MH8111 Assignment 1

Classification using Naïve Bayes

by Li Fengzhi (G1901809H)

Content Structure

- Source Code and Dataset Directory
- SMS Classification with Naïve Bayes (Book: Chapter 4)
- UCI Adult Dataset Classification with Naïve Bayes (My Own)

Source Code and Dataset Directory

- Two .R files
  - MH8111_ExampleFromBook.R
    - I followed the Machine Learning with R's Chapter 4
    - The classification method was Naïve Bayes
  - MH8111_MyOwnDataSet.R
    - I used Adult Dataset from UCI to predict income
    - https://archive.ics.uci.edu/ml/datasets/Adult
- Two Dataset files
  - sms_spam.csv is used by MH8111_ExampleFromBook.R
  - adult.data, which I downloaded, is used by MH8111_MyOwnDataSet.R

SMS classification with Naïve Bayes

- I followed the book example, and I’ve learnt the Naïve Bayes classification model.
- Some of the library functions used in the book is deprecated now, and I have to come up with my own work around.
- Overall, the example is easy to follow and it helped me understand Naïve Bayes model step by step.
- I’m not going to discuss this part as it’s very straight forward.

Adult Dataset Classification with Naïve Bayes

I will dedicate the rest of the slides to fully document the details of classification using the Adult dataset, here is a summary of what I'm going to cover in the following sections,

- **Data Description**
- **Data Exploration & Missing Data Handling**
- **Feature Engineering (The Major Component)**
- **Model Training & Prediction and Performance Evaluation**
- **Model Improvement & Performance Re-evaluation**
- **Summary and Afterwords**

Data Description

 # 0. income: >50K, <=50K. (class label)

1. age: continuous.

2. work_class: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov,

3. fnlwgt: continuous.

4. education: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc,

5. education-num: continuous.

6. marital_status: Married-civ-spouse, Divorced, Never-married, Separated,

7. occupation: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty,

8. relationship: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.

9. race: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.

10. sex: Female, Male.

11. capital-gain: continuous.

12. capital-loss: continuous.

13. hours-per-week: continuous.

14. country: United-States, Cambodia, England, Puerto-Rico, Canada, .

Data Exploration & Missing Data Handling

- **Data Exploration**
  - 14 features with 1 additional **income** as class label
  - In total 32561 records
    - > dim(adult_data)
    - [1] **32561   15**
- **Missing Data**
  - The missing data percentage is less than **1%**
    - sum(is.na(adult_data))/prod(dim(adult_data))
    - [1] 0.008726186
  - Given that the percentage of missing data is minimal, I’ve decided to drop all records with missing values. And after omitting the missing data records,
    - > dim(adult_data_full)
    - [1] **30162   12**
  - Hence, I’m using the **30162** records data set for this classification.

Feature Engineering

- **The following features have been converted/changed**
  - age
  - hours_per_week
  - education
  - marital_status
  - country
  - work_class
- **The following feature has been created**
  - investement
- **The following features have been removed**
  - education_sum
  - fnlwgt
  - capital_gain
  - capital_loss

Feature Engineering: age

Feature **age** is converted from continuous numerical to categorical using Binning,

- age<18 =**Youth**, probably haven't even started working
- 18<age<25 =**YoungAdult**, probably just started working
- 25<age<60 =**Adult**, main workforce
- age>60 =**SeniorAdult**, retiree

The splitting points chosen are based on general knowledge. Before 25, the work income usually won’t be much. While after 60, people starts to retire.

Feature Engineering: hours_per_week

Feature **hours_per_week** is converted from continuous numerical to categorical using Binning too,

- [0, 20] =**LOOSE**, not working or part-time working
- (20, 40] =**NORMAL**, normal working
- (40, 60] =**OVERTIME**, over normal working hours
- (60,] =**INSANE**, really long working hours, suggesting labor hardship or workholic

Different working hours per week definitely tells something on the nature of job.

Feature Engineering: education

The original **education** feature has **17** categories, and I've reduced the categories by grouping certain similar categories together, below is my **6** categories after reduction,

- PreHighSchool
- HighSchool
- Bachelor
- Master
- Doctor
- Other

Original education even contains 1th to 12th grades. I don't think it impacts income differently whether the candidate makes 4th grade or 6th grade. However, between an Bachelor degree holder and a 4th grade graduate, the impact is most likely significant.

Feature Engineering: marital_status & work_class

The original **marital_status** feature has **7** categories, and I managed to reduce it to **4** categories,

- Single
- Married
- MarriedBefore
- Widowed

The original **work_class** has **8** categories, and I have reduced it to **4** categories,

- PublicSector
- PrivateSector
- SelfEmployed
- Unemployed

Feature Engineering: country

The feature **country** is the one feature that I completely re-grouped based on location and level of development, from **41** countries to **9** categories,

- America_North (US, Canada)
- America_South (Columbia, etc)
- America_Latin (Guatemala, etc)
- Europe_West (developed European countries)
- Europe_East (less developed European countries, prior Soviet Union countries)
- Asia_SouthEast (Thailand etc)
- Asia_NorthEast (Japan)
- Asia_GreaterChina (China, Taiwan, HongKong)
- Other (all the rest)

The income levels between developed and development countries won’t be the same. In addition, neighbor countries of same development level tend to have similar income levels.

Feature Engineering: new investment feature

capital_gain and **capital_loss** are the two given features with continuous numerical values. Both **capital_gain** and **capital_loss** can take on either positive values or zero. I've added the new **investment** feature based on the following logic,

- investment=Gain
  - When capital_gain > 0
- investment=Loss
  - When capital_loss > 0
- investment=None
  - When capital_gain = 0 and capital_loss = 0

People don't make investment when they have no savings. Broadly speaking, just by checking whether a person invests or not, we can have some insights on whether he/she is financially healthy.

Furthermore, while investment gain has a positive effect on income, investment loss has a negative effect on income, that's why I split the investment into three categories.

Feature Engineering: dropped features

The following four features have been dropped,

- education_sum
  - This is a continuous numerical feature which is not Naïve Bayes friendly
  - It's redundant with education feature
- fnlwgt
  - Sampling weight, continuous numerical feature, again not Naïve Bayes friendly
  - Besides, I have no clue how to use it
- capital_gain
  - Created new investment feature based on it
- capital_loss
  - Created new investment feature based on it

Training, Prediction and Evaluation

- Training
  - The entire dataset has been roughly split into **2/3** for training and **1/3** for testing
    - train_data <- adult_data_full[1:20000,]
    - test_data <- adult_data_full[20001:30162,]
  - The proportion of class labels have been verified to make sure it's a balanced split,
    - > prop.table(table(**train_data**$income))
    - <=50K >50K
    - **0.7532** **0.2468**
    - > prop.table(table(**test_data**$income))
    - <=50K >50K
    - **0.7469002** **0.2530998**
  - Naive Bayes was used without any additional parameter
    - income_classifier <- naiveBayes(train_data, train_data$income)

Training, Prediction and Evaluation

- Prediction & Evaluation
  - CrossTable(test_pred, test_data$income, prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE, dnn = c('Predicted', 'Actual’))
  - Out of **10162** records in test data set, only **8** records were predicted wrongly.

Predicted	Actual		
	<=50K	>50K	Row Total
<=50K	7590	8	7598
>50K	0	2564	2564
Column Total	7590	2572	10162

Model Improvement and Re-evaluation

- Model Improvement with Laplace smoothing
  - I’ve introduce an additional parameter to the Naïve Bayes classifier
    - income_classifier2 <- naiveBayes(train_data, train_data$income, **laplace = 1**)
- Re-Evaluation
  - Out of **10162** records in test data set, I managed to reduce the **8** wrong predictions earlier to only **2** after applying Laplace smoothing.

Predicted	**Actual**		
	<=50K	>50K	Row Total
<=50K	7590	2	7592
>50K	0	2570	2570
Column Total	7590	2572	10162

Summary and Afterwords

One point worth mentioning is that, I found the Binning method to be very slow.

I've binned feature **age**, **hours_per_week** and **investment**, maybe my computer is really slow, it takes about 4~5 minutes to run the R codes.