Brent Claypool
IEMS 308 HW 2

Executive Summary:

We have a relational database of transactional data for Dillards stores. In the database, there are over 120 million transaction records to query from. Only a subset of these records were chosen for anaysis due to the computational expense of dealing with such a large set of data. There are 453 total stores across 299 different cities in 31 states included in the data. For this analysis, I considered the transactions that occurred at store 6109 in my hometown Palmdale, CA. There are 336128 total transactions recorded at this location. I ran apriori and association_rules to find the rules that resulted in the highest life, support, and confidence in order to discover the 100 SKUs that should be moved to optimize the layout of the store.

Problem Statement:
This project is tasked with finding 100 products that might need to be moved in order to optimize the layout of the store.

Methodology:
After careful exploration of the data, only a subset of the data was used for analysis on this matter. I selected only the transactions that occurred at the Dillards location in Palmdale, CA. The corresponding store number is 6109. This left me with 336128 transactions to analyze. I wanted to also only use the samples that were purchases, so I excluded all returns from the query and this left me with around 168000 transactions. I used a SQL database and query to find these values. The biggest challenge was setting column names for data files that were missing these. I looked in the data dictionary to find descriptions of each feature, then looked into the datasets and observed the distributions of the values to determine which features corresponded to the column names described in the data dictionary. The following query gave me every details about the SKU, Transaction Number, Register Number, Department Number, and Department Number of every item purchased at the Dillards in Palmdale, CA

```
select transact.sku,transact.store,transact.trannum, skuinfo.dept, deptinfo.deptdesc
 from TRANSACT JOIN skuinfo on transact.sku = skuinfo.sku
 join deptinfo on skuinfo.dept = deptinfo.dept
where transact.store = 6109 AND transact.saletype = "P"
```

I took the results of this query and imported them into Pandas as a .csv file as “trans_in_palmdale.csv”. The kernel in my python notebook kept crashing so I took a random 50000 samples to analyze. I created dummy variables for each SKU so I could then group all observations by Transaction Number to find out which of the products were purchased in the same transaction. I applied association rules using the functions apriori and association_rules to find association rules for these purchases.

Results:
Ultimately, my code did not finish running by the deadline, but I did everything according to how it was done in the lab session uploaded by the professor. I would have found the 100 rules

with the greatest lift support and confidence, then extracted the SKU's that involve them. I would have selected the 100 unique SKUs that resulted in the highest values as my recommendation to Dillards. I should have started much earlier in order to make sure my laptop had ample time to run the code, so I apologize for that, but my python notebook should reflect that I was on the right path to get the result.