1. **Decision Trees as Interpretable Models**

   (a) Download the Accute Inflamations data from `https://archive.ics.uci.edu/ml/datasets/Acute+Inflammations`. (5 pts)

   (b) Build a decision tree on the whole data set and plot it.[1] (5 pts)

   (c) Convert the decision rules into a set of IF-THEN rules.[2] (10 pts)

   (d) (10 points extra credit): Use cost-complexity pruning to find a minimal decision tree and a set of decision rules with high interpretability.

2. **Random Forests, Ridge Regularized, and LASSO Regularized Regression**

   (a) Download the Communities and Crime data[3] from `https://archive.ics.uci.edu/ml/datasets/Communities+and+Crime`. Use the first 1495 rows of data as the training set and the rest as the test set. (5 pts)

   (b) The data set has missing values. Use a simple data imputation technique (using the mean or median statistic of each feature) to deal with the missing values in the data set. The data description mentions that five features are nonpredictive. Ignore those features. (5 pts)

   (c) Plot a correlation matrix for the features in the data set. (5 pts)

   (d) Calculate the Coefficient of Variation CV for each feature, where $CV = \frac{s}{m}$, in which s is sample standard deviation and m is sample mean. (5 pts)

   (e) Pick $\lfloor\sqrt{122}\rfloor$ features with highest CV, and make scatter plots and box plots for them. Can you draw conclusions about significance of those features, just by the scatter plots? (10 pts)

   (f) Fit a linear model using least squares to the training set and report the test error. (10 pts)

   (g) Fit a ridge regression model on the training set, with λ chosen by 5-fold cross-validation. Report the test error obtained. (10 pts)

   (h) Fit a LASSO model on the training set, with λ chosen by 5-fold cross-validation. Report the test error obtained, along with a list of the variables selected by the model. Repeat with standardized[4] features. Report the test error for both cases and compare them. (10 pts)

[1] This data set is a multi-label data set. Sk-Learn seems to support building multi-label decision trees. Alternatively, you can use the label powerset method to convert it to a multiclass data set. Also, you can use the binary relevance method and build one decision tree for each label. It seems that the label powerset approach is more relevant here. Is that right?

[2] You can use the code in `https://www.kdnuggets.com/2017/05/simplifying-decision-tree-interpretation-decision-rules-python.html`.

[3] Question you may encounter: I tried opening the dataset and download it but the file is not readable. How to download the file? Just change .data to .csv.

[4] In this data set, features are already normalized.

(i) Fit random forest models to the training set using $m = \lfloor\sqrt{123}\rfloor$ and $m = 122$ and plot the train, out of bag (OOB), and test errors for $B \in \{1, 2, \ldots, 300\}$ on the same plot and compare them.[5] (10 pts)

(j) Build a variable importance plot (see p. 319 of ISLR) from your random forest. If the variable importance plot is hard to read, only keep the top 10 most important features. (10 pts)

(k) (5 points extra credit) Use an iterative data imputation technique and repeat the above.

(l) (20 points extra credit) Repeat 2i and 2j using Extra Trees. Use bootstrap subsamples whose size is 20% of the dataset.

[5]See p. 318 of ISLR.