Homework 2

Problem 1

Researchers in Canada enrolled 75 patients in a study to see if vaccination for COVID-19 helped alleviate symptoms of long COVID, also known as post-COVID19 conditions or PCC [1]. Of the 75 subjects, 36 were vaccinated, and 39 remained unvaccinated. After a period of two months, the subjects returned and took a survey reporting whether their well-being had improved, worsened, or remained unchanged and whether they had more, less, or the same number of PCC symptoms. The data are in the "PCC_study_train.csv" file.

(a) Calculate the probability that a subject reports their well-being score has improved and that they have fewer PCC symptoms, given that they were vaccinated.

(b) Calculate the probability that a subject reports their well-being score has improved and that they have fewer PCC symptoms, given that they were unvaccinated.

(c) Create two tables with the conditional probabilities for vaccinated and unvaccinated patients.

(d) *Assuming the variables are independent*, write a function that returns the likelihood of a certain outcome/set of symptoms occurring, given a table of conditional probabilities for that class and the probability of that class in the dataset (a.k.a., $P(\text{outcome}|\text{vaccination status})$, e.g., $P(\text{WB score "Improved", PCC symptoms "Same"}|\text{vaccination status}))$

(e) Using your function from the previous question, predict the likelihoods that a patient is vaccinated and unvaccinated, given that their well-being score decreased but the number of post-covid conditions stayed the same.

(f) A new dataset containing data for 60 additional patients is stored in the "PCC_study_test.csv" file. Using your function, predict whether each patient was vaccinated or not vaccinated.

(g) Now use the `sklearn.naive_bayes.CategoricalNB` to train the classifier. Confirm that your predictions are the same as the manual classifier–*they should be identical!* If they are not, you have made a mistake somewhere in the previous questions.

Problem 2

A famous machine learning data set is a collection of measurements on 150 irises, spread over three species (*Iris setosa*, *versicolor*, and *virginica*). The goal of this data set is to predict the species of an iris given the measurements.

(a) Load the data. Then, split the data into a training set with 80% of the observations and 20% of the data for a test set.

(b) Consider first the variable petal length–using the training set, create a plot with the empirical density (i.e., from the data) of petal length for each of the three species. Comment on the distributions.

(c) Using the training set, calculate the means and standard deviations of petal length for each species.

[1] Nayyerabadi, Maryam, Lyvia Fourcade, Swarali A. Joshi, Prabha Chandrasekaran, Arpita Chakravarti, Chantal Massé, Marie-Lorna Paul et al. "Vaccination after developing long COVID: Impact on clinical presentation, viral persistence, and immune responses." International Journal of Infectious Diseases 136 (2023): 136-145.

(d) The command for calculating the normal density function from the `scipy.stats` module is `scipy.stats.norm.pdf`. Using this function, calculate the likelihood that an iris with a petal length of 5.12 is `iris setosa`, `iris versicolor`, or `iris virginica`. Which species would you classify this iris as?

(e) Now consider all four measurements. Create three additional empirical density plots, so that you have the densities for every combination of variable and species.

(f) Using the training data, implement a Gaussian Naive Bayes Classifier. Given an iris with a sepal length of 5.42, sepal width of 3.81, petal length of 4.23, and petal width of 2.15, which species would you classify it as?

Problem 3

(a) Using your Gaussian Naive Bayes Classifier from Problem 2, predict the species of each iris in the test set.

(b) Give a confusion matrix for the test set.

(c) What is the accuracy of your classifier?

(d) What is the most commonly made error in your classifier? Is this consistent with the plots that you made in the previous questions?

Problem 4

The Kaggle "SMS Spam Collection Dataset" is a collection of 5000 text messages, 13% of which are "spam". Call the remaining 87% "ham".

(a) Load the .csv file into Python. Specify `encoding = "latin-1"`. Confirm that there are 5572 rows and 2 columns. (You may need to do a bit of cleaning to remove unnecessary columns.)

(b) The first column contains a label describing whether the text message is ham or spam. Split the big data frame into two smaller data frames–one for spam, and one for ham. How many messages are in each data frame?

(c) The second column contains the text messages. We need to clean the text messages for the actual data analysis, and it is easiest to do this by creating one long string of all of the messages for each data frame. Create a long string for spam and a string for ham using the `.str()` and `.cat()` methods. Then, split the long strings so that it is a list of shorter word strings, also known as "tokens".

(d) Now, we need to count the frequency of each word in every text message for both the ham and spam data frames. Luckily, there is a function that will do that for us! Install and import the `nltk`, or **N**atural **L**anguage **T**ool **K**it, module. Then, apply the `.FreqDist()` method to both strings.

(e) The output of `.FreqDist()` behaves like a dictionary. How many tokens are in each dictionary? What is the probability of the word "Sorry" occurring in the ham dictionary? In the spam dictionary?

(f) Write a function that, given a text message and a dictionary, will calculate the probability of the message occurring.

(g) What is the probability of the message "Sorry my roommates took forever, it ok if I come by now?" occurring in the ham dataset? In the spam dataset?

(h) If we want to compare probabilities, those that are equal to zero may present problems. Additionally, the probabilities for very long text messages will be very small, and therefore may be stored as zero in Python–which will cause issues later. Adapt your function to include some sort of penalty for words that occur in either dictionary zero times, and convert the function to return the log odds of the word occuring. Clearly document your choices for your penalty.

(i) Now, construct a function that "scores" each message by calculating the log odds ratio for text message being ham to spam.

(j) What is the score of the message "Sorry my roommates took forever, it ok if I come by now?"

(k) Apply the scoring function to every message 50 or fewer tokens long in the data set. Create a plot displaying the distributions of the scores for ham messages and spam messages, and describe the differences in the distributions.

Problem 5

Consider a scenario where you have a filter for spam text messages–if a text message is predicted to be spam, it is sent to a different folder. In this case, the "spam" label should be considered a "positive".

(a) Using the scores from the previous problem, assign all of the messages that have scores less than 1 to be predicted as spam and those that have scores less than one to be predicted as ham. Give the confusion matrix for your classifier, as well as the accuracy, sensitivity, and specificity.

(b) Then, change the penalty you used for words/tokens with zero frequency. Try at least two different penalties–what are the effects of changing the penalty on accuracy? Sensitivity? Specificity?