COMP132, Assignment 6

- Please enter all your answers in a single file called `Assignment6.ipynb` (the only acceptable format).
- After uploading your file to the assessment system, press the ”Run checks” button. It will perform a basic sanity check. Make sure that there are no errors.

This assignment has three questions and you are required to do Question 1, and either Question 2 or Question 3. This assignment will be marked out of 100.

The first two questions practice some basic aspects of machine learning using Python. Briefly, machine learning is the semi-automated extraction of knowledge from data. In particular, machine learning always starts with data, and our goal is to extract knowledge or insight from that data. We have a question in mind and we hypothesize that our question is answerable using the data. Next, in an automated process, we use the computer to gain the insight by applying some processes and algorithms to the data. However, machine learning is not a fully automated process and a successful machine learning task requires a lot of smart decisions by human.

For this assignment we focus on **supervised learning** which is one of the main categories in machine learning. Supervised learning is the process of making predictions using data. Here, there is an outcome we are trying to predict. The primary goal of supervised learning is to build a model that *”generalizes”*, meaning that it accurately predicts the future examples based on learning from the past examples. Therefore, supervised learning has two main steps:

1. **Model training**: Train a *machine learning model* using the existing *labeled data*. Labeled data is the data that has been labeled with the outcome. In this process the model is learning the relationship between the attributes of the data and its outcome.

2. **Prediction**: Make predictions on new data for which the label is unknown using the trained machine learning model.

For further reading on machine learning, please see the following book:
An Introduction to Statistical Learning (section 2.1, 14 pages). The PDF file of the book is available in the Assignment 6 page and was downloaded from
https://www.stat.berkeley.edu/~rabbee/s154/ISLR_First_Printing.pdf

Question 1 (50 pts)

For the rest of this question, look at the *Advertising.csv* dataset, and get yourself familiar with it. The data set was downloaded from:

https://github.com/Columbia-Intro-Data-Science/python-introduction-caitlinwang/blob/master/www-bcf.usc.edu/~gareth/ISL/Advertising.csv

This DataFrame shows the sale of a company based on its advertising on TV, radio, and newspaper. In this question, we will be using ’TV’, ’Radio’, and ’Newspaper’ columns as features to predict the ’Sales’ column.

(a) Set the ’Unnamed: 0’ column as the index column. [3 pts]

(b) Visualize each feature column (TV, Radio, and Newspaper) versus the Sales in three different plots in one figure. Set the features as the x axis and the sales as y axis information. Use scatter to create three plots in the same figure using different colors and labels. Show these colors/labels in the figure legend. Can you observe any linear relationship between any of the features and the Sales? Which feature has the most linear behavior in this manner? Which feature has the least linear behavior? [3 pts]

(c) Subset a feature DataFrame from the original DataFrame that contains TV, Radio, and Newspaper columns, and put it into variable X. Also, subset the outcome Series (the “Sales” column) from the original DataFrame, and put it into variable y. [3 pts]

(d) Import the traintestsplit function from the sklearn package as shown below to split X and y objects into two objects each: test and train. Use all the default values. [3 pts]

From sklearn.modelselection import traintestsplit

(e) Import the **LinearRegression** function from the sklearn package as shown below and instantiate the function assuming the default values for its parameters. [3 pts]

From sklearn.linearmodel import LinearRegression

(f) Use the fit function and training objects from *x* and *y* to fit the model to the training data. [3 pts]

(g) Print the intercept and coefficients values. Then print each pair of feature name and its coefficient. [3 pts]

(h) Using the predict function and the test object of X, make predictions on the testing set. Show a scatter plot for y-test value and the predicted values. Do you see any linear behavior between y-test and the y-predicted? Give an interpretation of the plot. [3 pts]

(i) To evaluate your regression model, import metrics from sklearn as shown below. Next, use **meanabsoluteerror** and **meansquarederror** to calculate and show Mean Absolute Error (MAE) and Mean Squared Error (MSE). Use the Internet to find out the specification of Root Mean Squared Error (RMSE) and calculate it too. [6 pts]

```
From sklearn import metrics
```

(j) Select TV and Radio features from X and put it in X2. Repeat steps (e)-(i) using a new instance (object) of the **LinearRegression** function. Can you observe any improvement in terms of RMSE? Explain the reason. [10 pts]

(k) Select TV feature from X and put it in X3. Repeat steps (e)-(i) using a new instance (object) of the **LinearRegression** function. (Note that even we only have one feature, you still need to use a list to represent the feature set to put the data into the right format.) Can you observe any decline in performance in terms of RMSE in comparison to the two previous cases? Explain the reason. [10 pts]

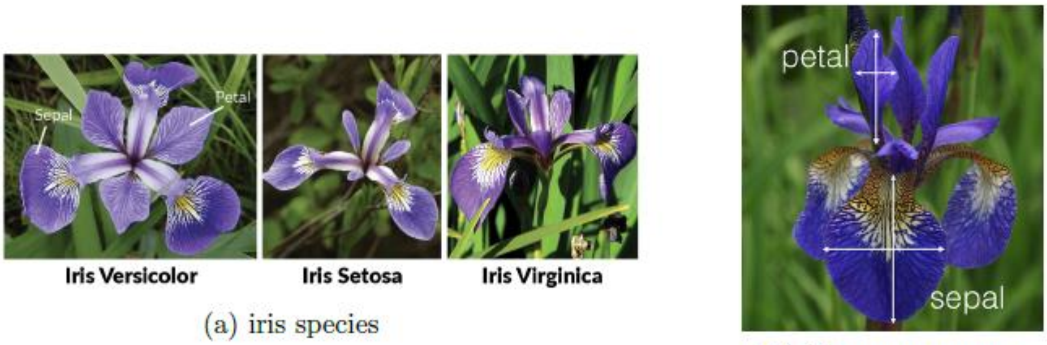
Option A: Question 2 (50 pts)

In this question, you will be practicing another useful concept in machine learning known as ***classification***. As the name indicates, classification is the problem of identifying to which of a set of categories (classes) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known.

The classification algorithm we use in this assignment is the K-Nearest Neighbors (KNN) algorithm, and the dataset that we use is the famous *iris* dataset. The iris dataset is a collection of 150 samples from three different species of iris flower (each specie has 50 samples). Each datapoint is a set of four numbers followed by the iris specie's name. The numbers are accordingly sepal length, sepal width, petal length, and petal width from an iris sample. Another equivalent terminology for sample is *observation* and another equivalent terminology for measurement is *feature*. The three iris species are setosa, versicolor, and verginica which are also called *outcome*. Fig. 1(a) shows a sample flower from each specie, and Fig. 1(b) shows how the measurements have been carried out.

Since there is a strong relationship between the measurements and the species in the iris dataset, the three species can be accurately distinguished from one another utilizing a proper machine learning technique and the measurements. Given a supervised learning scenario, the goal is to predict the species of a given iris using the available data. In other words, we attempt to learn the relation between the data (iris measurements) and the outcome which is the species of iris.

More information about the dataset can be found here:
https://archive.ics.uci.edu/dataset/53/iris



(a) iris species

(b) iris measurements

Figure 1: The iris dataset

The iris dataset is so popular that it has been built into scikit-learn. To import the dataset, first you need to import the loadiris function: from sklearn.datasets import loadiris

Next, load the dataset using the loadiris function into an arbitrary variable: iris = loadiris()

(a) The iris dataset has several attributes that reveals different specifications from it. Run the following attributes, see the results and then explain what they are: [3 pts]

data, featurenames, target, targetnames

In this question, you use KNN for a classification task on the iris dataset. First, you should use a portion of the iris dataset to train the KNN classification algorithm, and then use the other portion to test KNN in predicting the correct specie. Next, you should measure the classification accuracy of KNN using built-in functions. Also, to see the effect of changes in the number of nearest neighbor parameter (K), you should find the optimal value of K for which the highest classification accuracy is acquirable.

Now, import the KNN class as below:
from sklearn.neighbors import KNeighborsClassifier

(b) Create an instance (estimator object) of the KNN class in which $K = 1$ and save it in an arbitrary variable. [3 pts]

(c) Print the instance to see other parameters with their default values. Remember that scikit-learn provides sensible values for these parameters so that you can get started without worrying about them. [3 pts]

(d) Split the dataset (data and target) to a train and a test dataset using the traintestsplit function. Set the test-portion size to be 0.4 of the whole dataset. [3 pts]

(e) Train the created KNN instance with the train-portion of the data and target using the fit function. [3 pts]

(f) Predict the outcome of the test-portion of the data using the predict function. [3 pts]

To test the classification accuracy of our KNN object, we will be using the metrics module from scikit-learn. Import the module:

```
from sklearn import metrics
```

The function we use to measure the classification accuracy is accuracyscore from the metrics module.

The arguments for the accuracyscore function are the true response values and the predicted response values.

(g) Measure the classification accuracy for the KNN object using accuracyscore function and print the result. [3 pts]

(h) Create a new KNN object with $K = 20$ and repeat steps (e) through (g). Which case has a better classification accuracy, KNN with $K = 1$ or KNN with $K = 20$? [14 pts]

(i) Test the classification accuracy of KNN algorithm on the iris dataset with K ranging from 1 to 30 and the previous train and test data. Store the classification accuracy in a variable and plot it across K (y axis is classification accuracy and x axis is K). Report the K value(s) for which KNN has the highest and lowest accuracy (**hint:** use loops). [15 pts]

Option B: Question 3 (50 pts)

This question is to learn the fundamental features of BioPython. Biopython is a set of freely available tools for biological computation written in Python by an international team of developers. Questions in this homework are designed based on its tutorial available on:
https://biopython.org/docs/latest/Tutorial/index.html

3.1 This question is to show you basic features of the Bio.SeqIO package in BioPython, and its partner package SeqRecord. [18 pts, 3 pts for each sub-question]

(a) Read the file *ls_orchid.fasta*, and save the output as records1. Print the type of records1, and then convert the type of records1 into list and print the size of records1.

(b) Read the file *ls_orchid.gbk*, and save the output as records2. Print the type of records2, and then convert the type of records2 into list and print the size of records2.

(c) Print the id and the length of each record in records1.

(d) Print the id and the length of each record in records2.

(e) Find the records with the ids of
gi|2765573|emb|Z78448.1|PAZ78448,
gi|2765612|emb|Z78487.1|PHZ78487 and
gi|2765623|emb|Z78498.1|PMZ78498 from records1, and then print their lengths and indexes in records1.

(f) Find the records with the ids of
Z78504.1,
Z78497.1 and
Z78476.1 from records2, and then print their lengths and indexes in records2.

3.2 This question aims to learn the details of the Seq package. [18 pts, 2 points for sub-questions (a) to (f) and 3 pts sub-questions (g) and (h)]

(a) Read the first record in *ls_orchid.fasta*, and save its seq property as seq.

(b) Print the first 10 and the last 10 letters in seq and their index.

(c) Count the numbers of AT, TA, CG and GC in seq.

(d) Count the proportions of A, T, G and C in seq.

(e) Print the lower and capital cases of seq.

(f) Print the complement of seq.

(g) Print the reverse complement of seq.

(h) Create a sequence with a total length of 20, consisting of the equal amount of A, T, G and C.

3.3 This question involves knowledge about DNA, RNA, m-RNA proteins and the transcription as well as translation. [14 pts, 2 pts for (a) and 3 pts for the each of the other sub-question]

(a) Create a sequence of DNA, RNA and Proteins using AGTACACTGGT.

(b) Create a DNA sequence of GATCGATGGGCCTATATAGGATCGAAAATCGC, and print its first half and the second half.

(c) Create a DNA sequence of GATCGATGC, and another DNA sequence of ACGT. Connect them and then print the new sequence.

(d) Giving a template strand sequence TACCGGTAACATTACCCGGCGACTTTCCCACGGGCTATC, find its transcription

(e) Translate the RAN saved in *rosalind_prot.txt* into proteins based on the standard password table.