COSC 425: Intro to Machine Learning
Lab 4: Linear Regression and SVM

Due: October 28, 2022, 11:59 PM

Introduction

In this lab, you will be practicing using linear regression, SVMs, and applying them to datasets. Here, our dataset for linear regression will be Auto MPG dataset from the UCI Machine Learning repository: `https://archive.ics.uci.edu/ml/datasets/auto+mpg`. The actual CSV for the dataset you will be using is available on Canvas for download. Our dataset for SVM will be the Star Type Classification from NASA, available here and on Canvas: `https://www.kaggle.com/datasets/brsdincer/star-type-classification`

You will create a Jupyter notebook for each part that you will submit where you will put the code that you wrote for this lab. You will also submit a separate written report in PDF form in which you include the answers to the questions and/or plots required by the question.

Note: For all plots, you should include axes labels and titles.

Part 1: Linear Regression

Read in the auto-mpg.csv dataset using pandas and create the y label vector that we will use throughout, which will be the MPG column. The features we will be using for regression are the columns cylinders, displacement, horsepower, weight, and acceleration. You will ignore the model-year, origin, and car-name columns. Anywhere you are asked to create a train-test split on the data, you should use a fixed random state throughout (noted in your write-up) and a test_size=0.2.

Question 1.1 (60 points – 12 points for each feature): For each of the five features (cylinders, displacement, horsepower, weight, and acceleration), do the following:

- Create an X matrix where each row is the current feature of interest.
- Do the train test split as indicated above. Create a plot that shows the current feature on the x-axis and the MPG on the y-axis for the training data.
- In the report, note whether you expect linear regression will perform well using that feature to predict the MPG value and why you believe it will perform well or not perform well.
- Use sklearn's LinearRegression() to fit to the training data.

- Create a prediction vector based on the training data, and calculate and print the mean-squared error and the R^2 score using sklearn on the training set.
- Create a prediction vector based on the testing data, and calculate and print the mean-squared error and the R^2 score using sklearn on the testing set.
- Print the coefficient and intercept parameters from the sklearn model for that feature.
- Plot the line created from that linear regression along with points for the training data (as one color) and the testing data (as another color). Include a legend to denote which are training and which are testing. Note that the x-axis should be the feature you're predicting on and the y-axis should be MPG.

In the report, note which of the features you believe is best for predicting MPG and why you selected that feature.

Question 1.2 (10 points): Combining all of the data, do the following:

- Create an X matrix that includes all five features.
- Do the train test split as indicated above.
- Use sklearn's LinearRegression() to fit to the training data.
- Create a prediction vector based on the training data, and calculate and print the mean-squared error and the R^2 score using sklearn.
- Create a prediction vector based on the testing data, and calculate and print the mean-squared error and the R^2 score using sklearn.

Does using all of the data improve performance over using each of the features individually?

Part 2: SVM

Read in the Stars.csv dataset using pandas and create the y label vector, which will be the Type column. The features we will be using for prediction are the columns Temperature, L, R, and A_M. Anywhere you are asked to create a train-test split on the data, you should use a fixed random state throughout (noted in your write-up) and a test_size=0.2. Use a fixed random_state for the LinearSVC classifier throughout.

Question 2.1 (25 points): Finding hyperparameters:

- Create a validation set and subtraining set from the training set (a single validation set, rather than KFolds) with 0.125 of the training set serving as the validation set.

- Fit the LinearSVC classifier on the training set and evaluate on the validation set for all combinations of max_iter=[1000, 10000, 100000, 1000000] and C=[0.01, 0.1, 1, 10, 100, 1000].
- Print the validation accuracy for each combination and show a heatmap comparing of the validation accuracies across the combinations (include a colorbar).
- Note the best parameter combination and discuss why those parameter values might be the best for this dataset (1-3 sentences).

Question 2.2 (5 points): Use the maximum iterations and C values you found to perform best to inform how you create your LinearSVC() and fit on the whole training set. Compute and print the training and testing accuracy for the Stars dataset.

Submission Checklist

In this lab, you will submit on Canvas:

- Jupyter notebook for part 1
- Jupyter notebook for part 2
- PDF of your report

Late Penalty: 20 points off per day late.