## 1. Introduction

This Naive Bayes classifier is inspired by the question 3, homework 4. The Naive Bayes classifiers can be used to classify categorical data. The goal of Naive Bayes Classifier is to calculate conditional probability:

$$p(C_k | x_1, x_2, \ldots, x_n)$$

The Naive Bayes Classifier assign features and classes $C_k$ given observed training data $x[n] = (x_1[n], x_2[n], \ldots, x_m[n])$. Bayes Rule is shown below.

$$p(C_k | x[n]) = \frac{p(x[n] | C_k)p(C_k)}{p(x[n])}$$

The assumption of Naive Bayes is that the element and features are mutually independent

$$p(x[n] | C_K) = \prod_{i=1}^{N} p(x_i | C_k)$$

The conditional probability can be calculated as followed.

$$p(C_k | x[n]) \propto p(C_k, x_1, x_2, \ldots, x_n) = p(C_k) \sum_{i=1}^{n} p(x_i | C_K)$$

## 2. Method

The maximum likelihood estimation is used to estimate parameters, such as prior probability and conditional probability

$$p(C_k) = p(y = C_k) = \frac{\sum_{i=1}^{N} I(y_i = C_k)}{N}$$

One of the parameters, the prior probability,  can be calculated as followed

$$p(x_1 = a_j | y = C_k) = \frac{\sum_{i=1}^{N} I(x_i = a_j, y_i = C_k)}{\sum_{i=1}^{N} I(y_i = C_k)}$$

The Naive Bayes Classifier can calculate the probability to make a prediction:

$$y = argmaxp(y = C_k) \prod p(x | y = C_k)$$

When Naive Bayes Classifier is used to calculate posterior probability and make prediction on categorical data, some of features is not shown in training data, which results in the problem of zero probability. Laplace smoothing can be used to solve this problem.

$$p(C_k) = p(y = C_k) = \frac{\sum_{i=1}^{N} I(y_i = C_k) + \lambda}{N + K\lambda}$$

$$p(x_i = a_j | y = C_k) = \frac{\sum_{i=1}^{N} I(x_i = a_j, y_i = C_k) + \lambda}{\sum_{i=1}^{N} I(y_i = C_k) + A\lambda}$$

Where K denotes the number of different values in label $y$ and $A$ denotes the number of different values in features $a_j$. Normally the value of $\lambda$ is equals to 1, so I applied $\lambda = 1$ in my project.

## 3. Application

The training data that I used is entirely from Problem 3.

| x | | | | | $C_k$ |
|---|---|---|---|---|---|
| Weekday | Weather | Time | Actions | Temperature | Caught by Police |
| Yes | Rain | Afternoon | Stop Sign Violation | Low | No |
| No | Sunny | Midnight | Speeding | High | No |
| No | Rain | Afternoon | Speeding | Low | No |
| Yes | Sunny | Morning | Speeding | High | No |
| No | Sunny | Midnight | Stop Sign Violation | Low | Yes |
| Yes | Rain | Afternoon | Stop Sign Violation | High | Yes |

There are two function can be used: the first one is *predictor*, another one is *predictor_smoothing.*

When these two functions are used to make predictions, input values are needed. Functions can be used as followed.

**predictor('No','Sunny','Morning','Stop Sign Violation','Low')**
**predictor_smoothing('No','Sunny','Afternoon','Speeding','Low')**

One of the advantage of Naive Bayes is that Naive Bayes Classifier can make prediction based on the unobserved data. To be specific, the classifier can make predictions based on five features, Weekday, Weather, Time, Actions and Temperature. If some of the features are not observed, or values of some features is null, Naive Bayes Classifier can also make prediction. When users input values for features, and if this value is not observed, users can input "Null". Two functions can be used as followed.

**predictor('No','Sunny','Afternoon','Stop Sign Violation','Null')**
**predictor_smoothing('Yes','Rain','Null','Stop Sign Violation','Null')**

## 4. Reference

[1] Homework 4, problem 3, Detection and Estimation Theory, 2020 Spring
[2] https://en.wikipedia.org/wiki/Naive_Bayes_classifier