**PROJECT 2**

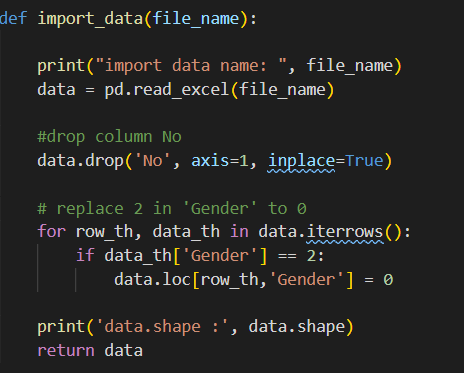
**AI 2020 Naïve Bayesian Project**

**Le Van Hung (黎文雄)**

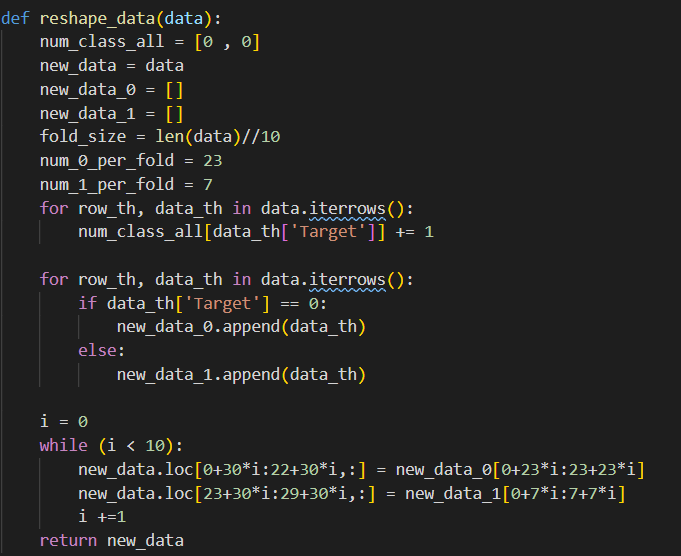
**Student ID: 0860831**

1. **Data Preprocessing**

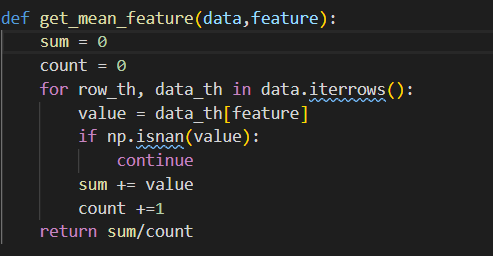
Reading data by using pandas.read\_excel(data). Because “No” is not one of the features, so drop column “No”, after that replace value 2 in column “Gender” by value 0.



With training set, we have 300 data points totally. The original class distribution is 0/1 = 23/7. I will perform Stratified 10-fold Cross validation, in which each fold follows the ratio 23/7. So, in each fold, we have 23 data with class 0 and 7 data with class 1. Function reshape\_data(data) will do this job.



Because this data set include NaN value only in the numerical feature, so I will replace this NaN value with the mean of this feature by using function get\_mean\_feature(data, feature)



1. **Formula**
2. **Introduction Naïve Bayes’ Theorem**

In this project, we use Naïve Bayes’ Theorem, in simple terms, this theorem will assume that independence among the features in class.

Navie Bayes’ Theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c) like this equation below:



**Where:**

* P(c|x) is the posterior probability of class (c is target) given predictor (x is attributes or features)
* P(c) is the prior probability of class
* P(x|c) is the likelihood which is the probability of predictor given class. We can easy to calculate the likelihood if we suppose that each feature in class is independent.
* P(x) is the prior probability of predictor.

1. **Apply to project**

In this project, class c (presented at column ‘Target’) can receive 2 value: 0 or 1. We have 36 features totally (19 categorical features and 17 numerical features). So, the likelihood P(x|c) can present as:



With each data in testing set, we need to calculate P(x|c) and P(c) at c =0 and c=1 and compare the result. The predict class can be received by:



Because, we have lagre features (36 features), so to avoid the error, I use log function to get the prediction.



* **Calculate P(c)**



* **Calculate P(xi|c)**

If xi is value of numerical feature, I use **Gaussian Naive Bayes** to calculate this probability.

With **i**th-data xi and class c ( 0 or 1), xi will base on **gaussian distribution** with mean and variance . P(xi|c) can be calculated by the below equation:



**Where:**



If xi is value of categorical feature, I use **Multinomial Naïve Bayes** to calculate this probability. I use min\_category variable as **minimum of number of xi in class c** to avoid zero when using log function, fortunately, no zero value appears in this project.

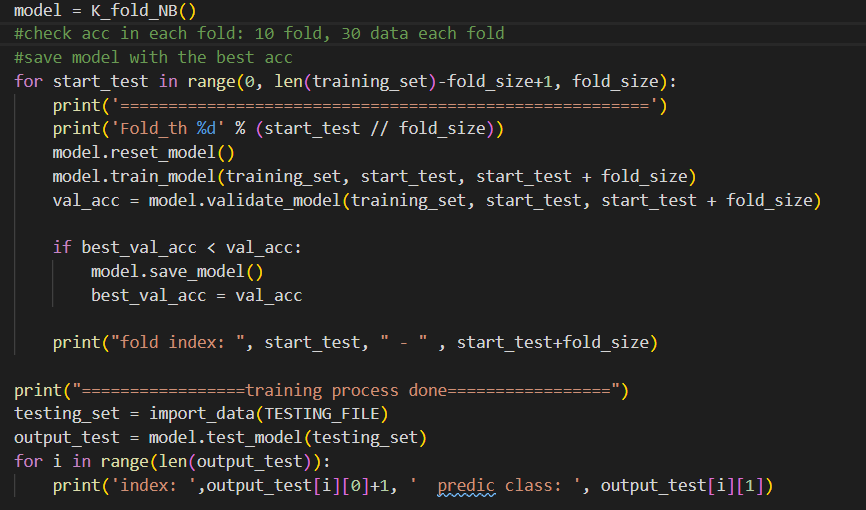


1. **Verification method**

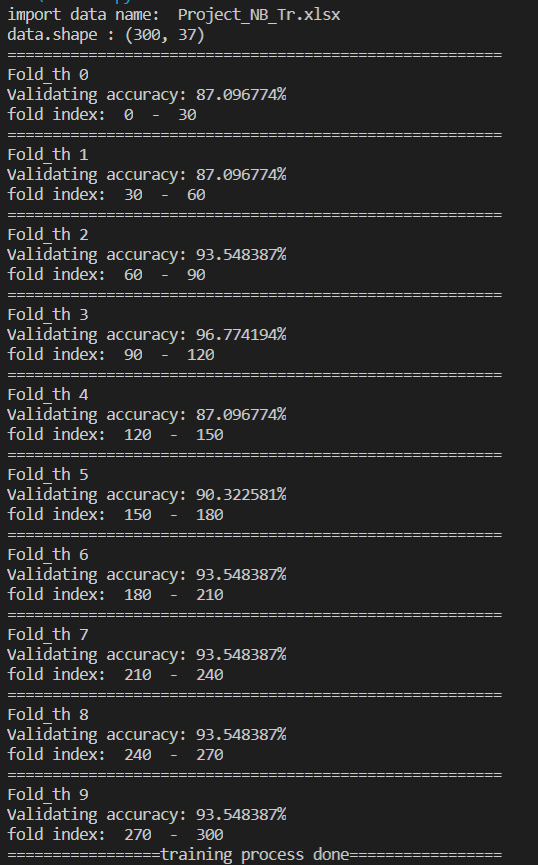
With training set, we have 300 data points totally. The original class distribution is 0/1 = 23/7. To validate the training set, I perform Stratified 10-fold Cross validation, in which each fold follows the ratio 23/7. So, in each fold, we have 23 data will class 0 and 7 data with class 1.

In each time for training (9 folds for training and 1 fold for validation), Call function **model.reset\_model()** to reset model and **model.train\_model()** to calculate mean, variance. After that, call function **model.validate\_model()** to estimate the accuracy in this validation set. If the accuracy is higher than last time training, call function **model.save\_model()** to save this training and testing set.

Do this process in 10-fold CV, we will find the best model and use this for testing.



**Result of training and validation process**



1. **How to run code**

**My Environment to run:**

* Window 10 64 bit
* IDE: Visual Studio Code (VS Code)
* Python 3.7.8 with numpy 1.19.0, pandas 1.0.1 and xlrd 1.2.0 (only use pandas for read file)

**Run code:**

* 1st step: copy file **Project\_NB\_Tr.xlsx** and **Project\_NB Ts.xlsx** to the same directory with 0860831.py file.
* python 0860831.py or Ctrl+F5 to run
* F5 to debug

*P/S: If you do have any problem with my report, don’t hesitate to contact me via* ***hungle0804@gmail.com***