for predicting `Winnings`? Explain your answer.

(c) (3 points) A data scientist working with the PGA Tour has determined that a simple linear regression model that only uses a single independent variable `AverageScore`, would strike the best balance between interpretability and performance in this application domain. The data scientist is considering using one of two possible dependent variables: `Winnings` as before, or a logarithmic transformation `log(Winnings)`. Figure 4 shows scatter plots on the training data of these two possible dependent variables versus `AverageScore`. Based on Figure 4, which dependent variable choice would you recommend in order to get the best predictive performance?

Figure 4



4. (6 points) James Chapter 4, Exercise 2 (LDA decision rule).

3 Data Analysis Exercises – 30 Points

Instructions: These questions will involve a coding aspect as well as written responses. Your code must be clearly documented so as to explain what the code is doing. You are encouraged to submit a Jupyter notebook that includes your code, documentation, output, and written responses all together. Alternatively, you may submit your written responses and include your code, documentation, and output as an appendix. Please be appropriately succinct but provide sufficient justification and explanations for all written responses. Your grade will depend on the clarity of your response, the reasoning you have used, the correctness of your answer, and the extent to which your code and output correctly justifies your answer.

1. (15 points) Yelp is a widely popular platform that publishes information and reviews of local businesses such as restaurants, plumbers, hair salons, and others. Any user of Yelp is able to write a review, and each review includes a star rating between 1 and 5 in addition to written comments. In this problem, you will build models for predicting the star ratings of restaurants in Las Vegas, Nevada based on attributes contained in their Yelp profiles.

   The data for this problem is contained in the files `yelp242_train.csv` and `yelp242_test.csv`, and was retrieved from the larger Yelp Dataset provided by Yelp.

   We have performed a random 70/30 split, resulting in 6,272 observations in the training set and 2,688 observations in the test set. Each observation contains the average star rating, number of reviews, and a list of attributes collected from the Yelp page of a particular restaurant in the Las Vegas area. These attributes are described in Table 1. Note that **variable selection is not required** for this problem. NB: in order to have a standardise result, set the seed/random state to 10 at the beginning of the notebook (to prevent random disturbance in the cross validation of part b.

Table 1: Variables in the dataset `yelp242`.

Variable	Description	Levels	Missing rate
`stars`	The average star rating of the business (from 1 to 5).		0.0%
`review_count`	The number of reviews received by the business.		0.0%
`GoodForKids`	Whether this business is good for kids.	T, F, (Missing)	30.19%
`Alcohol`	The kind of alcohol provided at this business.	Beer_and_wine, full_bar, none, (Missing)	34.43%
`BusinessAccepts CreditCards`	Whether the business accepts credit cards.	T, F, (Missing)	6.15%
`WiFi`	Whether the business provides WiFi.	free, no, paid, (Missing)	32.82%
`BikeParking`	Whether bike parking is available at the business.	T, F, (Missing)	29.39%
`ByAppointment Only`	Whether the business is by appointment only.	T, F, (Missing)	87.86%
`Wheelechair Accessible`	Whether the business is wheelechair accessible.	T, F, (Missing)	74.43%
`OutdoorSeating`	Whether the business provides outdoor seating.	T, F, (Missing)	27.69%
`Restaurants Reservations`	Whether the business takes any reservation.	T, F, (Missing)	31.60%
`DogsAllowed`	Whether the business allows dogs.	T, F, (Missing)	72.65%
`Caters`	Whether the business provides catering.	T, F, (Missing)	34.12%

a) (3 points) There are many missing entries in this dataset, denoted by **`(Missing)`** in the data files. In particular, all of the attribute features contain missing values and Table 1 reports the percentage of observations where each attribute is missing. In general, there are several approaches for dealing with missing values in supervised learning. Each attribute with missing values in our dataset is a categorical feature and, in the subsequent models that you will build, you should treat **`(Missing)`** as an explicit category. Do you think this modeling approach is reasonable or not? Explain your answer.

b) (9 points) Let us start by building regression models for predicting **`stars`** based on all of the provided features listed in Table 1. All of your models should, of course, be built only using the training data provided in the **`yelp242_train.csv`** file.

i) First build a linear regression model. Remember to use all of the provided independent variables, and you do not have to do variable selection in this problem. For each of the categorical variables, you should use **`(Missing)`** as the reference level to be incorporated into the intercept term. This does not affect the predictive performance of the model, but it does lead to a cleaner interpretation. This can be achieved in statsmodels with a slight modification to the R-style formulas. For example, you could use code like

```
"stars ~ review_count + C(GoodForKids, Treatment(reference='(Missing)'))"
```

for a model regressing **stars** on **review_count** and **GoodForKids**.[1]

ii) Now build a regression tree model (using an implementation of the CART algorithm). Select the complexity parameter (i.e., **ccp_alpha** in sklearn) value for the tree through 5-fold cross-validation, and explain how you did the cross-validation and how you selected the complexity parameter value.

iii) Using the test set data provided in the **yelp242_test.csv** file, compute the OSR^2 values of your linear regression and regression tree models. Also, compute the MAE (mean absolute error) values of both models. How do you judge the performance of the two models?

c) (6 points) Please reset the seed to 10. Now build a random forest model, choosing the `max_features` by means of a 5-fold cross validation, explaining why we want to limit the number of features used at any split. Please make sure you are using one hot encoding and having the `Missing` category as a category for this exercise.

d) (9 points) Describe the bootstrap procedure as a general statistical procedure and explain why it is useful. Then, using bootstrap, find some confidence interval at 95% level for the differences in performance between the random forest model and the regression tree. How confident are you in telling there is a difference in the performance?

[1] More details may be found at
https://stackoverflow.com/questions/22431503/specifying-which-category-to-treat-as-the-base-with-statsmodels