COSC 425: Intro to Machine Learning
Lab 2: K-Nearest Neighbors

Due: September 23, 2022, 11:59 PM

Introduction

In this lab, you will be practicing using K-nearest neighbors and applying it to a dataset. Here, the data will be statistics about the Olympic athletes from the 2016 Rio Olympic games (available here: `https://www.kaggle.com/datasets/rio2016/olympic-games`).

You will create a Jupyter notebook 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.

Questions

Question 1 (15 points): Read in the data from the given CSV file. Down-select to create a dataframe that only includes three columns (sport, height, and weight) for all of the males who competed in either basketball or cycling, then perform dropna() to remove any missing rows. This should result in a dataframe with 3 columns and 450 rows. Convert the non-numerical feature (sport) to numerical values. Further, convert the height values from meters to centimeters. Include in your report a brief description (1-2 sentences) of why it might be a good idea to use the data in centimeters rather than meters for k-nearest neighbors.

Question 2 (15 points): For each class (basketball, cycling) and feature (height, weight), compute the following statistics: minimum, maximum, mean, standard deviation, and signal-to-noise ratio (SNR = mean/standard deviation). Create a table that shows the statistics for each class and feature. Do any of the statistics give rise to concern? Provide a brief description (1-2 sentences) in the report.

Question 3 (10 points): Create a scatter plot of the data, where you color the basketball data and cycling data differently. Make sure to label your axes and include a legend for the labels. Does the scatter plot indicate potential success or failure with respect to separating the two classes using KNN? Why? Provide a brief description (1-2 sentences) in the report.

Question 4 (5 points): Divide the data into training and test data using the standard 80-20 ratio. Apply 10-fold cross-validation to the training data.

Question 5 (25 points): Create and iterate over different numbers of neighbors from 1 to 21 (inclusive, but odd numbers only). For each hyperparameter, calculate the mean and standard deviation of classification accuracy across the different folds of the data. Create a table for your report showing the mean and standard deviation for each hyperparameter. Select

which hyperparameter to use based on the mean classification accuracy on the validation data. Note which hyperparameter value is selected in the report.

Question 6 (10 points):): Build a new K-nearest neighbor classifier using all your training data with the hyperparameter defined in Question 5. Use the resulting model to classify the test data (which haven't used until now). Calculate and report overall training and test data performance.

Question 7 (20 points): Create a 2D plot that visualizes the classifier's performance (a decision boundary). Run a mesh of data through the classifier to determine basketball and cycling decision regions. Color them two different light colors (alpha=0.2) that are easy to visually separate. Then, overlay the test data using two different colors for basketball and cycling. Make sure all plots are labeled and that a legend is included.

Helpful Hints

Use pandas.read_csv to read the input file. Use numpy.min(), etc., to calculate the data statistics.

Use plt.scatter() or seaborn.scatterplot() to produce raw data visualization and the classifier test data overlay.

Use numpy.meshgrid and matplotlib.pyplot.contourf (as we did in class) to produce the classifier visualization (the decision boundary).

Use sklearn.model_selection.train_test_split and KFold to split up the data. Use sklearn.metrics to determine classification accuracy and sklearn.neighbors.KNeighborsClassifier to produce the K-nearest neighbor model.

Finally, use the reference manuals for all of the above.

Submission Checklist

In this lab, you will submit on Canvas:

- Jupyter notebook
- PDF of your report

Late Penalty: 20 points off per day late.