COMP40370 Practical 4

[redacted]

October 14, 2020

1 Question 1: Association rules with Apriori

1.1 Question 1 - 1

Filter Out the Count Attribute from the data frame using

```
df.pop(”count”)
```

1.2 Question 1 - 2

Use the Apriori algorithm to generate frequent itemsets from the input data.

First we have to expand the DataFrame so that the columns are each features and that the rows indicate if the transaction includes that feature using true/false. I made a function which will do this

```
def expand_df(dfx):
    cols = list(dfx.columns)
    f_c = cols.pop(0)
    df1 = pd.get_dummies(dfx[f_c], prefix=f_c).reset_index()
    for c in cols:
        tmp = pd.get_dummies(dfx[c], prefix=c).reset_index()
        df1 = df1.merge(tmp, left_on='index', right_on='index')
    df1.pop(”index”)
    return df1
```

Then all we need to do is pass that dataframe to the mlxtend libriaries Apriori function and it will generate the Itemsets for us.

```
adf = expand_df(df)
apriori_df = apriori(adf, min_support=0.15, use_colnames=True, verbose=True)
```

How many frequent itemsets are produced?
20

How big are they?
13 sets are size 1
7 sets are size 2

1.3 Question 1 - 3

Saving the Output ItemSets

```
apriori_df.to_csv('./output/question1_out_apriori.csv', index=False)
```

1.4 Question 1 - 4

We can use mlxtends association_rules function to filter out all the rules that don't have a confidence above 90%

```
rule9 = association_rules(apriori_df, metric="confidence", min_threshold=0.9)[reqd]
```

How many rules are produced?
1 Rule is produced
For each rule, include a short description
This rule is that if someone is in the age range (21...25) they are a Junior

1.5 Question 1 - 5

Saving the Output Rules (confidence 0.9)

```
rule9.to_csv('./output/question1_out_rules9.csv', index=False)
```

1.6 Question 1 - 6

Again we use mlxtends association_rules function but this time to filter out all the rules that don't have a confidence above 70%

```
rule7 = association_rules(apriori_df, metric="confidence", min_threshold=0.7)[reqd]
```

How many rules are produced this time?
3 Rules are produced
For each rule, include a short description
Rule 1:(confidence 100%)
if someone is in the age range (21...25) they are a Junior
Rule 2:(confidence 71%)
If someone is majoring in philosophy then they are in the age range (26...30)
Rule 3: (confidence 80%)
If someone is a PhD then they are in the age range (26...30)

1.7 Question 1 - 7

Saving the Output Rules (confidence 0.7)

```
rule7.to_csv('./output/question1_out_rules7.csv', index=False)
```

2 Question 2: Association rules with FP-Growth

2.1 Question 2 - 1

Filter Out the ID Attribute from the data frame using

```
df.pop("id")
```

2.2 Question 2 - 2

To Discretize the numeric attributes into 3 bins of equal width we do the following

```
dfd = df.copy()
dfd["age"] = pd.cut(dfd["age"], 3, precision=0, duplicates="drop")
dfd["income"] = pd.cut(dfd["income"], 3, precision=0, duplicates="drop")
dfd["children"] = pd.cut(dfd["children"], 3, precision=0, duplicates="drop")
```

2.3 Question 2 - 3

First we have to expand the DataFrame so that the columns are each features and that the rows indicate if the transaction includes that feature using true/false. I made a function which will do this

```
def expand_df(dfx):
    cols = list(dfx.columns)
    f_c = cols.pop(0)
    df1 = pd.get_dummies(dfx[f_c], prefix=f_c).reset_index()
    for c in cols:
        tmp = pd.get_dummies(dfx[c], prefix=c).reset_index()
        df1 = df1.merge(tmp, left_on='index', right_on='index')
    df1.pop("index")
    return df1
```

Then all we need to do is pass that dataframe to the mlxtend libriaries FP-Growth function and it will generate the Itemsets for us

```
fpg_df = expand_df(dfd)
fpgrowth_res = fpgrowth(fpg_df, min_support=0.2, use_colnames=True)
```

How many frequent itemsets are produced? How big are they? Include this information in your report.
How many rules are produced?
231 Rules are produced
How big are they?
The Item-Sets are between 1 and 4 items big

2.4 Question 2 - 4

Saving the Output ItemSets

```
fpgrowth_res.to_csv('./output/question2_out_fpgrowth.csv', index=False)
```

2.5 Question 2 - 5

The Value to achieve atleast 10 rules while still having a high confidence is 0.79 (79%)

```
rules10 = association_rules(fpgrowth_res, metric="confidence", min_threshold=0.79)
```

2.6 Question 2 - 6

Saving the Output Rules

```
rules10.to_csv('./output/question2_out_rules.csv', index=False)
```

2.7 Question 2 - 7

Intesting rule 1:

```
['current_act_YES', 'age_(18.0,␣34.0]'] => ['income_(4956.0,␣24386.0]']
```

(confidence: 0.9019607843137255%)]] Explanation: this rules says that we can predict with high confidence if that your income is in the range 4956.0-24386.0 if we know your age and if you hold a current account.

Intesting rule 2:

```
['mortgage_NO', 'save_act_YES', 'pep_NO'] => ['married_YES']
```

(confidence: 0.8450704225352114%)
Explanation: this rules says that we can predict with high confidence if you are married if we know your PeP, Mortgage and savings status is