Elo Merchant Category Recommendation

This project is intended to help understand customer loyalty and build a recommendation engine with discount from credit card provider

Overview

This project focuses on

Data Wrangling – Methods used to transform data into statistical usable format

EDA – Visual insights into data and correlation

Featuring Engineering – To create Features which will generate

Prediction model – Machine learning algorithms used and methods applied to predict the model

Conclusion – Findings of the Machine learning models

Introduction

ELO, one of the largest payment brands in Brazil, has built partnerships with merchants in order to offer promotions or discounts to cardholders.

Data is at https://www.kaggle.com/c/elo-merchant-category-recommendation/data

This project intends to clean data and perform EDA.

This project is divided into three parts **Data Wrangling, EDA, Featuring Engineering and Machine Learning Model.**

Data Dictionary

There are 6 Data sets

1. **train.csv** - contain card_ids and information about the card itself - the first month the card was active, etc. train.csv also contains the target

2. **test.csv** - contain card_ids and information about the card itself - the first month the card was active, etc.

3. **historical_transactions.csv** - designed to be joined with train.csv, test.csv, and merchants.csv. They contain information up to 3 months' worth of historical transactions for each card_id

4. **new_merchant_transactions.csv -** designed to be joined with train.csv, test.csv, and merchants.csv. They contain information about two months' worth of data for each card_id containing ALL purchases that card_id made at merchant_ids that were not visited in the historical data

5. **merchants.csv** - additional information about all merchants / merchant_ids in the dataset. Merchants can be joined with the transaction sets to provide additional merchant-level information.

6. **sample_submission.csv** - a sample submission file in the correct format - contains all card_ids you are expected to predict for.

Data Wrangling

Following data cleaning methods are used **merchant.csv**

- **Missing Data**
  - Columns having inf are replaced first with NaN and then are imputed based on datatype of column as described below.
  - Columns with object datatype having NaN values are imputed with "other"
  - Columns with int and float datatype having NaN values are imputed with median
- **Outliers -** Outlier identification is applied for following columns. Other columns are either categorical or ID's. **3-Sigma** Rule is applied to impute outliers.
  - numerical_1
  - numerical_2
  - avg_sales_lag3

Data Wrangling

- **Outliers -** contniued
  - avg_purchases_lag3
  - avg_sales_lag6
  - avg_purchases_lag6
  - avg_sales_lag12
  - avg_purchases_lag12

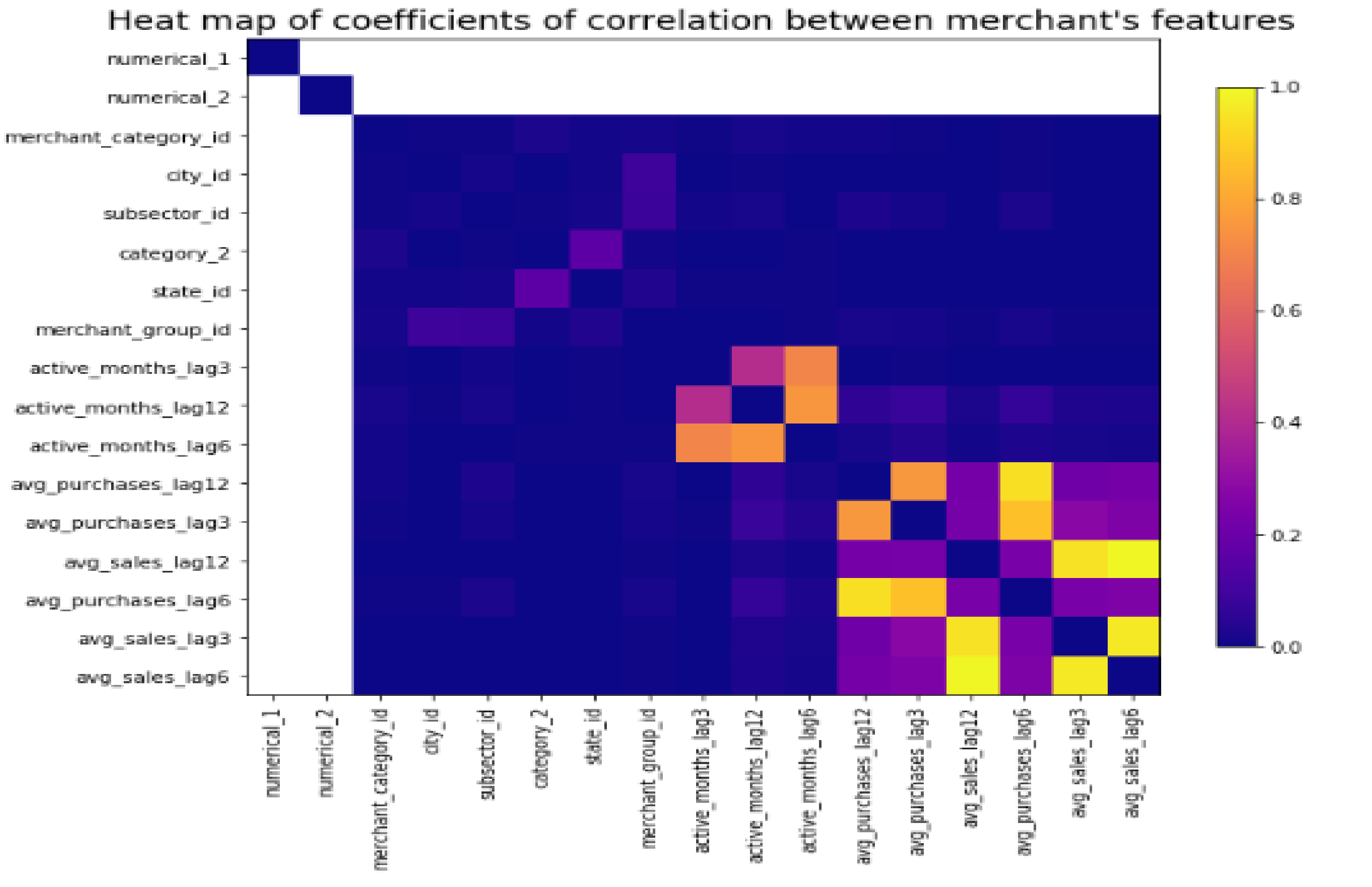
Data Wrangling

- For datasets **historical_transactions.csv** and **new_merchant_transactions.csv** –
  - **Missing values** (**NaN**)are imputed with “**other**” for columns with object datatype, **median** for columns with int and float datatype, and **new** category is added for columns with categorical datatype.
  - **Outliers** are imputed with **3-Sigma** rule for columns “**purchase_amount**” and “**installments**”
- Datetime features are created for “**purchase_date**”
  - Purchase year
  - Purchase month
  - Purchase day of the week
  - Purchase week of the year
  - Purchase weekend
  - Purchase hour
  - month difference - difference in numbers of months from current date to purchase date

EDA –merchant.csv data

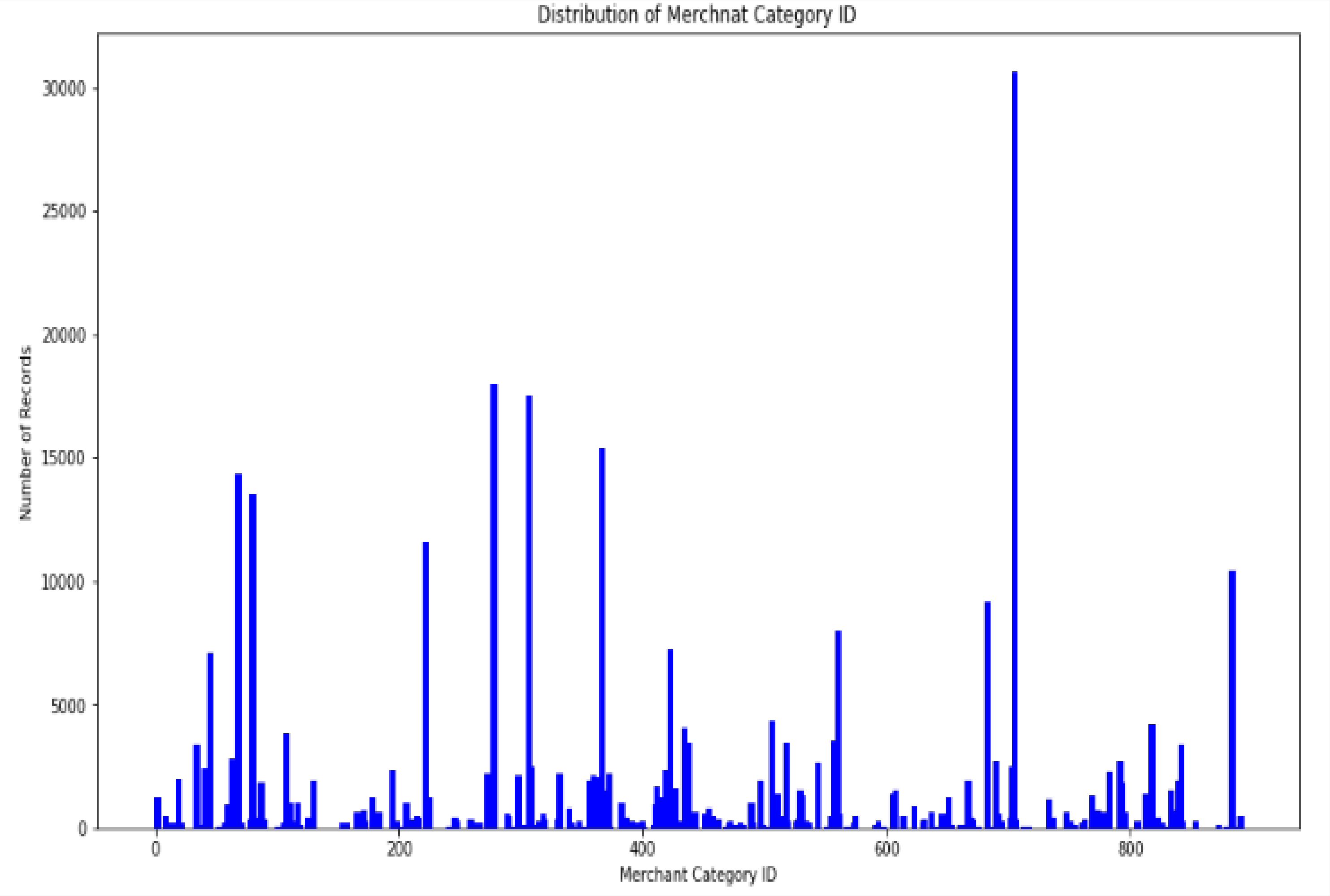
- There is no corelation numerical_1 and numerical_2 feature.

- There is correlation between avg_sales and avg_purchases of 3, 6 an 12 month.



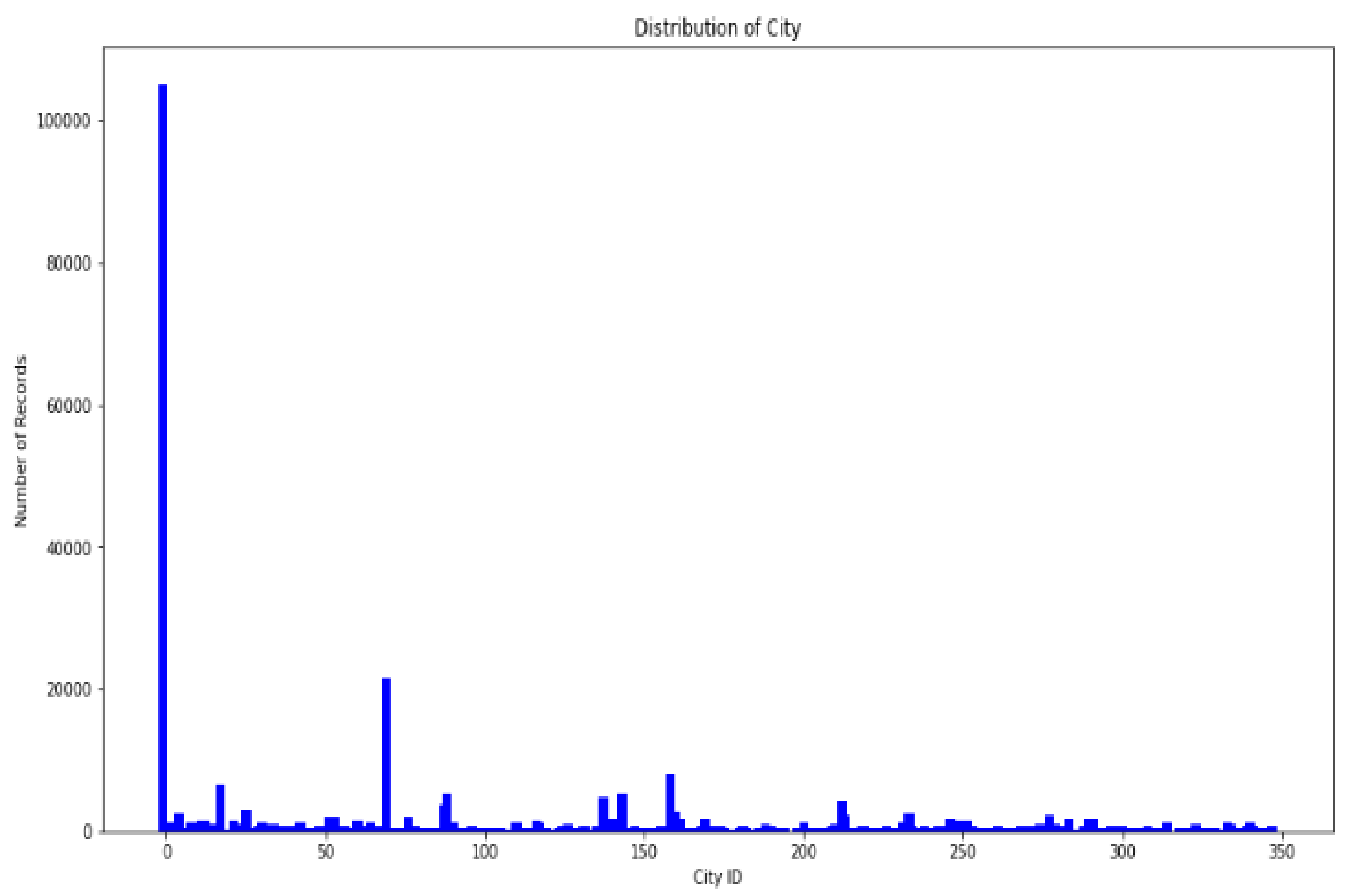
EDA –merchant.csv data

Merchant category ID 705 is the most famous merchant category with 9% sales



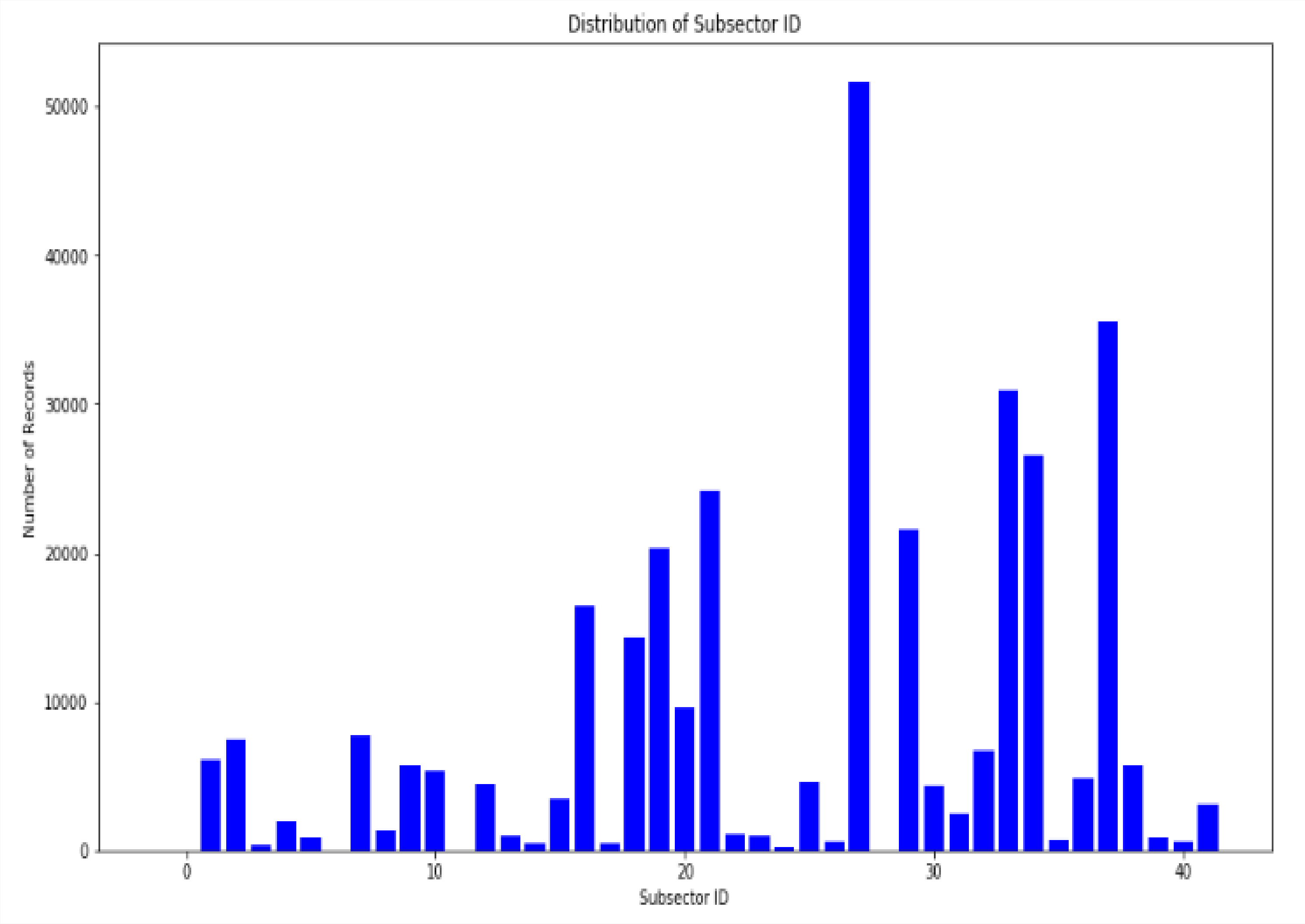
EDA –merchant.csv data

City ID -1 has over 100000 transactions and amounts to 31% of transactions



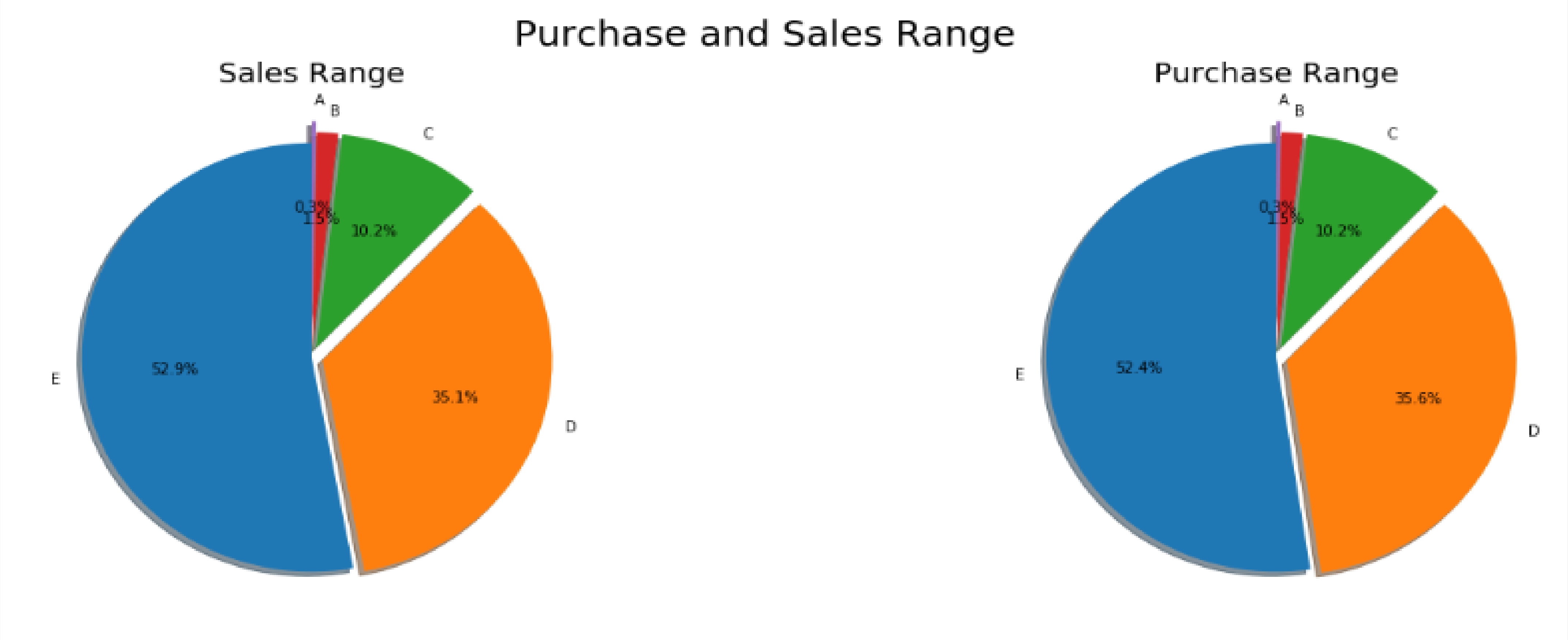
EDA –merchant.csv data

Subsector ID 27 has over 50000 transactions and amounts to 15% of transactions



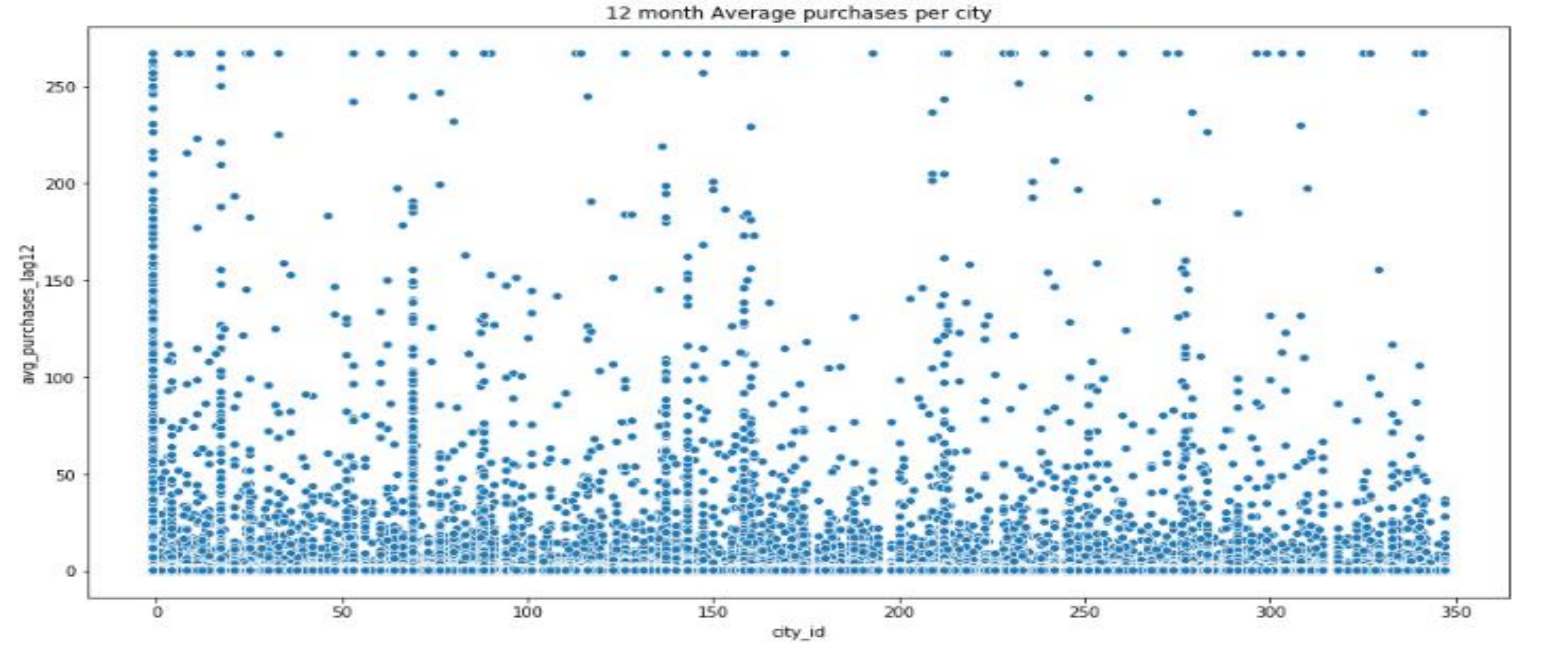
EDA –merchant.csv data

EDA –merchant.csv data



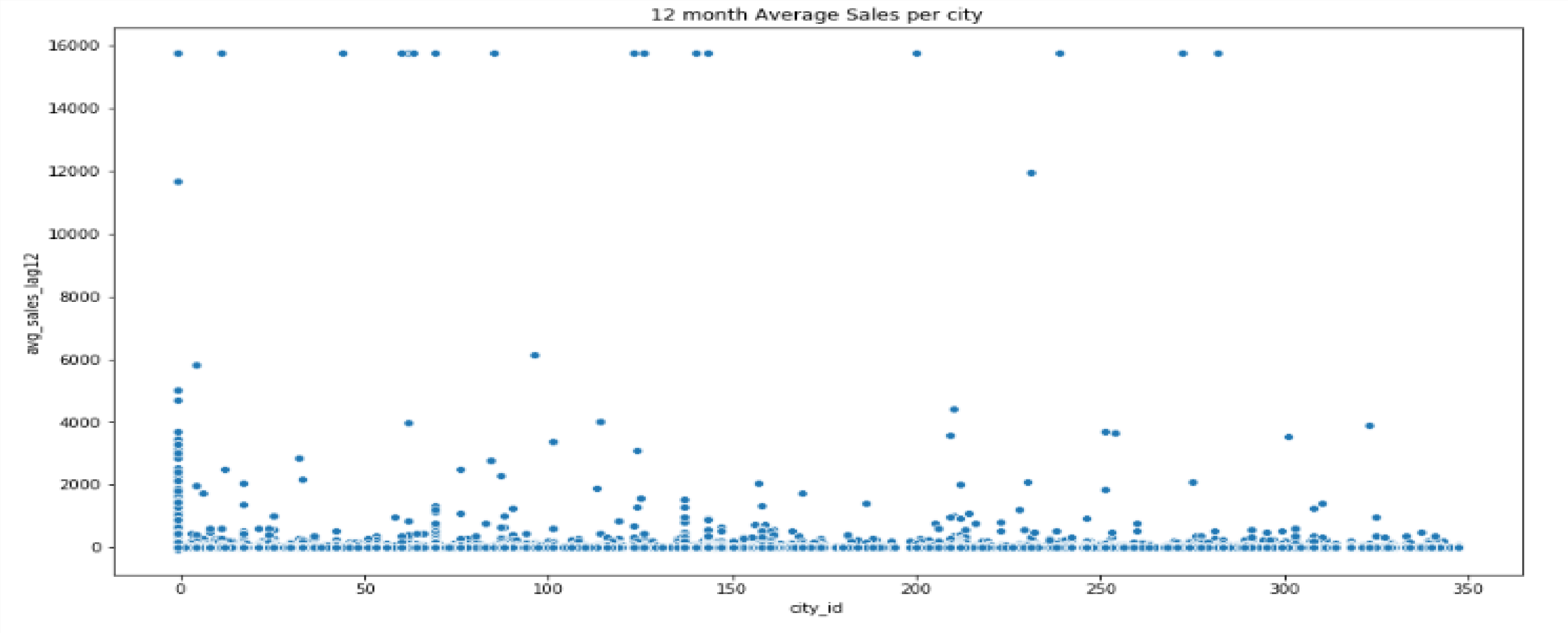
EDA –merchant.csv data

12 Month average purchases distribution per city



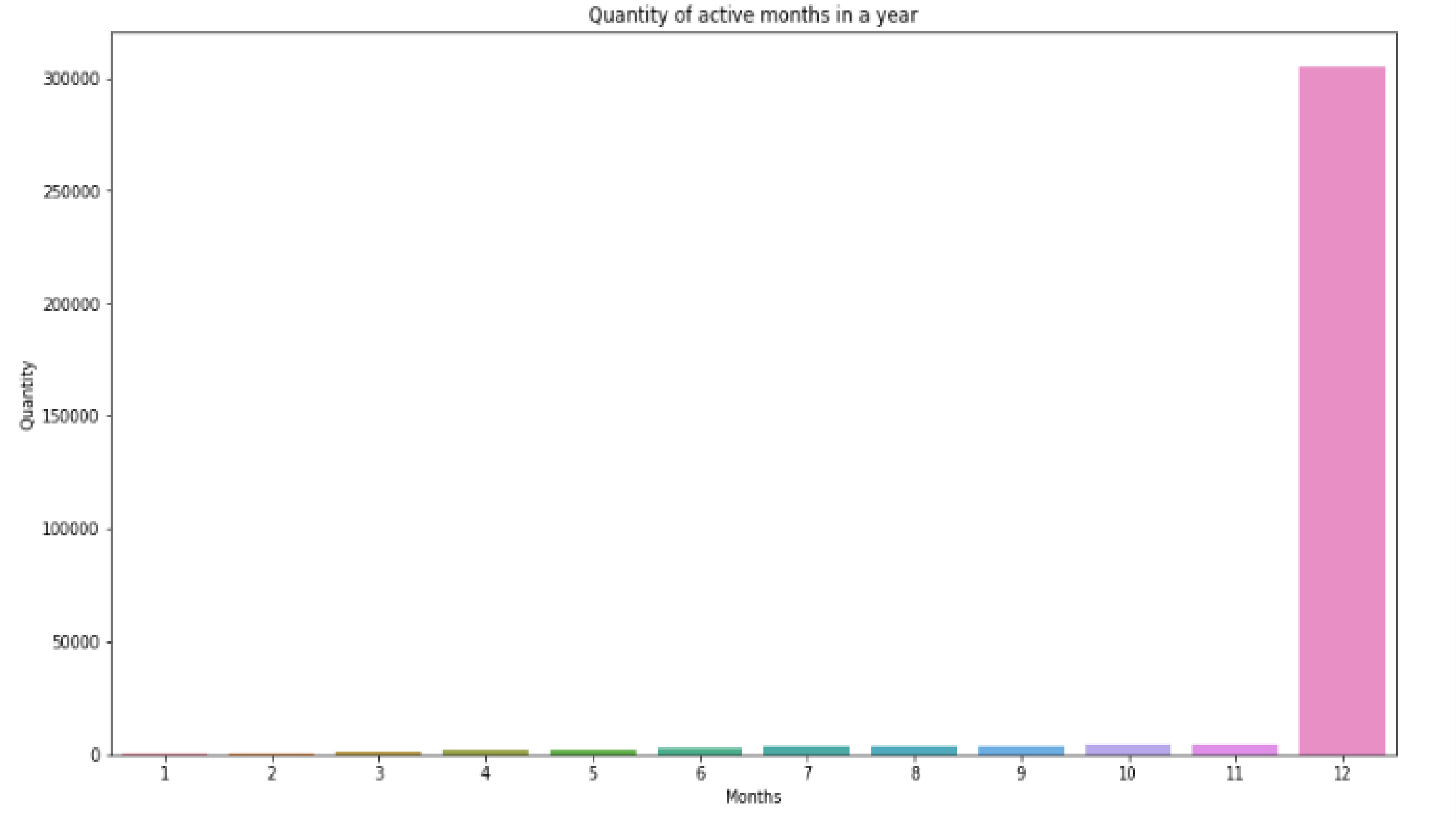
EDA –merchant.csv data

12 Month average sales distribution per city



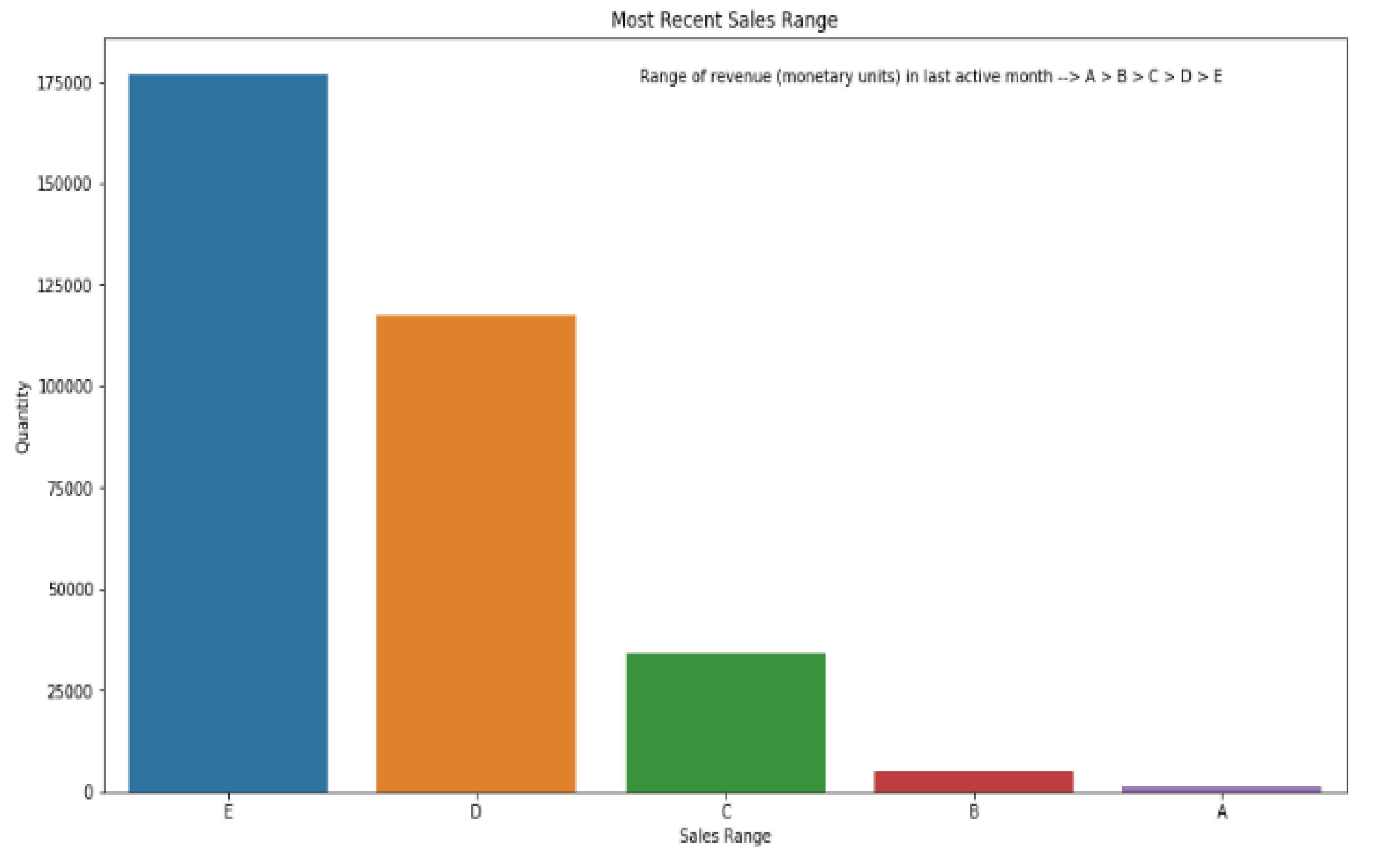
EDA –merchant.csv data

Most Sales are in the month of December



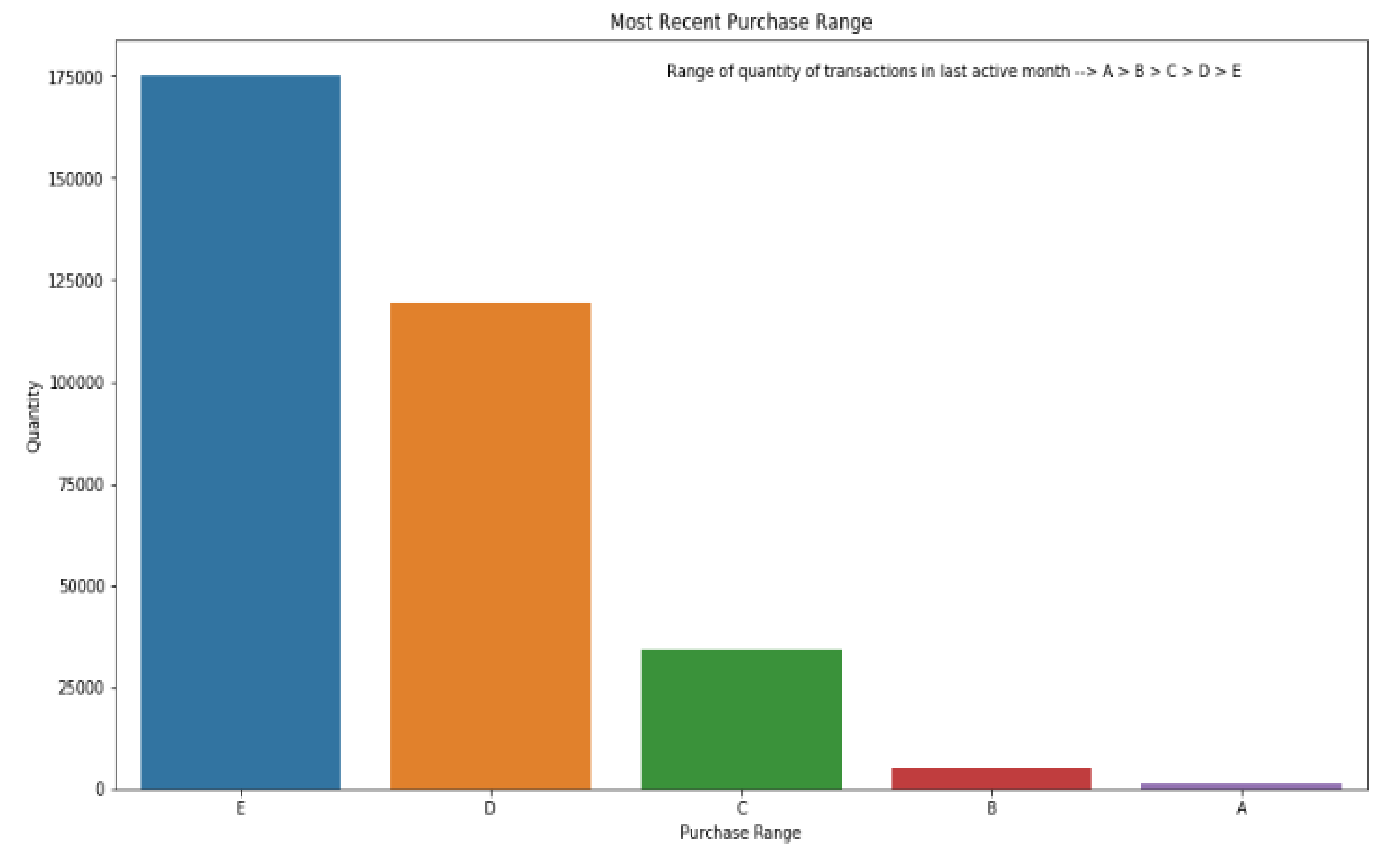
EDA –merchant.csv data

Most number of sales are in E category Range.



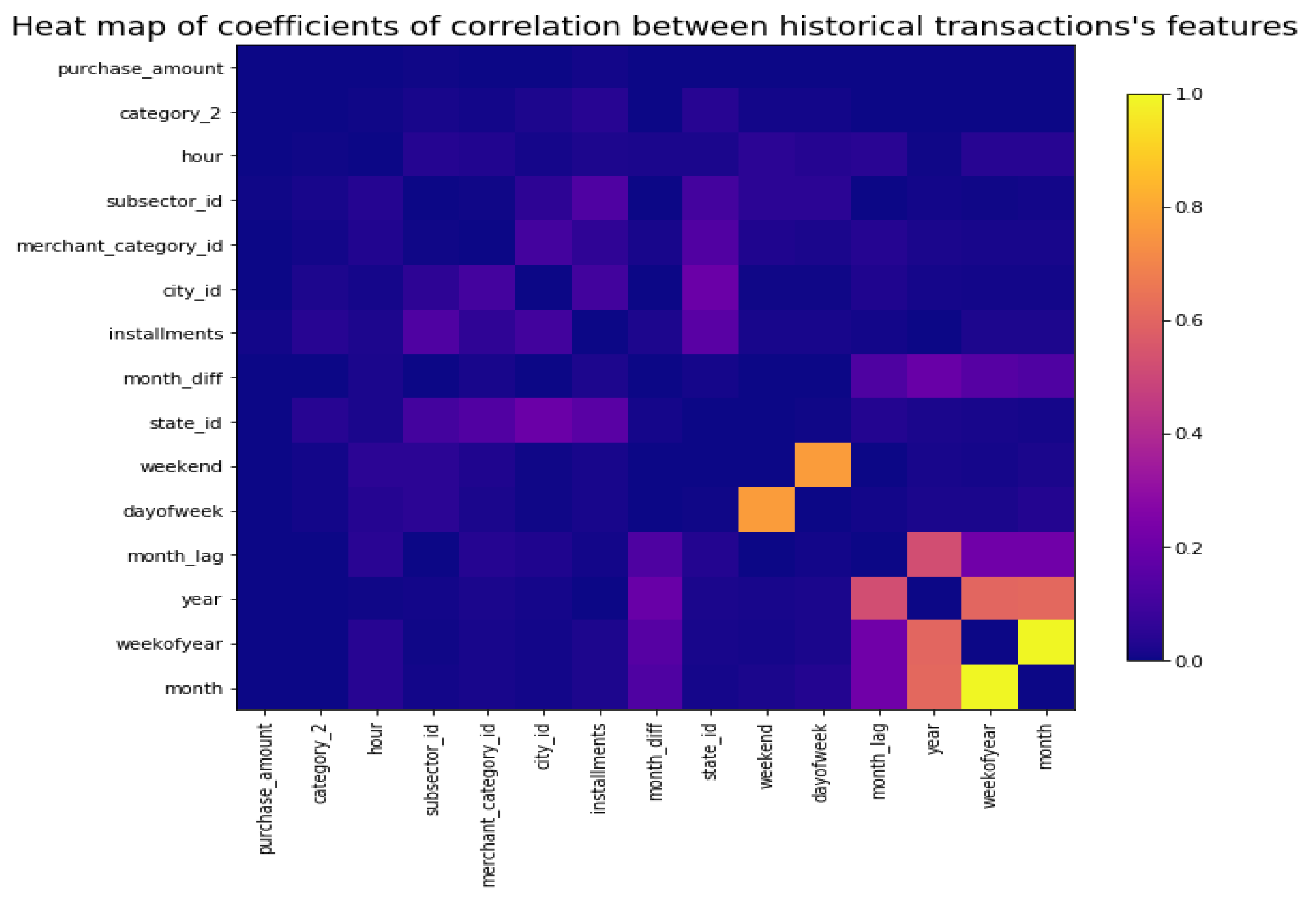
EDA –merchant.csv data

Most number of purchases are in E category Range.



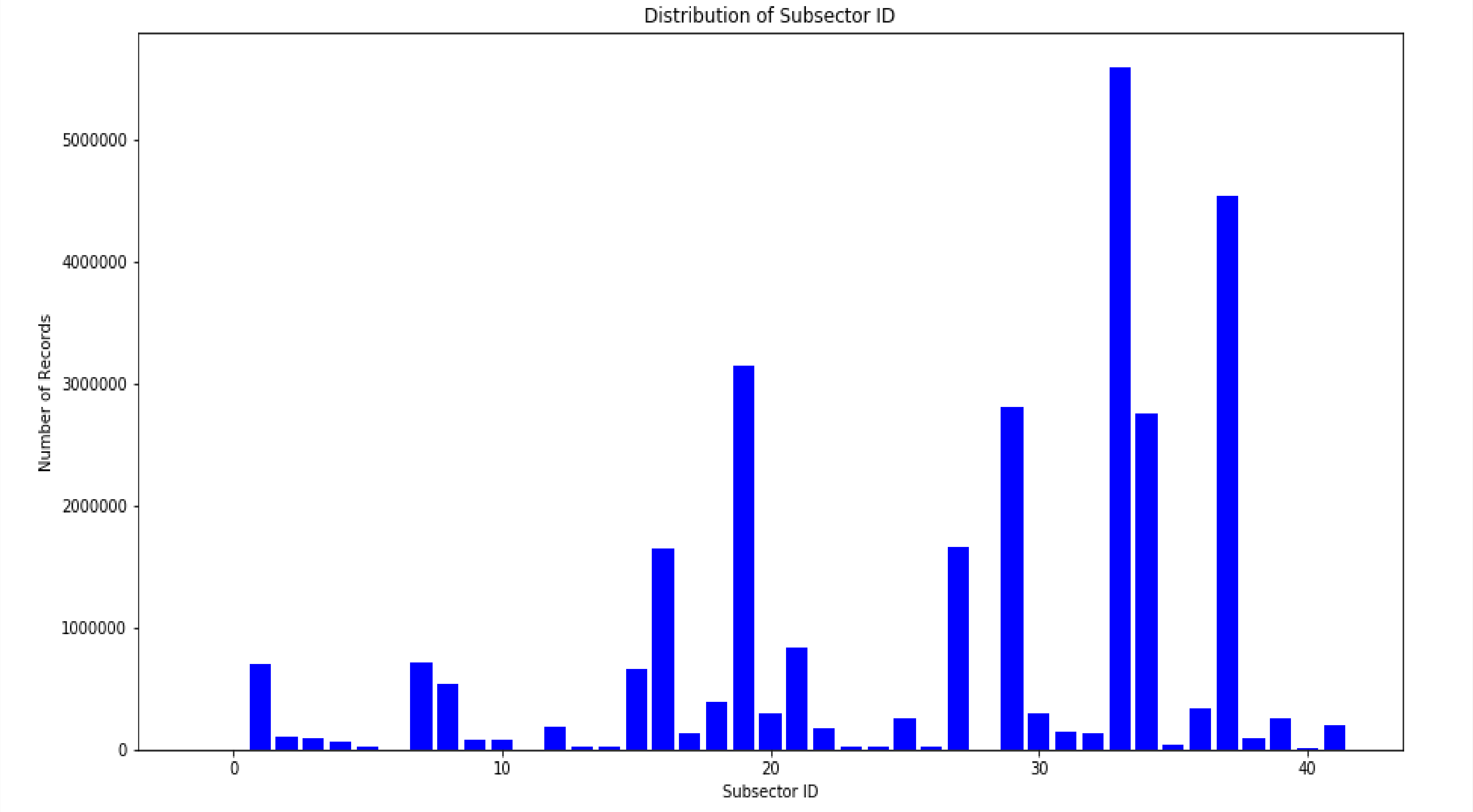
EDA –historical_transactions.csv data

There seems to be no correlation between features.



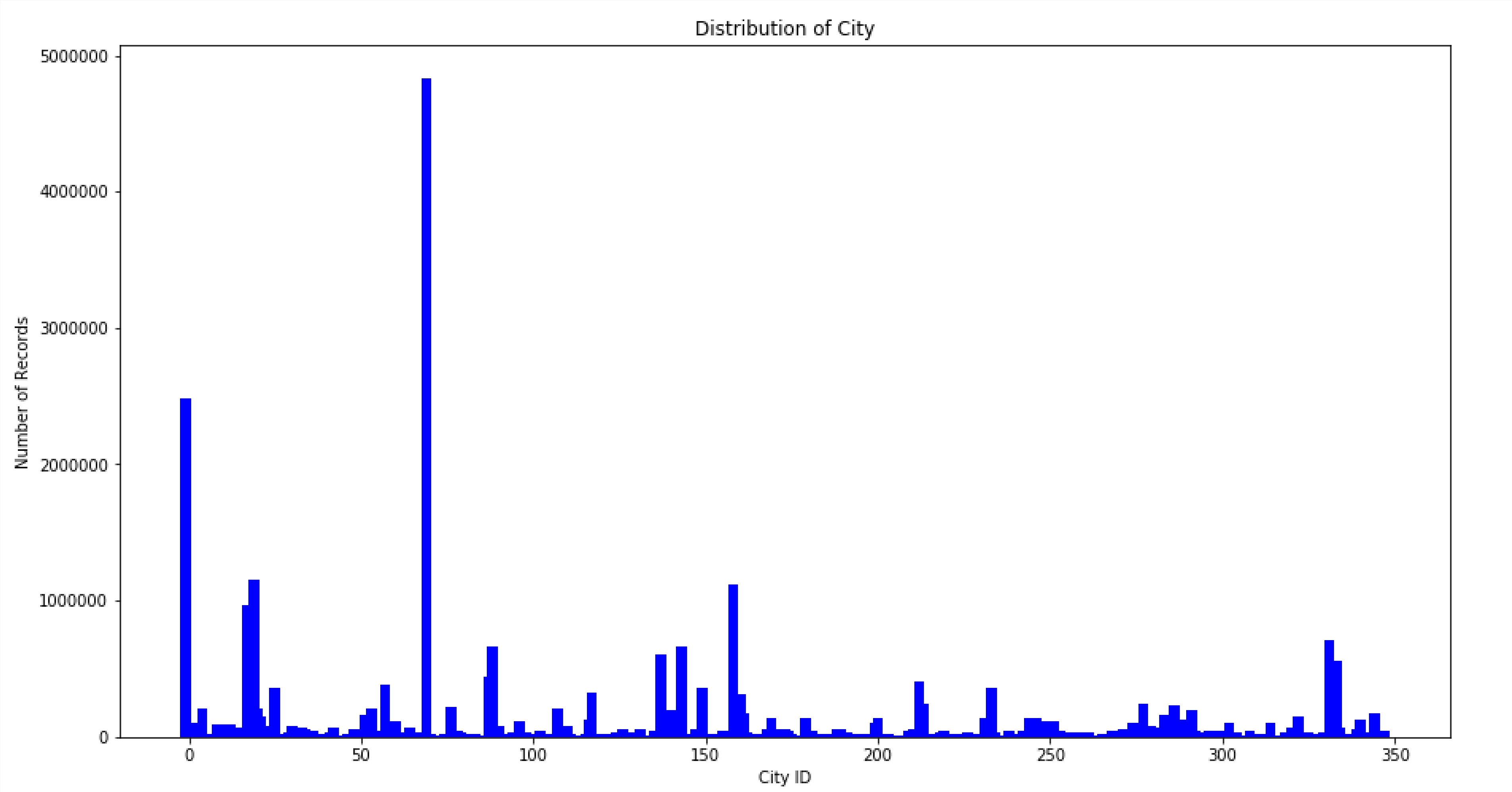
EDA –historical_transactions.csv data

Subsector ID 33 has over 5000000 transactions and amounts to 19% of transactions

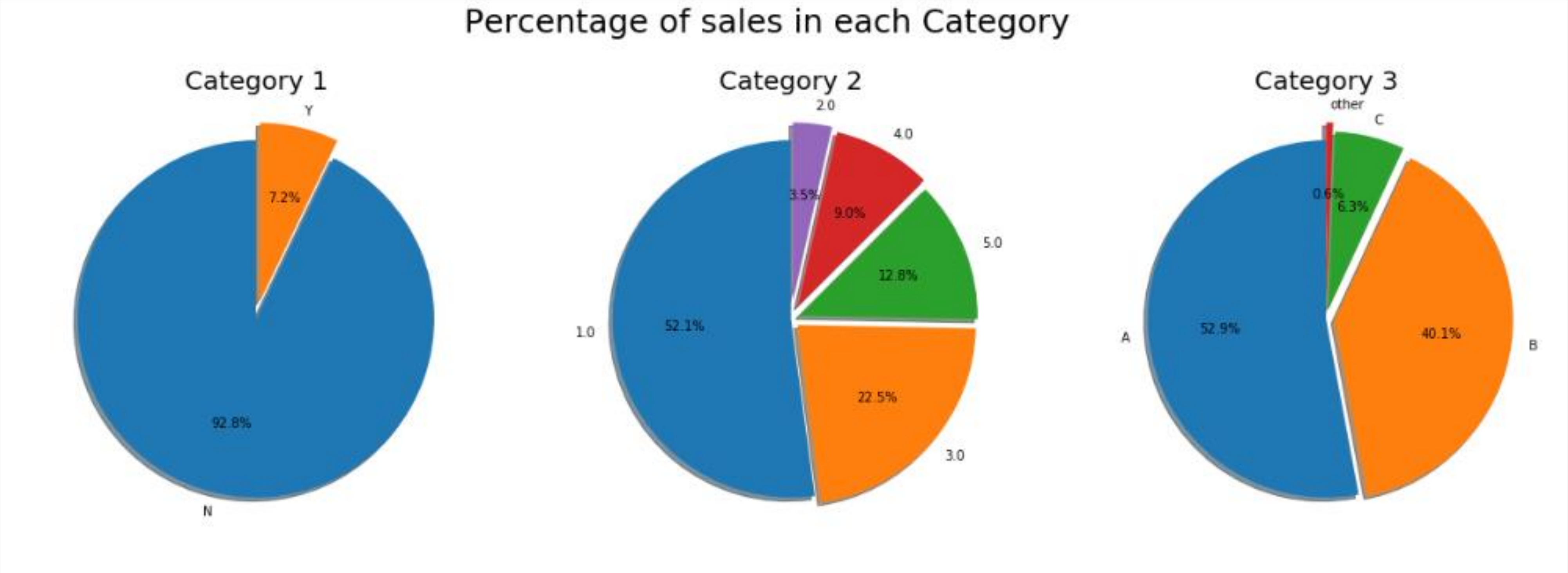


EDA –historical_transactions.csv data

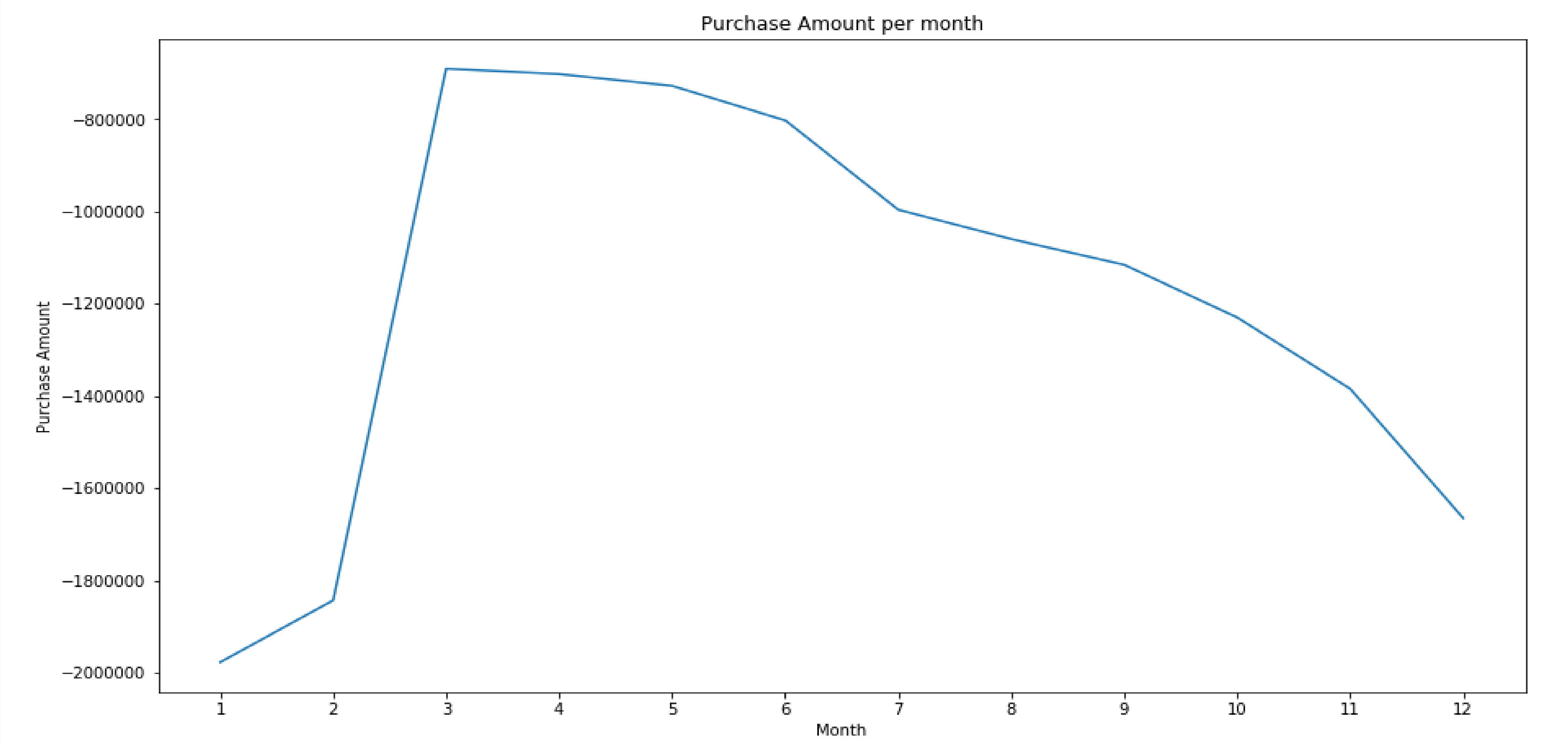
City ID 33 has over 4000000 transactions and amounts to 16% of transactions



EDA –historical_transactions.csv data

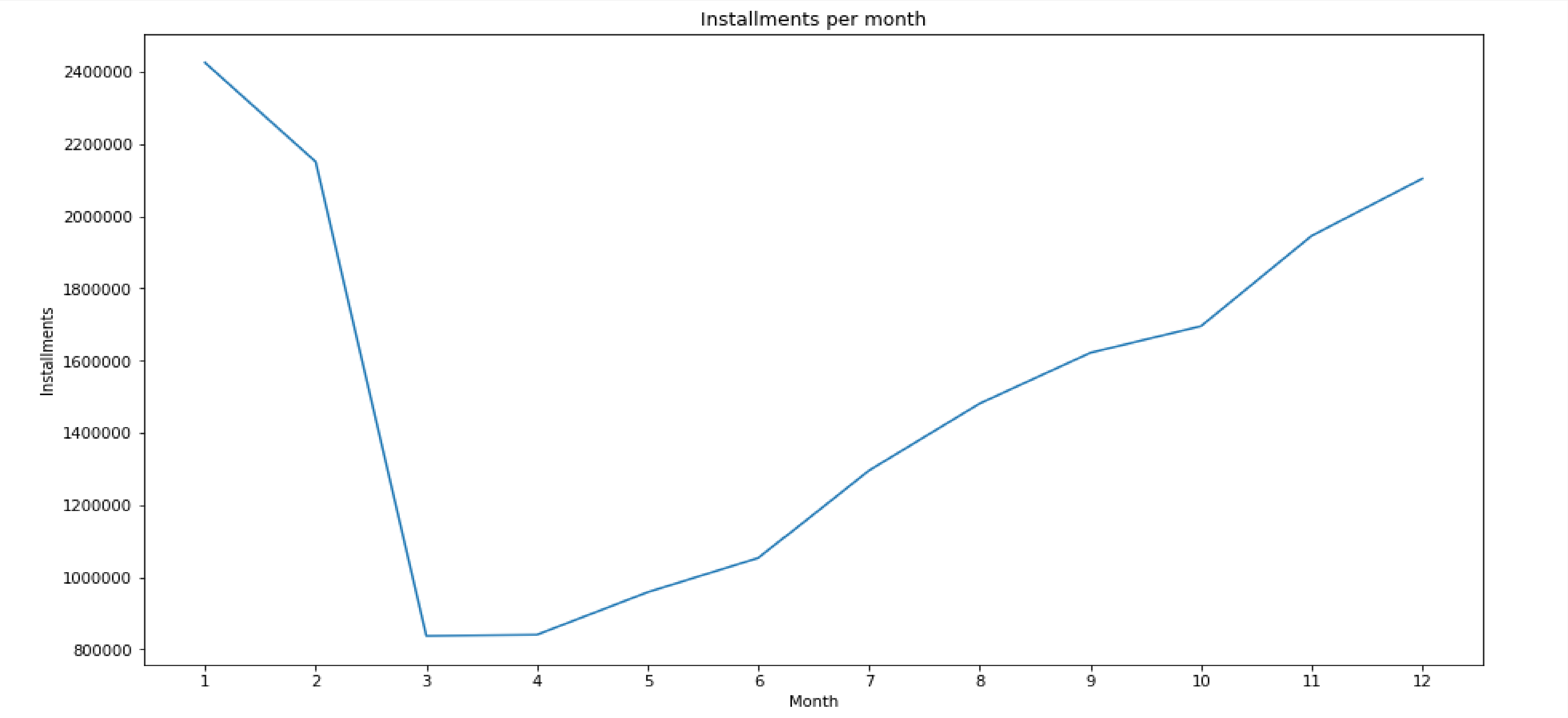


EDA –historical_transactions.csv data



March has most purchases per month.

EDA –historical_transactions.csv data



January has most installments per month.

EDA –historical_transactions.csv data

Most number of purchases are not part of category 1.

Highest number of purchase in category 2 are in **1.0**.

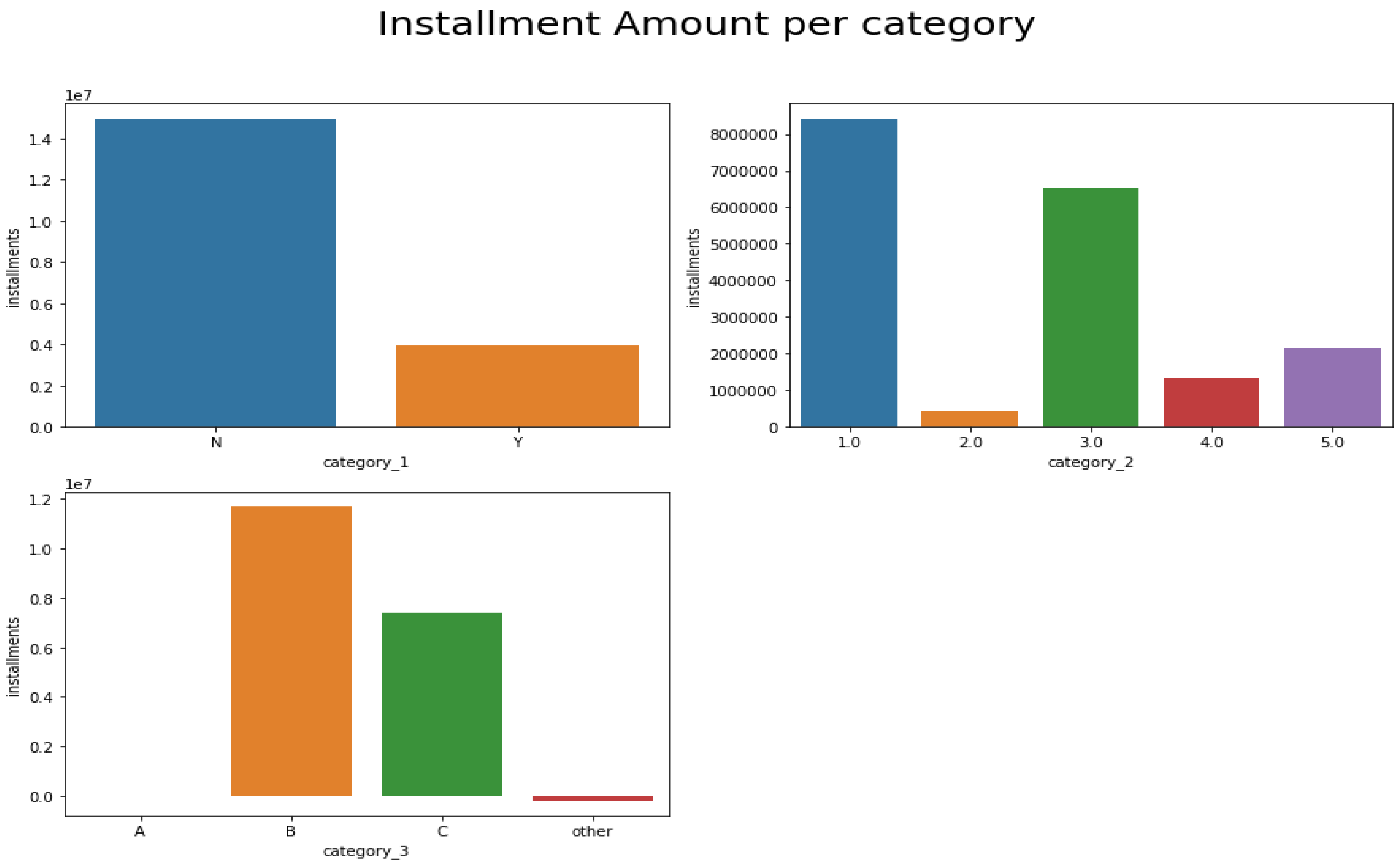
Highest number of purchase in category 3 are in **A**.

EDA –historical_transactions.csv data

Most number of installments are not part of category 1.

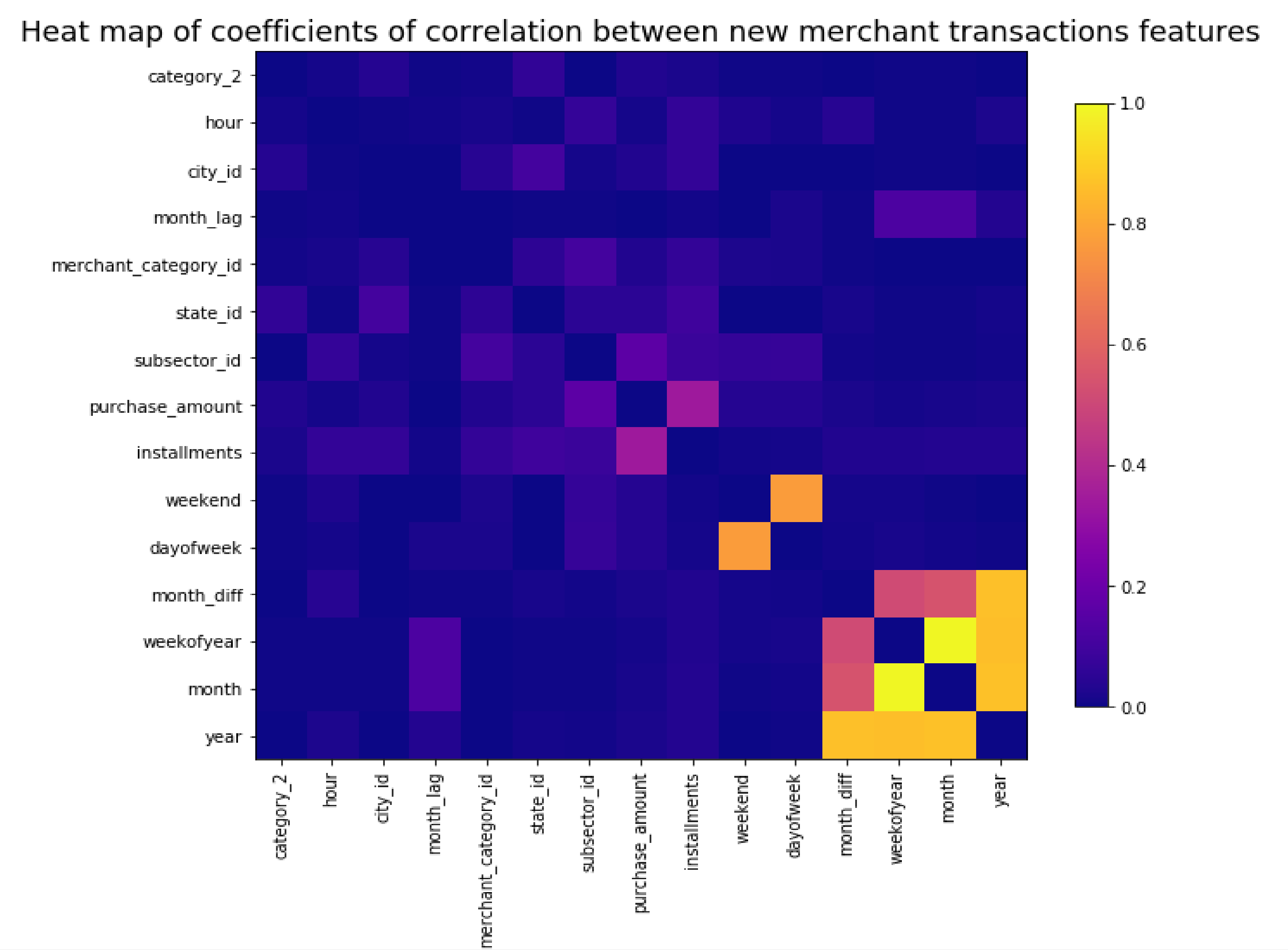
Highest number of installments in category 2 are in **1.0**.

Highest number of installments in category 3 are in **B**.



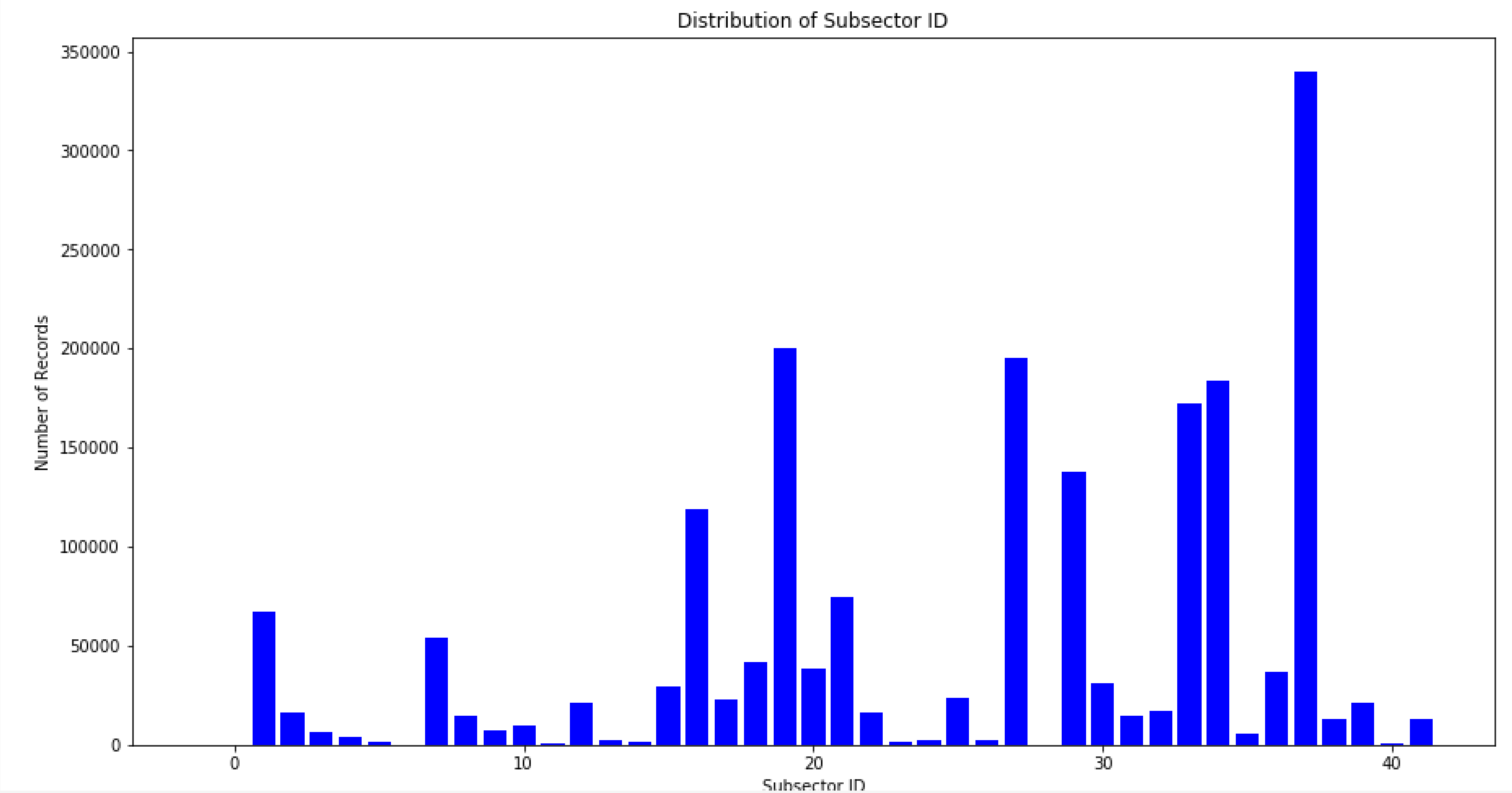
EDA – newMerchant_transactions.csv data

There seems to be a correlation purchase amount and number of installments.



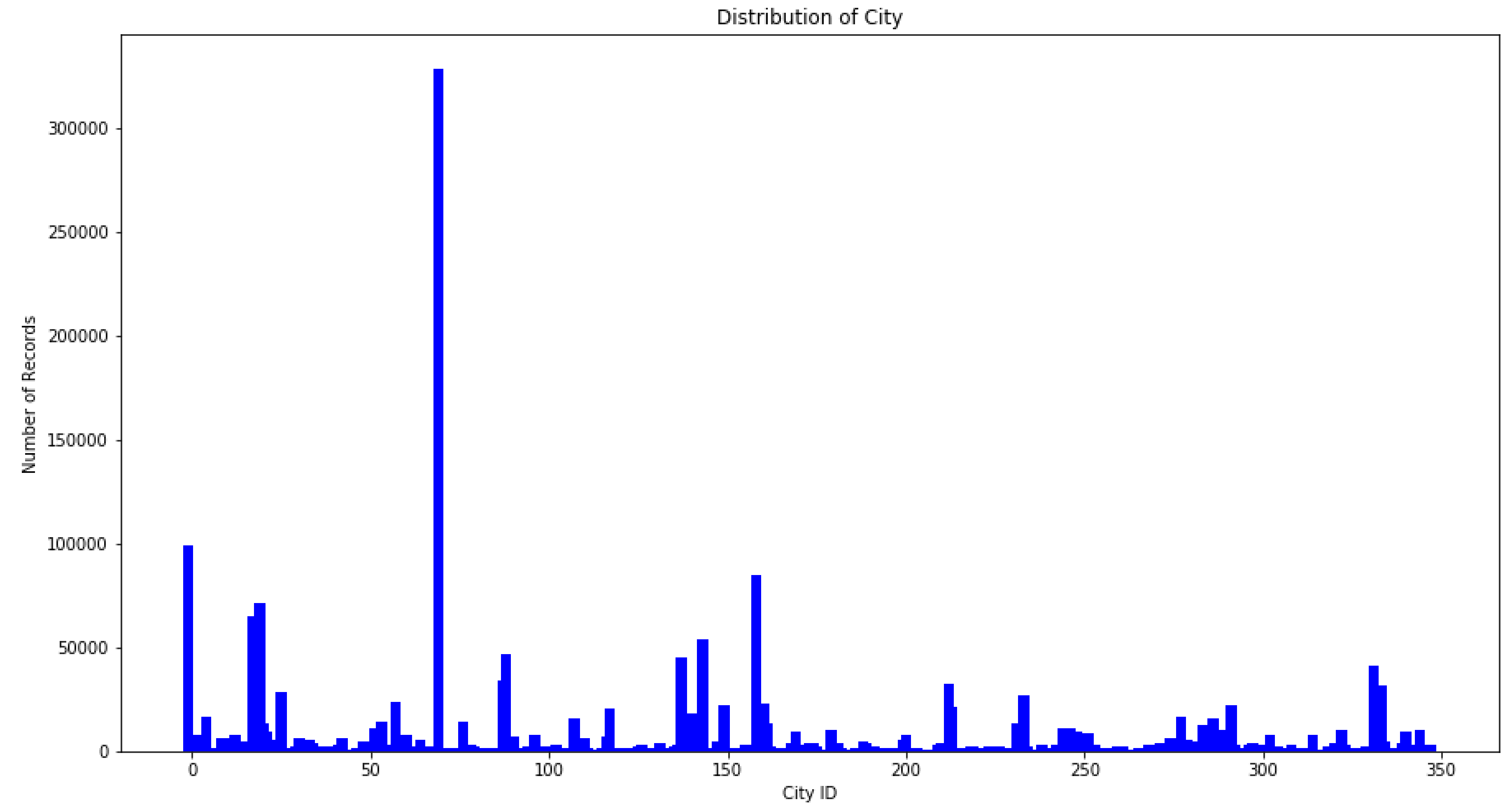
EDA – newMerchant_transactions.csv data

Subsector ID 37 has over 340053 transactions and amounts to 17% of transactions

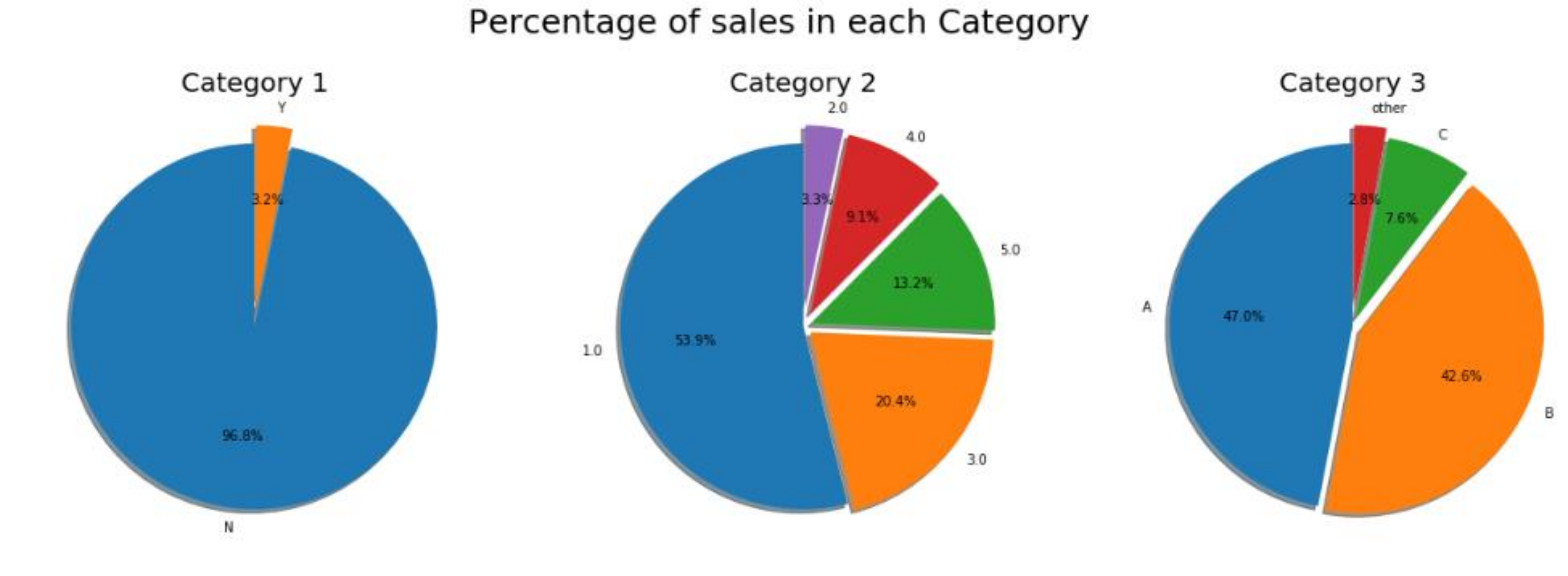


EDA – newMerchant_transactions.csv data

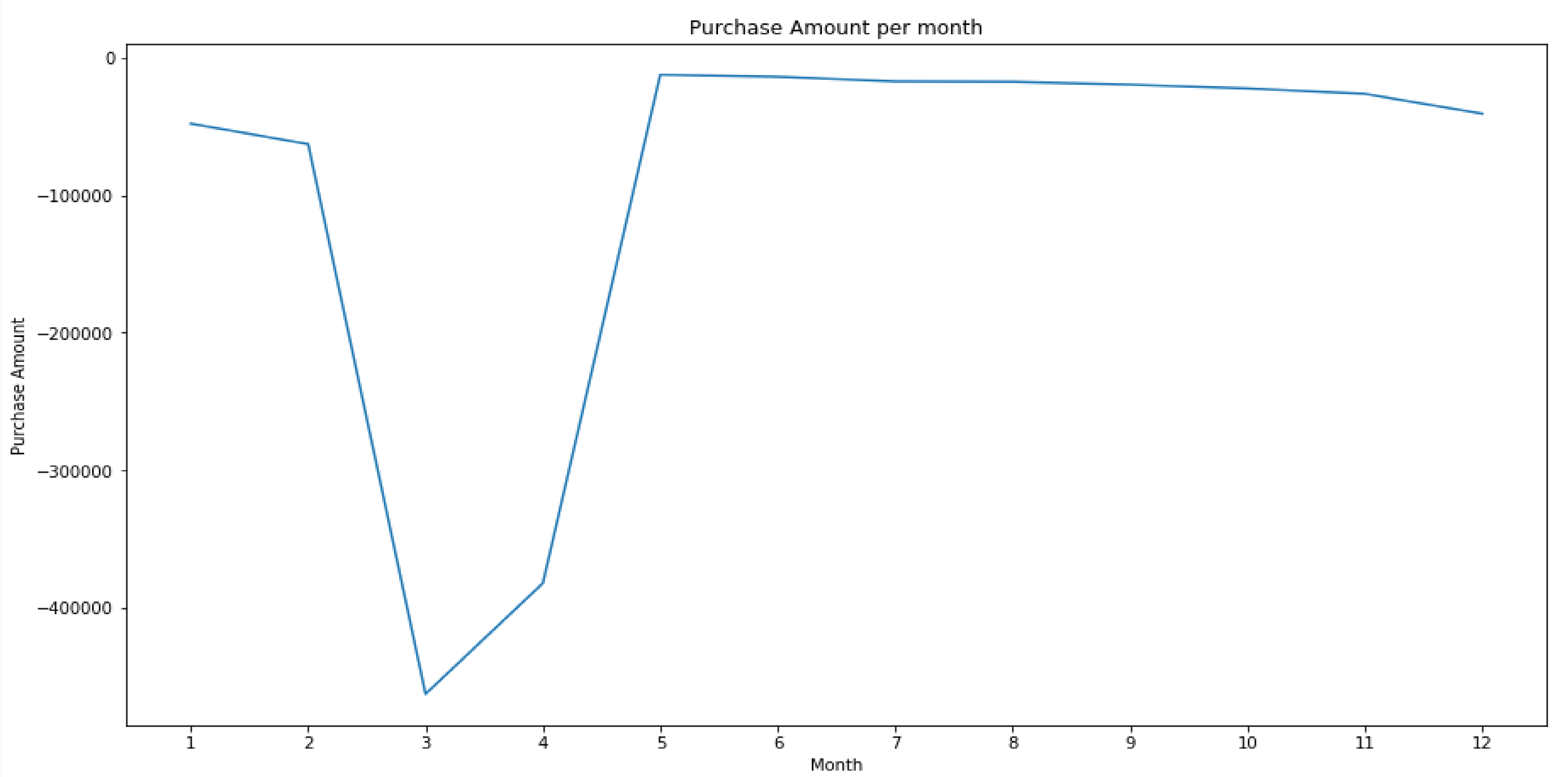
City ID 69 has 328916 transactions and amounts to 17% of transactions



EDA –EDA – newMerchant_transactions.csv data

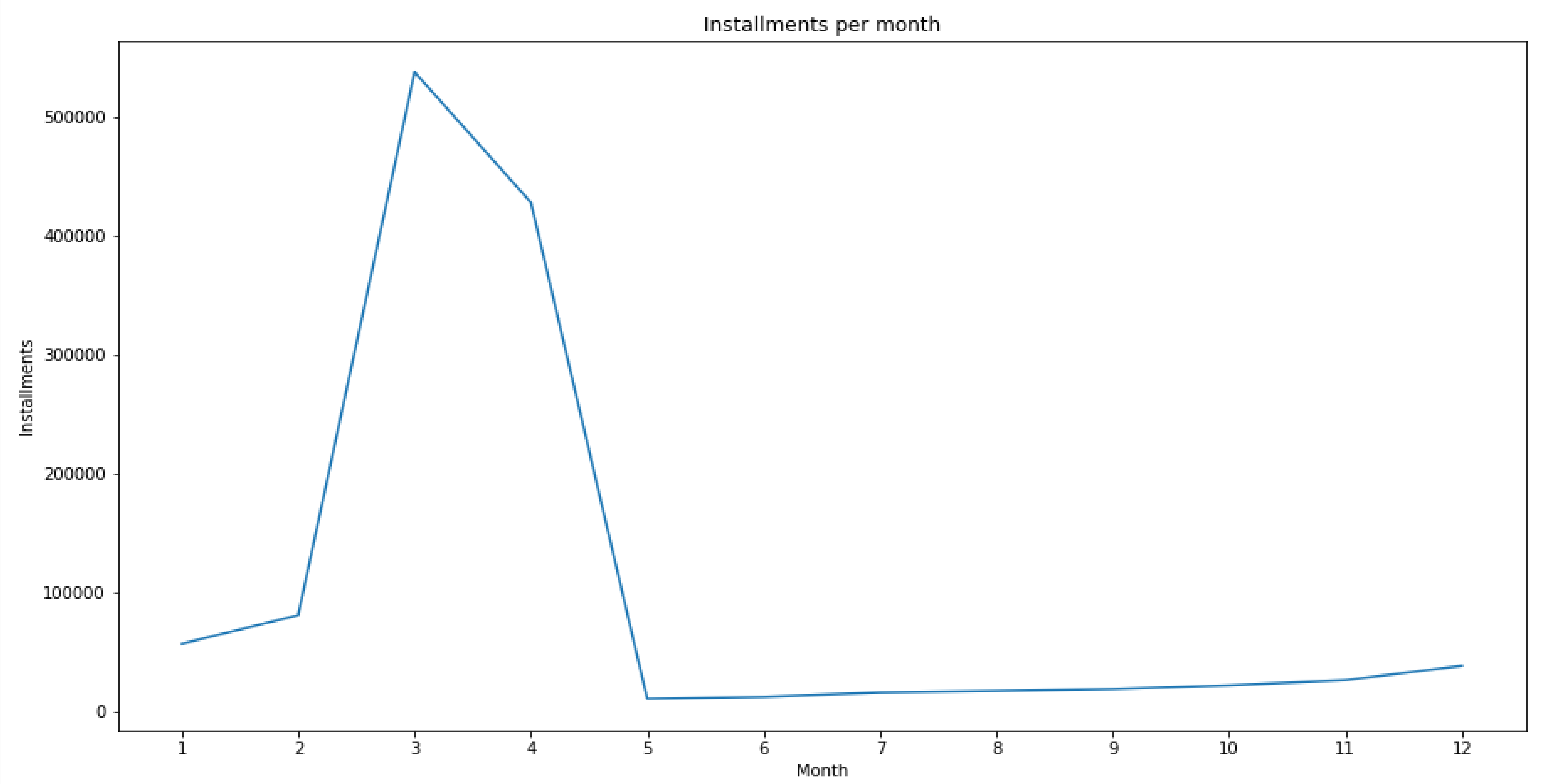


EDA – newMerchant_transactions.csv data



March has least purchase per month and there is constant purchases from May to December

EDA – newMerchant_transactions.csv data



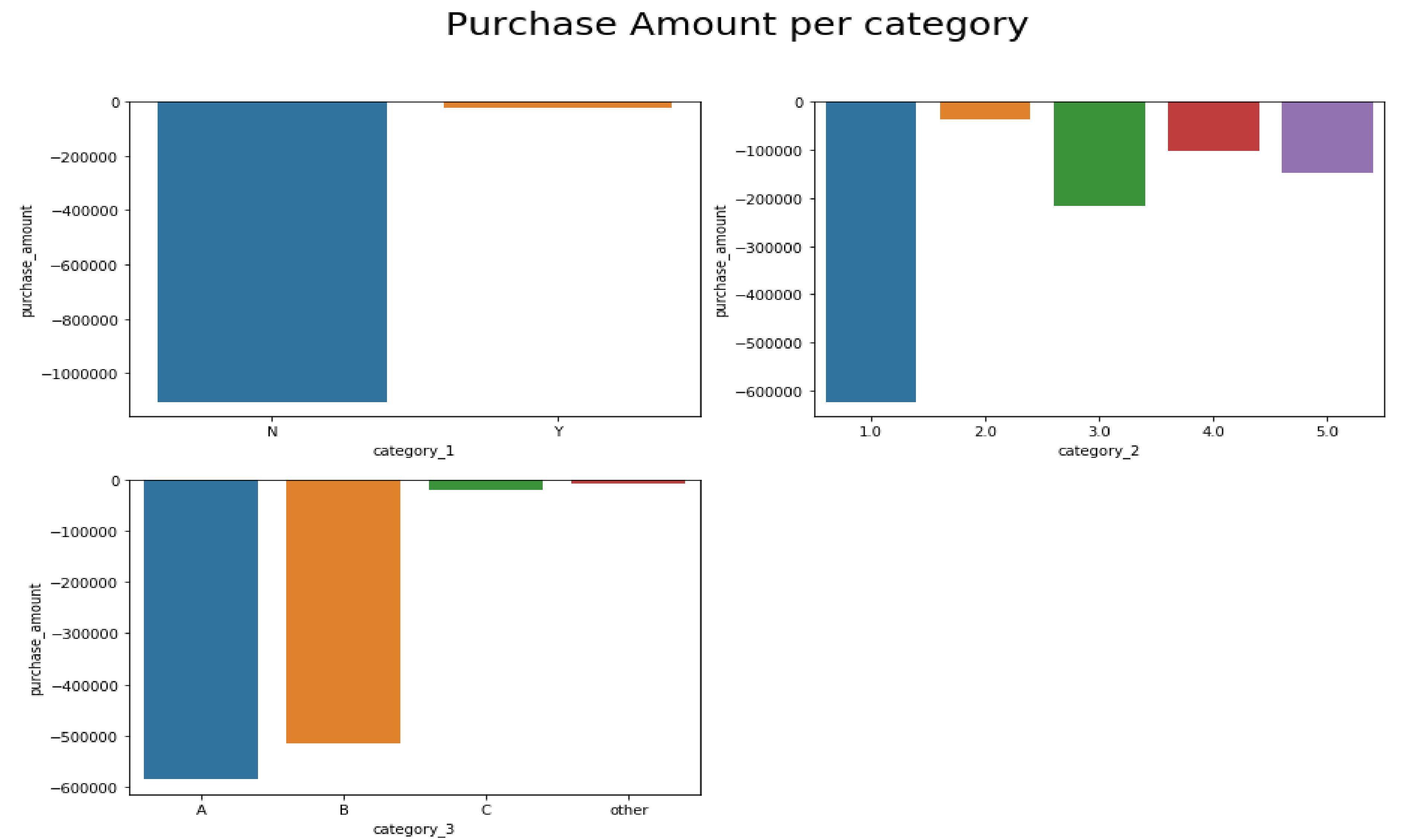
March has most installments per month.

EDA – newMerchant_transactions.csv data

Most number of purchases are not part of category 1.

Highest number of purchase in category 2 are in **1.0**.

Highest number of purchase in category 3 are in **A**.

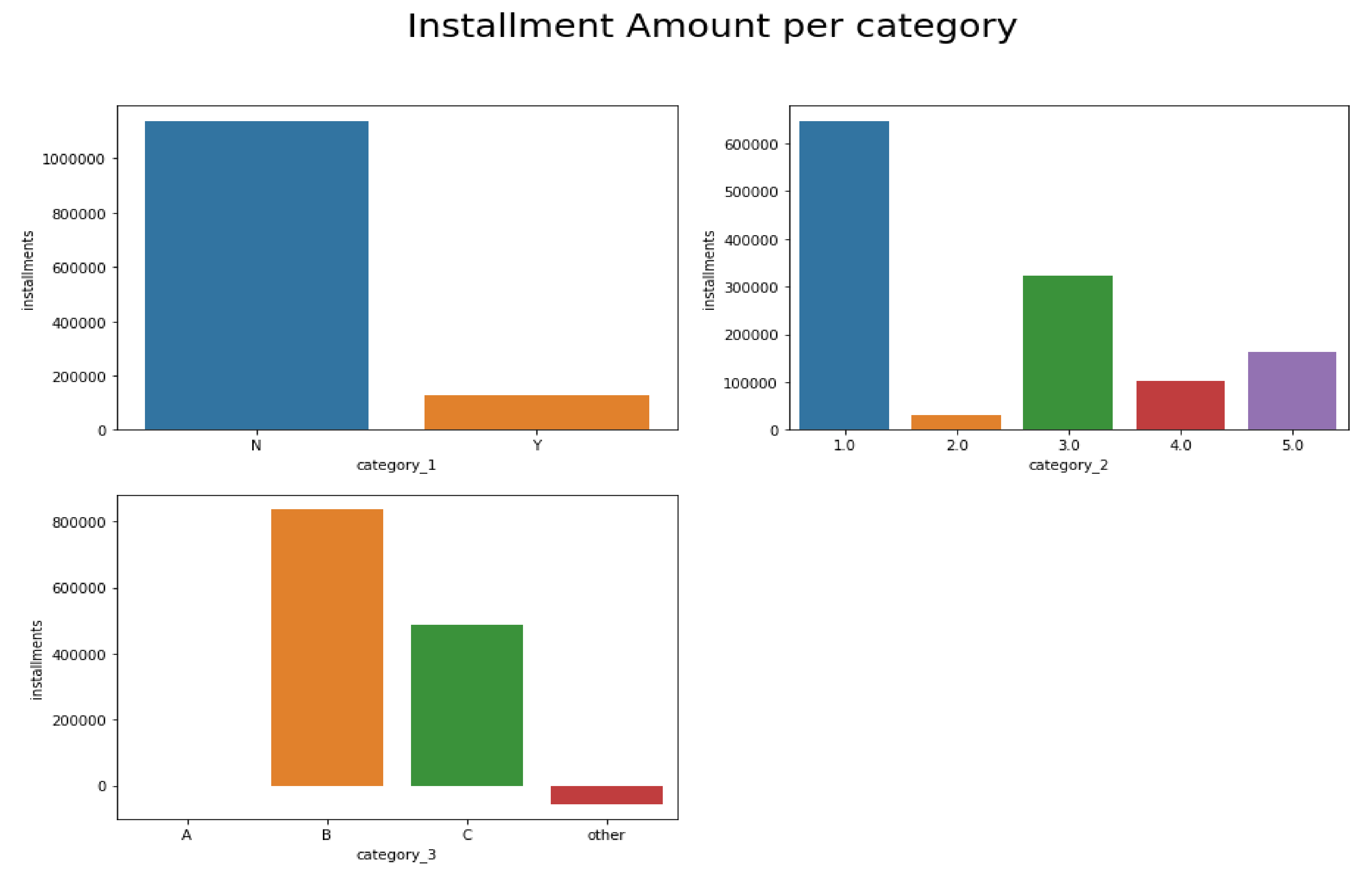


EDA – newMerchant_transactions.csv data

Most number of installments are not part of category 1.

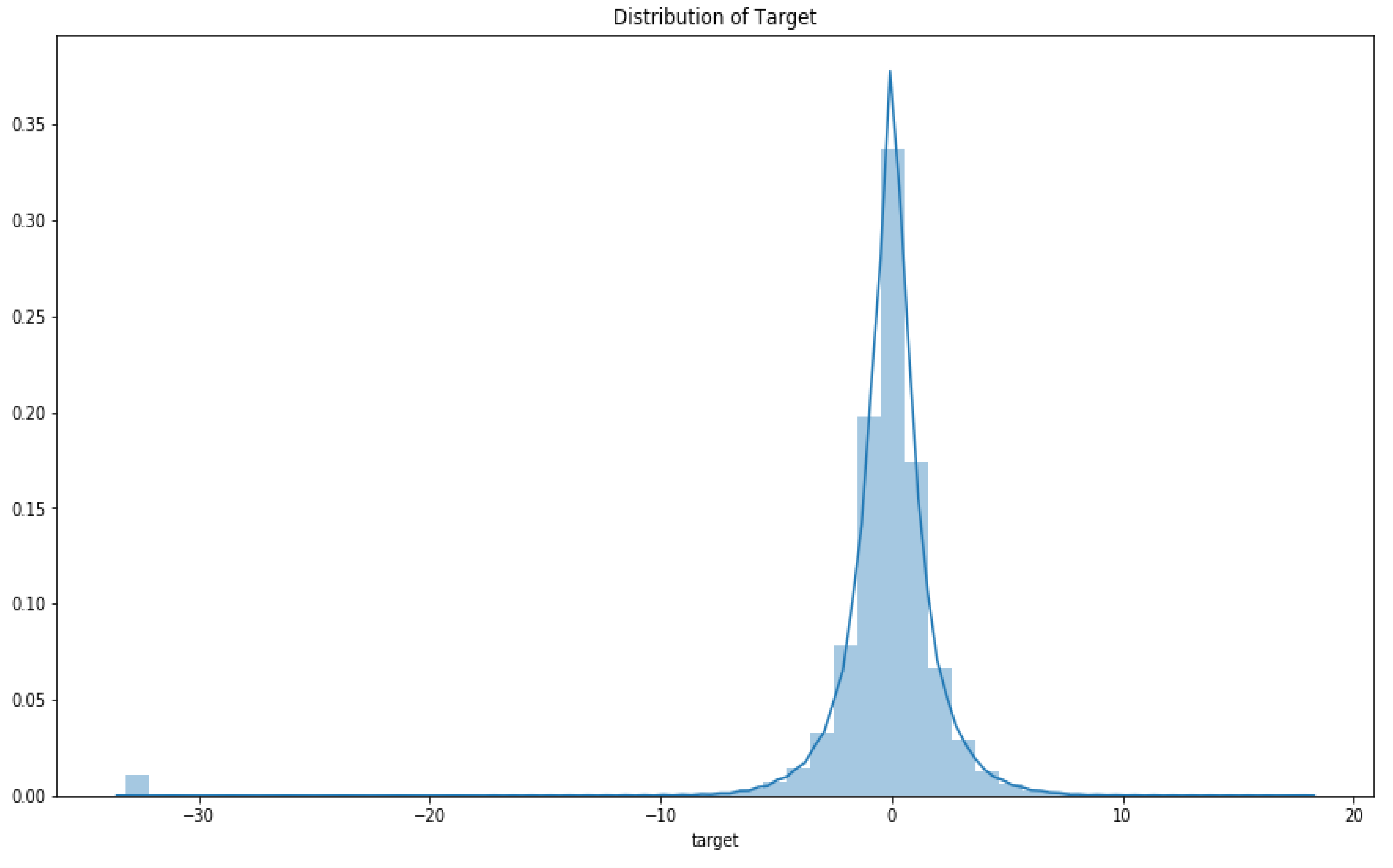
Highest number of installments in category 2 are in **1.0**.

Highest number of installments in category 3 are in **B**.



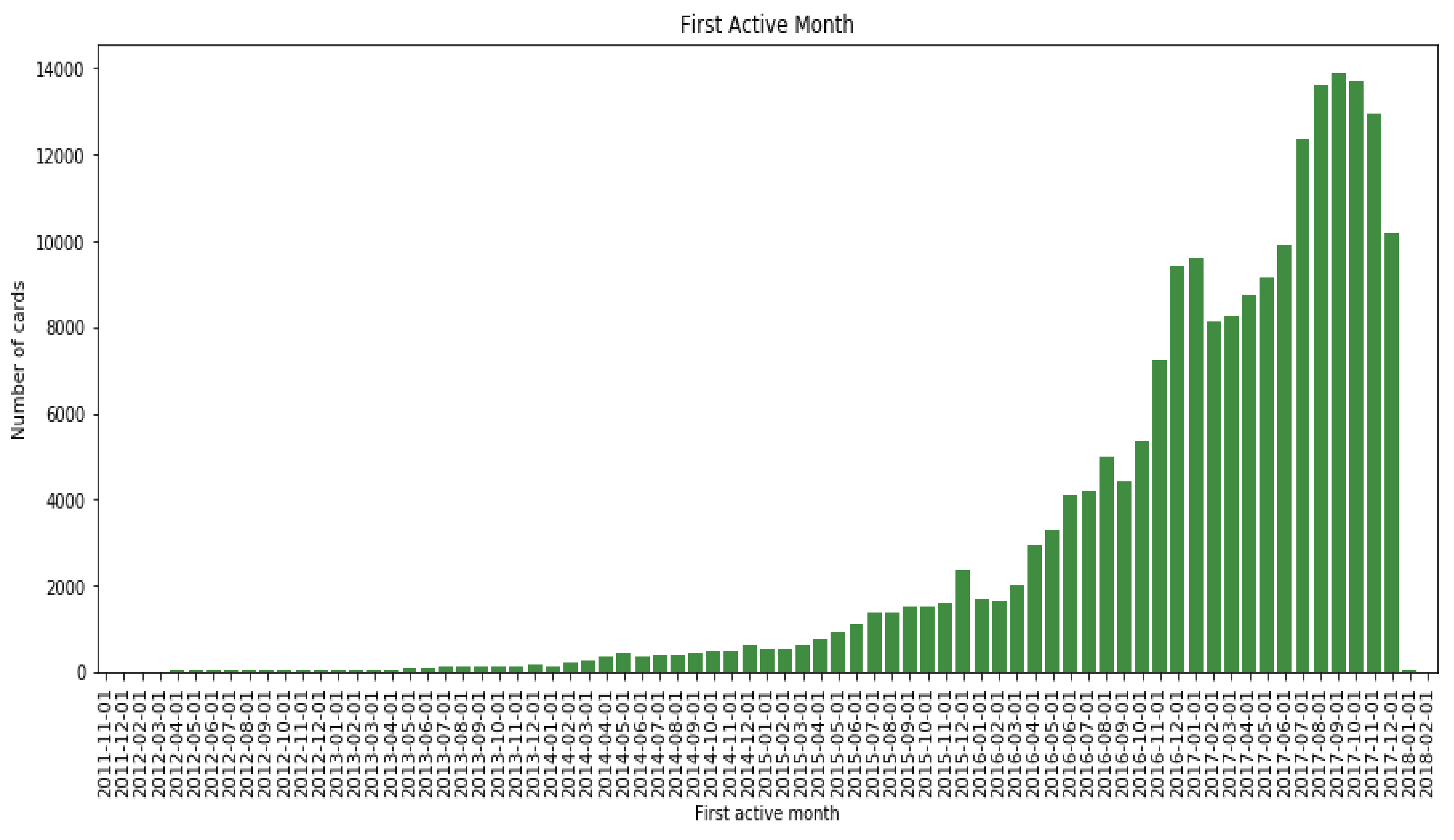
EDA – train.csv data

Target is mostly normally distributed except there is an outlier over -30 score.

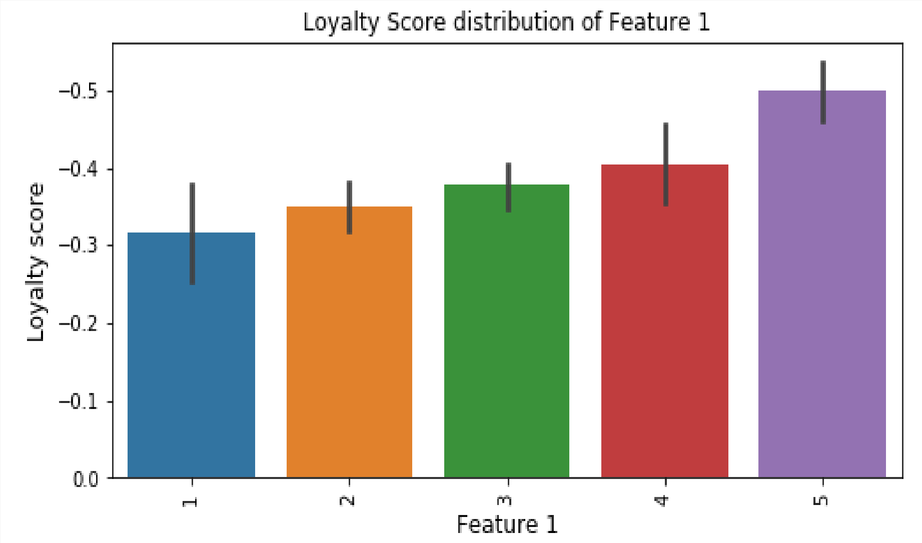


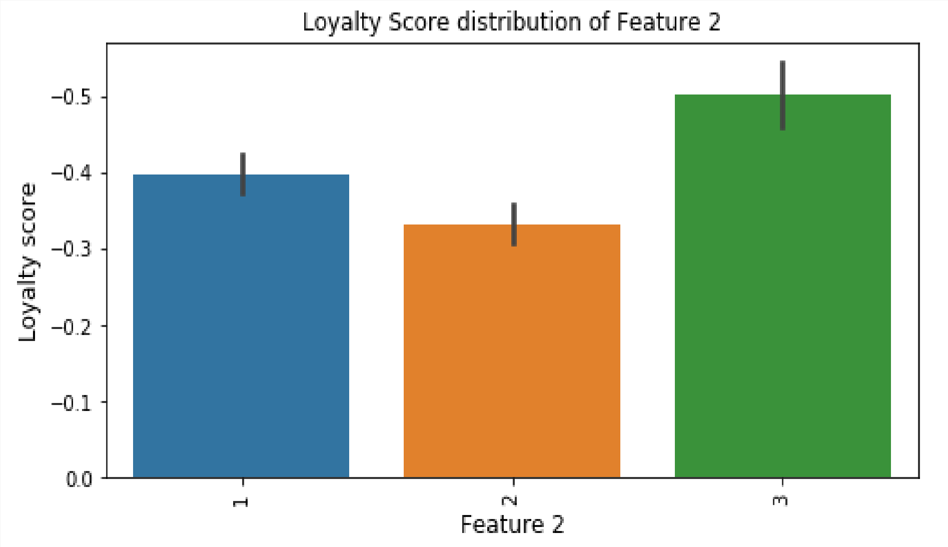
EDA – train.csv data

There is a steady increase in number of first time used cards since 2015-Jul-01.

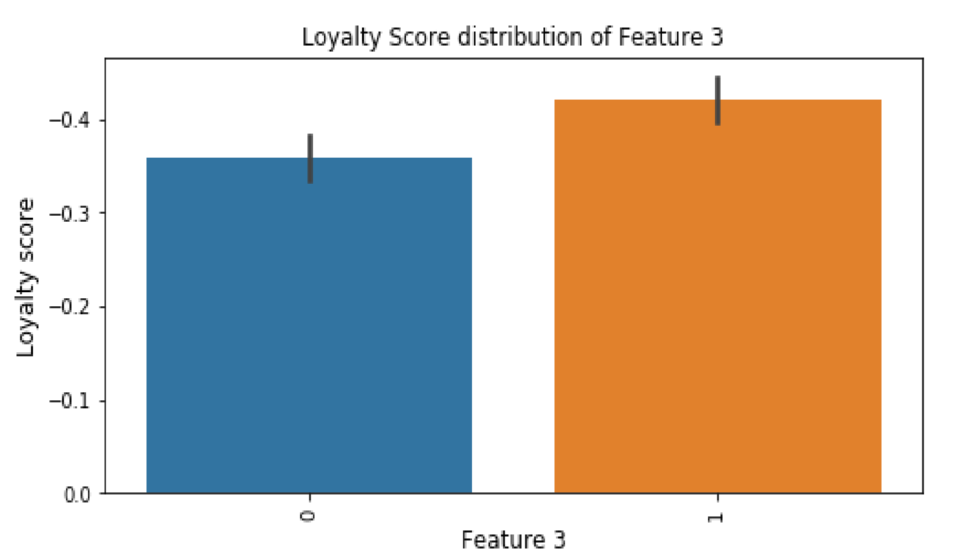


EDA – train.csv data





Loyalty score is balanced distributed across feature_1, feature_2 and feature_3.



EDA - Findings

Merchant transactions Data

- There is strong corelation numerical_1 and numerical_2 feature.
- There is a correlation between avg_sales and avg_purchases of 3, 6 an 12 month.
- Merchant category ID 705 has most sales with 9% sales
- City ID -1 has over 100000 transactions and amounts to 31% of transactions
- Subsector ID 27 has over 50000 transactions and amounts to 15% of transactions

- Percentage of sales in each Category
  - 98% of the transactions does not belong to category 1
  - 48 % of category 2 transactions are in 1.0
  - 71 of the transactions does not belong to category 4
- Purchase and Sales Range
  - 53% of sales and transactions are in E range
- Quantity of active months in a year
- December is most active sales month of the year

EDA - Findings

Historical transactions Data

- There seems to be no correlation between data
- Subsector ID 33 has over 5000000 transactions and amounts to 19% of transactions
- City ID 33 has over 4000000 transactions and amounts to 16% of transactions
- March has most purchases per month.
- January has most installments per month

- Percentage of sales in each Category
    - 92% of the transactions does not belong to category 1
    - 52 % of category 2 transactions are in 1.0
    - 53 of category 3 transactions are in A

EDA - Findings

New Merchant transactions Data

- There is a correlation between installments and purchase_amount.
- Subsector ID 37 has over 340053 transactions and amounts to 17% of transactions
- City ID 69 has 328916 transactions and amounts to 17% of transactions

- Percentage of sales in each Category
  - 97% of the transactions does not belong to category 1
  - 54 % of category 2 transactions are in 1.0
  - 47 of category 3 transactions are in A
- March has most installments per month.
- March has least purchase per month and there is constant purchases from May to December.

Feature engineering and Machine Learning Model

General process followed for featuring engineering is

1. One hot encoding is applied to categorical features to **merchant.csv**, **historical_transactions.csv and new_merchant_transactions.csv.**
2. Categorical features and anonymized in **merchant.csv** are merged to **historical_transactions.csv** and **new_merchant_transactions.csv**
3. Aggregate functions (mean, count, sum, nunique) are applied to datasets **historical_transactions.csv and new_merchant_transactions.csv** by grouping by card_id.
4. Datetime features are added to aggregated Data Fames.
5. Aggregated Data Fames are merged with train and test data
6. Datetime features are added to merged **train** and **test** data frame and outlier feature is added to **train** data frame to handle outliers.
7. Training data is trained on **XGBOOST** ML algorithm
8. **RandomizedSearchCV** is used for tuning **XGBOOST** algorithm hyperparameters
9. **RMSE** is used for evaluation
10. Feature importance is generated on the trained model.

Featuring Engineering

merchant.csv –

- One hot encoding is applied to categorical features “category_4”, “category_1”, 'category_2', 'most_recent_sales_range', 'most_recent_purchases_range’.
- New date Frame with categorical and anonymized measure features is created for merging **historical_transactions.csv and new_merchant_transactions.csv,** other features are dropped as they are only informational features about merchant ID.
- Features considered for merging are **'merchant_id','numerical_1', 'numerical_2', 'category_2_0.0’, 'category_2_1.0', 'category_2_2.0', 'category_2_3.0', 'category_2_4.0', 'category_2_5.0', 'category_4', 'category_1’**

historical_transactions.csv and new_merchant_transactions.csv –

- Categorical and anonymized measure features are merged with datasets **historical_transactions** and **new_merchant_transactions**
- Rows with **NaN** values are dropped after merging datasets as rows with **NaN** values are around **1%**
- **Category_2/category_3_purchaseAmt_mean** is added by grouping **category_2/category_3** and aggregating by mean over **purchase_amount** feature.

Featuring Engineering

- **One hot encoding** is applied to categorical features ‘authorized_flag’, ‘category_1’, 'category_2', 'category_3’.
- Following aggregration functions is applied by grouping **historical_transactions** and **new_merchant_transactions** by card_id
  - **'authorized_flag': ['sum', 'mean'],**
  - **'category_1':['sum', 'mean'],**
  - **'category_2_1.0': 'mean',**
  - **'category_2_2.0': 'mean',**
  - **'category_2_3.0': 'mean',**
  - **'category_2_4.0': 'mean',**
  - **‘category_2_5.0': 'mean’,**
  - **'category_3_A': 'mean',**
  - **'category_3_B': 'mean',**
  - **'category_3_C': 'mean',**
  - **'category_3_other': 'mean',**

Featuring Engineering

- ‘state_id': 'nunique',
- 'city_id': 'nunique',
- 'purchase_amount': ['sum', 'mean', 'count', 'max', 'min', 'std'],
- 'installments': ['sum', 'mean', 'max', 'min', 'std'],
- 'purchase_date': ['min', 'max'],
- 'month_lag': ['mean', 'max', 'min', 'std'],
- 'card_id': ['count'],
- 'month_diff': ['mean'],
- 'weekend' : ['sum', 'mean'],
- 'month': 'nunique',
- 'hour': 'nunique’,
- 'weekofyear': 'nunique',
- 'dayofweek': 'nunique'

Featuring Engineering

- 'year': 'nunique',
- 'subsector_id': 'nunique',
- 'merchant_id': 'nunique',
- 'merchant_category_id': 'nunique',
- 'category_2_purchaseAmt_mean' : 'mean',
- 'category_3_purchaseAmt_mean' : 'mean',
- 'merchDF_numerical_1': ['mean', 'sum'],
- 'merchDF_numerical_2': ['mean', 'sum’],
- 'merchDF_category_2_0.0': 'mean',
- 'merchDF_category_2_1.0':'mean',
- 'merchDF_category_2_2.0':'mean',
- 'merchDF_category_2_3.0':'mean',
- 'merchDF_category_2_4.0':'mean',

Featuring Engineering

- **'merchDF_category_2_5.0':'mean',**
- **'merchDF_category_4': 'mean',**
- **'merchDF_category_1': 'mean'**

- Datetime features are added to aggregated data frame
  - **purchase_date_diff ---- purchase_date_max - purchase_date_min**
  - **purchase_date_average ----- purchase_date_diff/card_id_count**
  - **purchase_date_tillToday ----- Today's date - purchase_date_max**

train and test dataset –

- Aggregate Data frames generated from **historical_transactions** and **new_merchant_transactions** are merged to **train** and **test** dataset
- Datetime features are added from **first_active_month**
  - **Day of the week**
  - **Week of year**
  - **month**

Featuring Engineering

- **elapsed_time** - Time elapsed from first active month
- **histDF_first_buy** - number of days from the first buy in historical transactions dataset
- **newMerchDF_hist_first_buy** - number of days from the first buy in new merchant transactions dataset
- Convert datetime features ‘**histDF_purchase_date_max**’, '**histDF_purchase_date_min**’, '**newMerchDF_purchase_date_max**’, '**newMerchDF_purchase_date_min**’ to numeric
- **card_id_total** - card Id count total (count of card ID in **historical_transactions** and **new_merchant_transactions)**
- Outlier feature is added to **train** dataset
- Outlier feature is aggregated to mean by grouping on feature_1/2/3. Aggregated data frame is mapped to feature_1/2/3 in **test** and **train**

Machine Learning Model

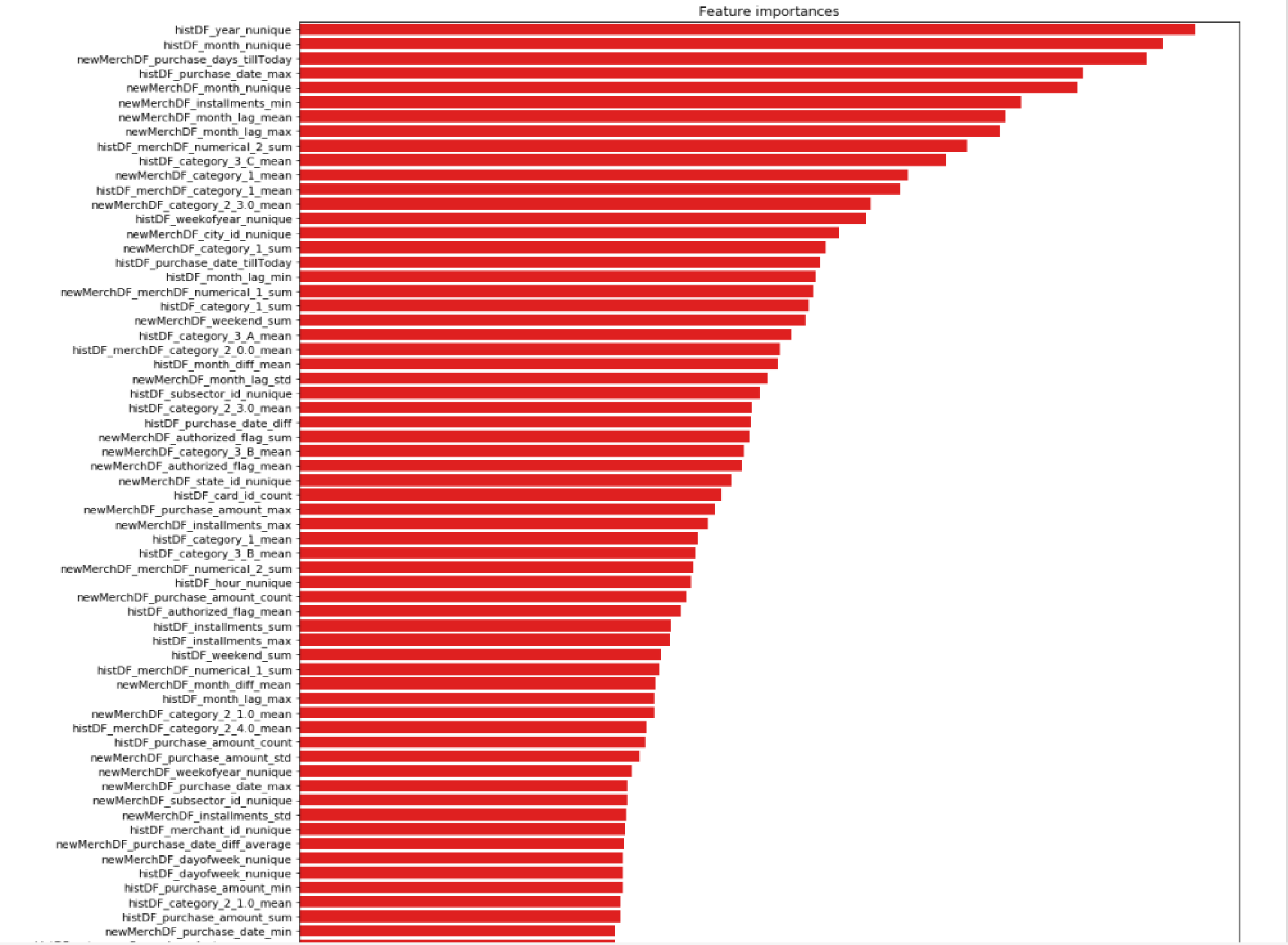
- Feature List is generated excluding features
    - card_id
    - first_active_month
    - target
    - merchant_id
    - outliers
- In this model hyperparameters are tuned using **RandomizedSearchCV**. Hyperparameters found in **RandomizedSearchCV** are used to for learning **XGBClassifier**.
- **Hyperparameters Tuning**
  - **n_estimators** - number of trees to grow. Larger the tree size better the model, but more numbers of trees can be computationally expensive and affects the performance of the model n_estimators = [4, 8, 16, 32, 64, 100, 200]
  - **max_depth** - depth of the tree, the more splits it has and it captures more information about the data. But as the tree gets very deep, it might lead to overfitting max_depth = [4, 8, 10, 12, 16, 32, 64]

Machine Learning Model

- **Hyperparameters Tuning continued….**
  - **min_child_weight** - Minimum sum of instance weight needed in a child. min_child_weight = [2, 4, 6, 8, 10, 12, 16, 32, 64]
  - **gamma** - [0.1, 0.2, 0.3, 0.4, 0.5]
  - **colsample_bytree** - Subsample ratio of columns when constructing each tree. colsample_bytree = [0.2, 0.4, 0.6, 0.8]
  - **colsample_bylevel** - Subsample ratio of columns for each split, in each level colsample_bylevel = [0.2, 0.4, 0.6, 0.8]
- **Tuned Hyperparameters** are **n_estimators** - 100, **max_depth** - 8, **min_child_weight** - 32, **gamma** – 0.2, **colsample_bytree**- 0.2, **colsample_bylevel** – 0.6
- **RMSE** is calculated on target and values predicted from train dataset, which is **3.38569**

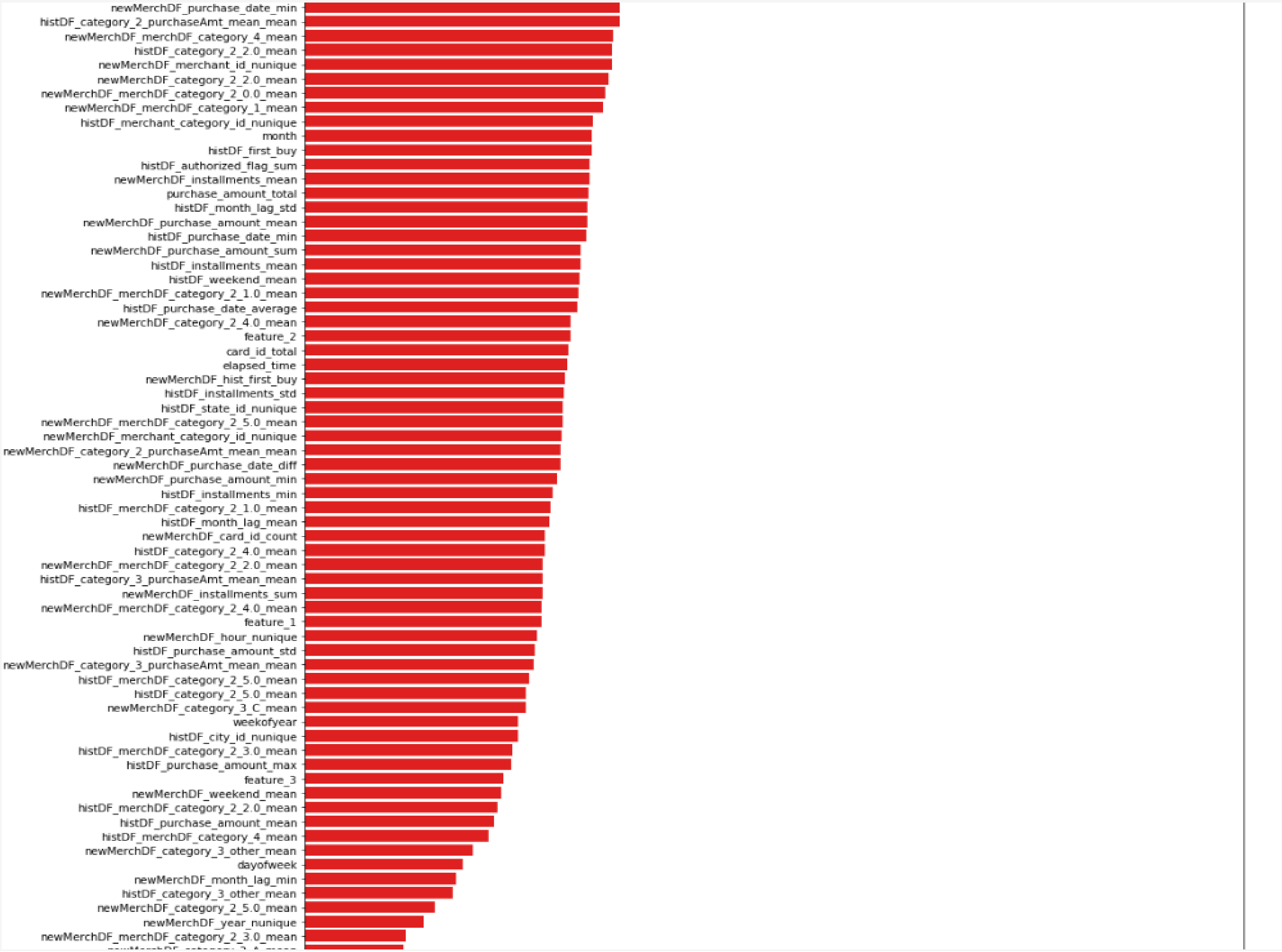
Machine Learning Model

Feature Importance



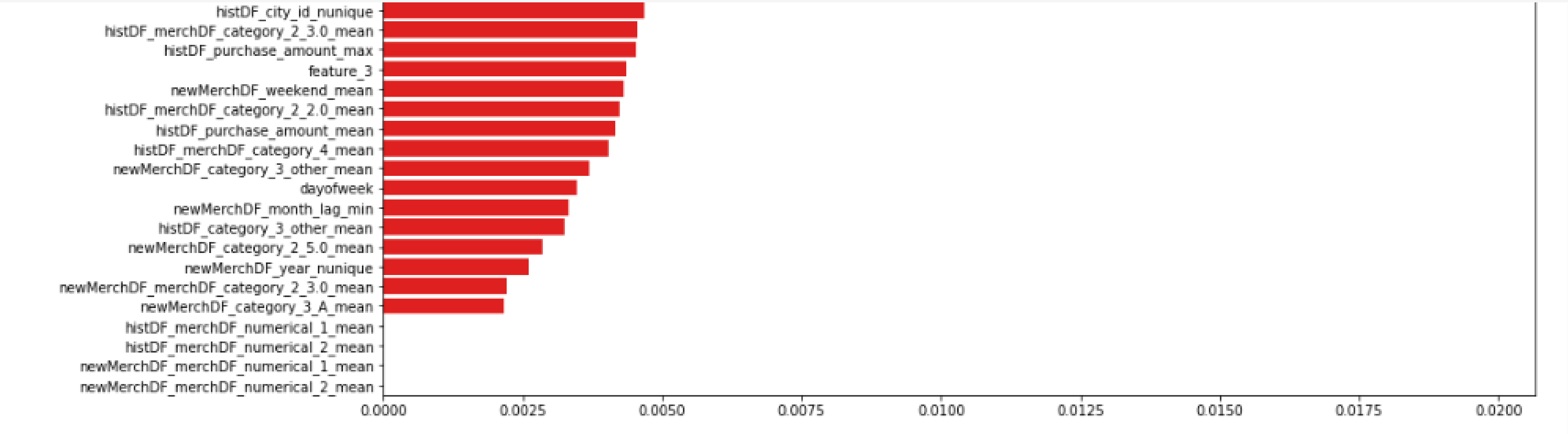
Machine Learning Model

Feature Importance



Machine Learning Model

Feature Importance



Conclusion

Top five features impacting model impacting loyalty score

1. **histDF_year_nunique** -- number of unique year in a card ID transactions in Historical transactions dataset
   - **Card ID with more number of unique year means the card is actively used and hence has most impact on loyality score. As number of unique year decreases which means card ID is less active.**
2. **histDF_month_nunique** -- number of unique months in a card ID transactions in Historical transactions dataset
   - **Card ID with more number of unique month means the card is actively used and hence has most impact on loyalty score.**
3. **newMerchDF_purchase_days_tillToday** -- number of purchase days from last purchase date in new merchant transactions dataset
   - **As the number of days since last purchase made impacts loyalty score**
4. **histDF_purchase_date_max** -- Most recent purchase date of card ID in Historical transactions dataset
   - **As the number of days since last purchase made impacts loyalty score**
5. **newMerchDF_month_nunique** -- number of unique months in a card ID transactions in new merchant transactions dataset
   - **Card ID with more number of unique month means the card is actively used and hence has most impact on loyality score.**

Recommendation -

1. If the loyalty score of a card is low, then discount in top important category can sent to card holder.
2. Loyalty score can be monitored monthly and if the loyalty score decrease then a discount in most important category can set to card holder.