DS5220 Supervised Machine Learning and Learning Theory: Homework 2

Due 2024/10/14 11:59pm

Instructions The homework must be submitted in **one pdf file including contents generated from the jupyter notebook**. Everyone needs to make a submission. Please write your nine-digit NUID and your northeastern.edu email. For the file names, please include your NUID. For example, "DS5220 HW2 (xxxxxxxxx)". Please specify the random seed at an appropriate place in your code chunk. It may not be the at the start, but specify it when you are executing codes that involve randomness. There are 100 points in total in this homework.

Instructions for Python To download `Python` packages, use

```
pip install packageName
```

To use the installed package, type

```
import packageName
```

Problem 1 [20pts]
In this problem, you will develop a model to predict whether a given car gets high or low gas mileage based on the `Auto` dataset. The `Auto` dataset has gas mileage, horsepower, and other information for cars. You can find the description of this dataset at `https://rdrr.io/cran/ISLR/man/Auto.html`. To load the data,

```
import pandas as pd
 from ISLP import load_data
 df = load_data('Auto')
 df = df[df['horsepower'].notna()]
```

where the last line is to remove the observations with missing values in `horsepower`.

(a) [3pts] Create a binary variable, `mpg01`, that contains a 1 if mpg contains a value above its median, and a 0 if mpg contains a value below its median. (Hint: You could compute the median using the median() function. You could add the `mpg01` column in `df`.)

(b) [3pts] Explore the data graphically in order to investigate the association between `mpg01` and the other features. Which of the other features seem most likely to be useful in predicting `mpg01`? Scatterplots and boxplots may be useful tools to answer this question. Describe your findings. (Hint: You may find

`pd.plotting.scatter_matrix()`

helpful and you can set the argument `figsize` as $(10, 10)$)

(c) [3pts] Split the data into a training set and a test set with 80% observations in the training set and 20% observations in the test set. (Hint: You may find

`from sklearn.model_selection import train_test_split`

helpful)

(d) [6pts] Perform logistic regression on the training data in order to predict `mpg01` using `cylinders`, `weight`, `displacement`, and `horsepower`. What is the test error of the model obtained? (Hint: You may find

`from sklearn.linear_model import LogisticRegression`

and `predict_proba()` helpful)

(e) [5pts] Perform LDA on the training data in order to predict `mpg01` using `cylinders`, `weight`, `displacement`, and `horsepower`. What is the test error of the model obtained? (Hint: You may consider using

`from sklearn.discriminant_analysis import LinearDiscriminantAnalysis`)

Problem 2 [10pts]
We derive 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.

(a) [5pts] What is the probability that the first bootstrap observation is not the first observation from the original sample? Justify your answer. What is the probability that the last bootstrap observation is not the first observation from the original sample? Justify your answer.

(b) [5pts] When $n = 10$, what is the probability that the first observation is not in the bootstrap sample? Likewise, what happens when $n = 100$?

Problems 3-5 are based on the `College` dataset. This dataset has statistics for 777 US Colleges from the 1995 issue of US News and World Report. You can find the description of this dataset at `https://rdrr.io/cran/ISLR/man/College.html`.[1] Let us first create a variable of acceptance rate, `Accept.Rate`, that is the number of applications accepted (`Accept`) divided by the number of applications received (`Apps`). We will now try to predict the acceptance rate using all variables other than `Accept` and `Apps`. We can remove `Accept` and `Apps` from the data frame.

Problem 3 [30pts]

(a) [3pts] Provide an estimate for the population mean of `Accept.Rate`. Call this estimate $\hat{\mu}$. (Hint: You may find `mean()` helpful.)

(b) [3pts] Provide an estimate of the standard error of $\hat{\mu}$. Interpret this result. (Hint: You can compute the standard error of the sample mean by dividing the sample standard deviation by the square root of the number of observations. You may find `std()` and `np.sqrt()` helpful.)

(c) [6pts] Now estimate the standard error of $\hat{\mu}$ using bootstrap resampling for 1,000 times. How does this compare to your answer from (b)?

(d) [3pts] Based on your bootstrap estimate from (c), provide a 95 % confidence interval for the mean of `Accept.Rate`. (Hint: You can approximate a 95% confidence interval using the formula $[\hat{\mu} - 2 \cdot \text{se}(\hat{\mu}), \hat{\mu} + 2 \cdot \text{se}(\hat{\mu})]$)

(e) [6pts] Consider a linear regression model to predict `Accept.Rate` using `Top10perc`. Compute estimates for the standard errors of the intercept β_0 and coefficient β_1 of `Top10perc` using bootstrap. Compare the standard errors to the standard errors reported using the statsmodels package.

(f) [6pts] Consider a KNN regression to predict `Accept.Rate` using `Top10perc` with $K = 10$. Compute the standard error of the predicted `Accept.Rate` when `Top10perc=76` using bootstrap.

(g) [3pts] Repeat the above with $K = 5$ and $K = 50$. Compare the results with linear regression, and comment on your findings.

[1]The dataset can be downloaded here: `https://www.kaggle.com/ishaanv/ISLR-Auto?select=College.csv`.

Problem 4 [20pts]

(a) [6pts] Perform the best subset selection of the one-predictor model. Fit all models that contain exactly one predictor, and report the best one-predictor model based on the largest R^2. Note that the intercept term is always included in the model, but is not counted as a predictor.

(b) [6pts] Perform the best subset selection of the two-predictor model. Fit all models that contain exactly two predictors, and report the best two-predictor model based on the largest R^2.

(c) [6pts] Use forward stepwise selection to select the best two-predictor model. Consider all models that augment the best one-predictor model from (a), and report the best two-predictor model based on the largest R^2. Do you get the same best two-predictor model as (b)?

(d) [2pts] How many models are fitted in (a), (b) and (c)? Comment on your findings.

Problem 5 [20pts]

(a) [0pts] Split the data into a training set and a test set with 80% observations in the training set and 20% observations in the test set. (Hint: You may find

```
from sklearn.model_selection import train_test_split
```

helpful)

(b) [10pts] Fit a ridge regression model on the training set, with λ chosen by cross-validation. Report the test error obtained. [Hint: You may find `sklearn.linear_model.Ridge()` useful.]

(c) [10pts] Fit a lasso regression model on the training set, with λ chosen by cross-validation. Report the test error obtained, along with the number of non-zero coefficient estimates. [Hint: You may find `sklearn.linear_model.Lasso()`.]

(d) [0pts] Compare your answers from (b) and (c). Comment on your findings.