**FINAL EXAM. DSCI35600. SPRING 2023. 11 PROBLEMS.**

**Problem 1. (20 points)** Write **T or F** next to each of the following statements.

1. We use the validation set to compare different models. \_\_\_\_
2. The Ridge algorithm is a regularized version of least squares regression that tries to avoid \_\_\_\_

overfitting by applying a penalty to models based on the size of their coefficients.

1. Increasing the hyper-parameter K in a KNN model will typically make it less \_\_\_\_

likely that the model will overfit.

1. When selecting a value of alpha in a Ridge or LASSO model, we typically choose \_\_\_\_

the value that results in the smallest r-squared score for the training set.

1. It is important to scale features when constructing a ridge regression or a KNN model. \_\_\_\_
2. The output of a logistic regression model is an estimate of the probability that the

provided observation belongs to a particular class. \_\_\_\_

1. The lasso algorithm is a regularized version of least squares regression that tries to avoid

overfitting by applying a penalty to models based on the size of their coefficients. \_\_\_\_

1. At each node the RandomForestClassifier searches for the best feature among

a random subset of features instead of searching thru all features. \_\_\_\_

1. Pasting uses sampling of the training set without replacement. \_\_\_\_
2. The “out-of-bag” training instances are used for validation of their predictor. \_\_\_\_
3. Bagging is the method which uses the same training algorithm for every predictor

and trains them on different random subsets of the training set. \_\_\_\_

1. When comparing potential cuts, a decision tree algorithm selects the cut with the lowest Gini score . \_\_\_\_
2. PCA is a technique for removing the non-relevant features from the original data set . \_\_\_\_

1. When selecting a value for the parameter *C* in a logistic regression model,

we typically choose the value that results in the greatest accuracy for the validation set . \_\_\_\_\_

1. The coefficients in a logistic regression model are trained by minimizing \_\_\_\_

the sum of squared errors objective function.

1. We use negative log-likelihood rather than likelihood when score a logistic regression

model because negative log-likelihood is less affected by rounding issues. \_\_\_\_\_

1. Logistic regression is a regression algorithm (as opposed to a classification algorithm). \_\_\_\_\_
2. When using MinMaxScaler() to scale numerical features, we will create and fit \_\_\_\_\_

separate instances of the scaler for the training, validation, and testing sets.

1. Increasing the hyper-parameter alpha in a LASSO regression model will typically \_\_\_\_\_

make it more likely that the model will overfit.

##### **Problem 2 (10 pts).** Assume that you are provided with a feature array named X and a label array named y. Consider the code below. Provide the output for the last 6 print statements on the lines provided. Provide your answers exactly as they would be displayed in Python.

##### **CODE** **OUTPUT**

print(X.shape) (100, 6)

print(y.shape) (100,)

from sklearn.model\_selection import train\_test\_split

X\_tr, X\_temp, y\_tr, y\_temp = train\_test\_split(X, y, t**est\_size=0.3**, random\_state=1)

print(X\_tr.shape) \_\_\_\_\_\_

print(X\_temp.shape) \_\_\_\_\_\_ \_\_\_\_\_\_

X\_val, X\_test, y\_val, y\_test = train\_test\_split(X\_temp, y\_temp, **test\_size=0.5**, random\_state=1)

print(X\_val.shape) \_\_\_\_\_\_

print(X\_test.shape) \_\_\_\_\_\_

print(y\_val.shape) \_\_\_\_\_\_

**Problem 3. (12 pts)** The **confusion matrix** for a testing set in a classification problem with three classes is provided below. Find the precision and recall for each class, as well as the overall accuracy of the model. **Round to two decimal places.**

|  |  |  |
| --- | --- | --- |
|  | **Precision** | **Recall** |
| **Class 0** |  |  |
| **Class 1** |  |  |
| **Class 2** |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Class 0** | **Class 1** | **Class 2** |
| **Class O** | 40 | 6 | 4 |
| **Class 1** | 3 | 50 | 16 |
| **Class 2** | 7 | 13 | 80 |

**Problem 4 (16 pts).** Assume you are provided with a dataset containing 5 observations and three features to use in creating a regression model. The table below contains information for two different proposed models, with each row relating to a different model. The first four columns provide the proposed coefficient estimates for each model. Columns 5 - 9 contain the **predicted** *y* values resulting from each of the models. The **true** *y* values are provided in the bottom row of the t able.

For both of the proposed models, calculate its loss using **the linear regression loss function,** the **ridge regression loss function with *a* = 2**, and t**he LASSO regression loss function with *a* = 2**. In each case, use SSE rather than MSE as the "baseline" loss. **SHOW YOUR WORK**.

After calculating each of the loss values, circle the best loss for each of the three different loss functions . In other words, for each of the last three columns, circle the best loss value that appears in that specific column .

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  | Linear Regression | Ridge Regression | LASSO Regression |
| 5 | 4 | -3 | 2 | 9 | 11 | 6 | 6 | 5 |  |  |  |
| 6 | 2 | -2 | 2 | 6 | 12 | 5 | 4 | 5 |  |  |  |
| **TRUE y VALUES 🡪** | | | | **7** | **11** | **6** | **5** | **4** |

­­­­­­­­­­

**Problem 5. (14 pts)** A training set for a classification task is provided below. The dataset has two features and one categorical label, *y,* which has two possible classes: " red" and " blue". You are asked to score a logistic regression model for this training data .

**Round all answers to three decimal places on this problem.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 5 | 6 | 2 | 7 |
|  | 6 | 3 | 5 | 2 | 1 |
| *y* | blue | blue | blue | red | red |

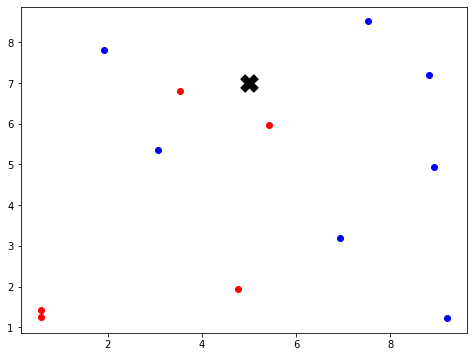
1. In the model below, is an estimate for the probability that an observation falls into the red class. Let for red observations and let for blue observat ions. For the following model, find and for each observation.

)

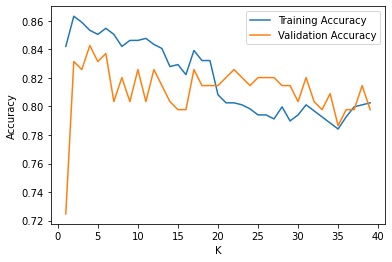
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Calculate the negative log-likelihood score for this model. SHOW THE WORK.

**Problem 6. (10 pts)** A scatter plot consisting of twelve points is shown below. Each point is labelled as one of two classes: "red" or "blue". Assume a KNN classification algorithm is trained on this dataset with *K Nearest Neighbors.* Determine what color is point X depending on the value of  **K=1, 3, or 5.**  A compass might be useful for this problem. SHOW THE DISTANCES ON THE GRAPH



**Problem 7. (6 pts)** The graph below the training and validation accuracy scores of a K-Nearest Neighbors models for varying values of K. Determine the value of K that would be the best to use for this classification task. Briefly explain your answer.



**Problem 8 (16 pts).** Assume that you are provided with three 2D arrays, named X\_train, X\_val, and X\_test.

The contents of these arrays are provided below.

X\_train X\_val X\_test

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 20 |  | 6 | 15 |  | 4 | 12 |
| 5 | 30 |  | 7 | 20 |  | 9 | 10 |
| 7 | 25 |  |  |  |  |  |  |
| 4 | 10 |  |  |  |  |  |  |

Complete the tables below to show the contents of the arrays Xs\_train, Xs\_val, and Xs\_test

after executing the following code:

from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

scaler.fit(X\_train)

Xs\_train = scaler.transform(X\_train)

Xs\_val = scaler.transform(X\_val)

Xs\_test = scaler.transform(X\_test)

X\_train X\_val X\_test

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

**Problem 9 (6 points). Given the Decision Tree model below consider the following points: P=[4, 1], Q=[ 2, 2], and R=[3,2]. W**rite the name of each point P, Q, R beneath the leaf node to which it belongs in the tree diagram below. Also classify the label of each point (0, 1, or 2).

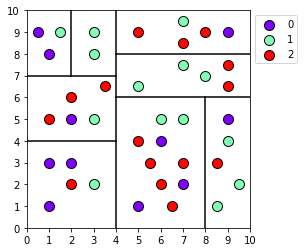
**Diagram

Description automatically generated**

**Problem 10. (12 pts)**  Four probabilistic models are used to create a single ensemble model for a classification task with 4 categorical labels. A single sample is fed into the ensemble, generating the following probability distributions:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Class 0 | Class 1 | Class 2 | Class 3 |
| Model 1 | 0.40 | 0.15 | 0.25 | 0.20 |
| Model 2 | 0.15 | 0.30 | 0.35 | 0.20 |
| Model 3 | 0.10 | 0.15 | 0.25 | 0.50 |
| Model 4 | 0.30 | 0.25 | 0.20 | 0.25 |

1. Assuming a soft-voting scheme is used, find the probability distribution that this model would return for this sample.
2. Which class would the ensemble predict for this sample if soft-voting is used?
3. Which class would the ensemble predict for this sample if hard-voting is used?

**Problem 11. (18 pts)** The plot on the right shows a dataset used to train a decision tree classification model. The horizontal and vertical lines represent show where the algorithm decided to split the dataset at each node.

Translate the information contained in this image to the tree structure provided below. Use 0 to indicate the horizontal axis and 1 to indicate the vertical axis.

Also, calculate the accuracy of this model, as evaluated on the training set displayed in the image. **Round the accuracy to two decimal places.**

**Accuracy =**

**pred:**

**axis:**

**t:**

**pred:**

**pred:**

**pred:**

**pred:**

**pred:**

**pred:**

**pred:**

**axis:**

**t:**

**axis:**

**t:**

**axis:**

**t:**

**axis:**

**t:**

**axis:**

**t:**

**axis:**

**t:**