Elo Merchant Category Recommendation
- Help understand customer loyalty

Xinyu, Yao
Xiaotian, Huang
Jingya, Gao

Introduction

- **Elo**
  - One of the largest payment brands in Brazil
  - Built partnerships with merchants
  - Offer promotions and discounts to cardholders
  - Do these promotions work for both consumer and merchant?
  - Are the promotions what customers needed?
  - Do merchants see repeat business?
  - Serve the most relevant opportunities to cardholders, by uncovering signal in customer loyalty.
  - Predict the target-loyalty score for each card_id

Description of the data set

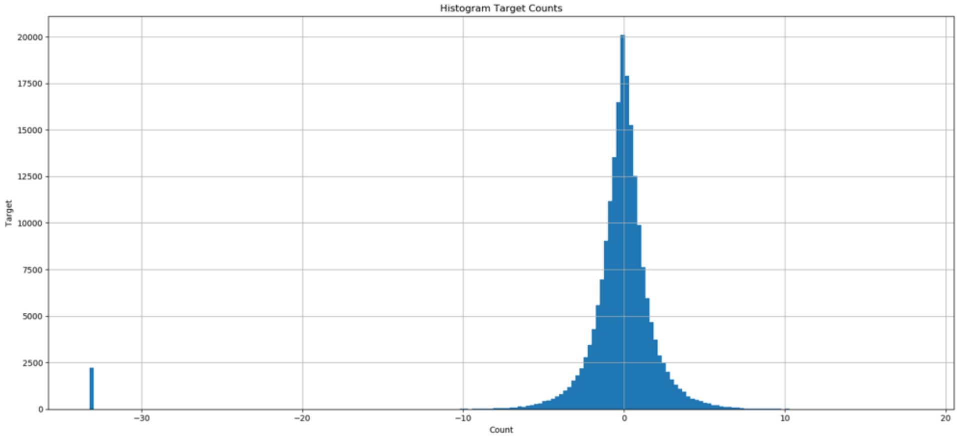
- train.csv - the training set
- test.csv - the test set
- historical_transactions.csv - up to 3 months historical transactions for each card_id
- new_merchant_transactions.csv - two months transactions that card_id made at merchant_ids that were *not visited in the historical data*.
- merchants.csv - additional information about all merchants / merchant_ids in the dataset.

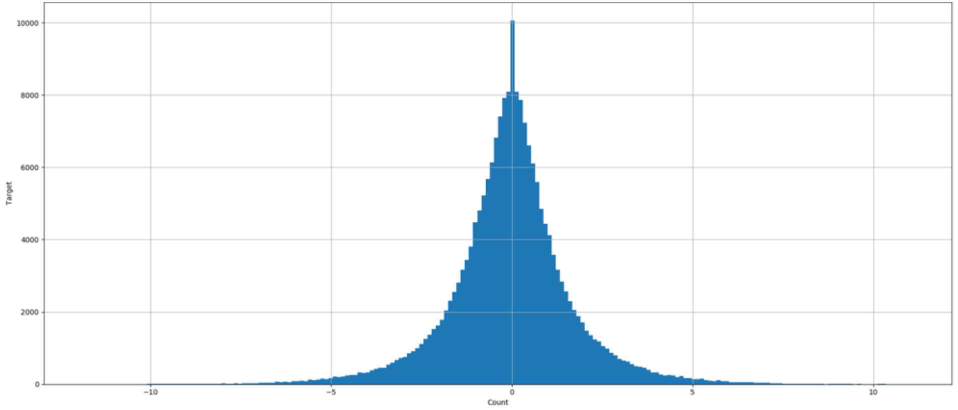
Algorithms used

- Cleaning algorithm: fillna, drop outliers
- Data mining:
  - Linear Regression
  - Decision Tree
  - Random Forest
  - Naive Bayes
  - K-Nearest Neighbor
  - Support Vector Machine
  - K-Mean
  - AGNES
  - DBSCAN

Pre-processing and Feature Engineering

- Check missing values of all tables
- Check unique values of columns that has na and fillna with values different from the unique values existed in those columns.
- Calculated z-score of target and drop outliers whose |Z-score| > 3
- Define functions that aggregate the info by grouping on card_id and month_lag
- Merge all the dataframes and then to_csv





Linear Regression

$$y = w_0 + w_1 x_1 + \cdots + w_j x_j + \cdots + w_n x_n$$

$$J(w) = \frac{1}{m}\sum^{m}(y_i - \hat{y}_i)^2 + c\sum_{j=1}^{n} w_j^2$$

$$w_j = w_j - \eta * \nabla J(w) = w_j + 2\eta\left(\frac{1}{m}\sum^{m}(y_i - \hat{y}_i)x_j - c w_j\right)$$

Best_score (*neg_mean_squared_error*): -0.8982704791215965

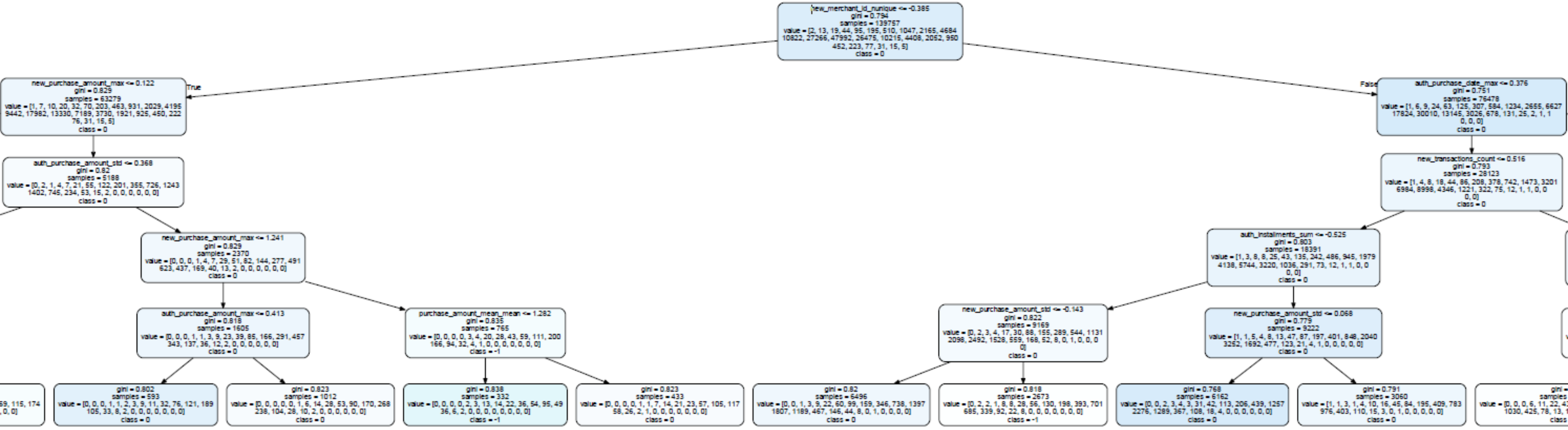
best_params: {'estimator__eta': 0.1}

Mean_squared_error: 2.65

r2_score: 0.09

Decision Tree

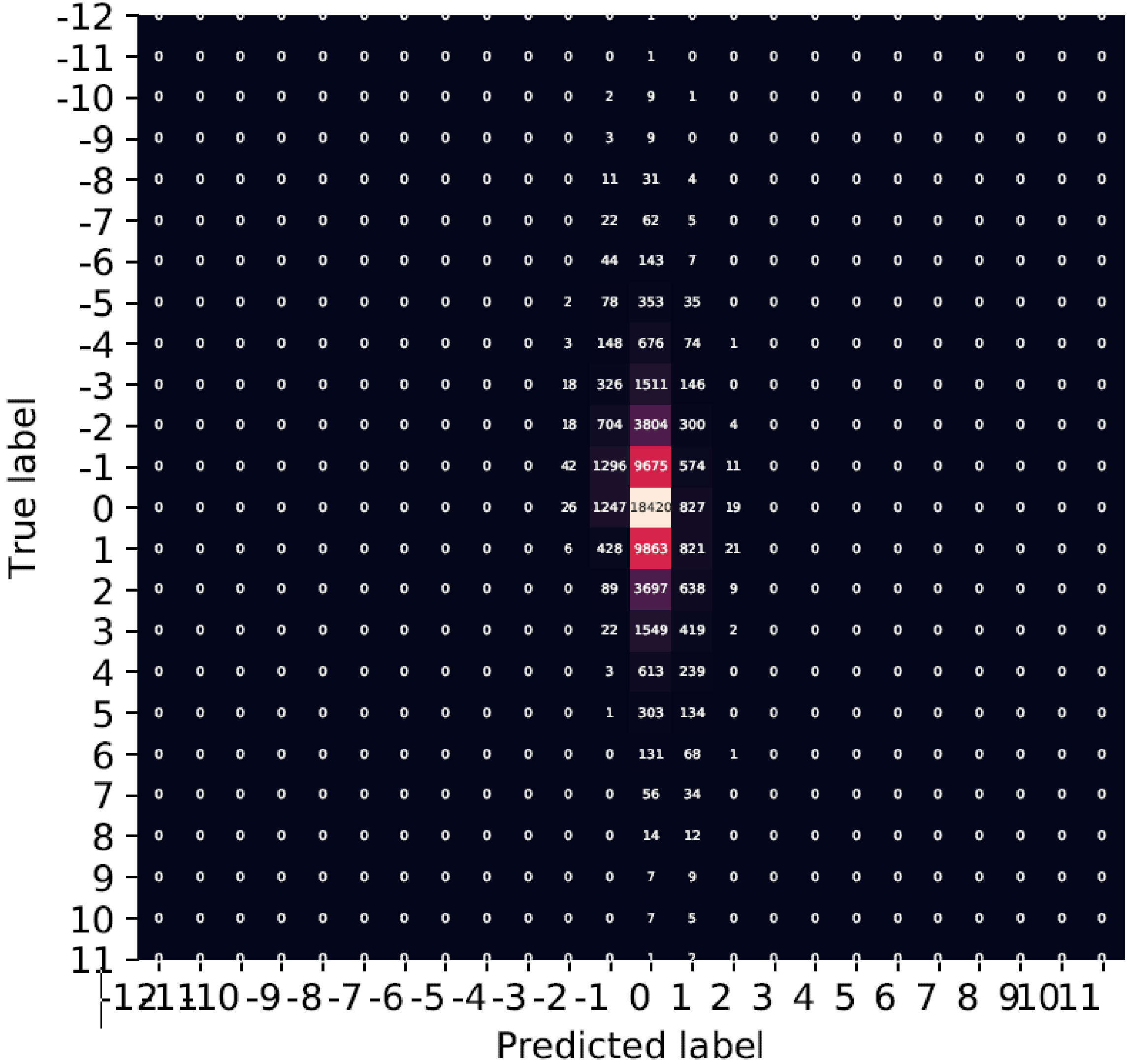
'max_depth': 6, 'min_samples_leaf': 1, 'min_samples_split': 2



Decision Tree

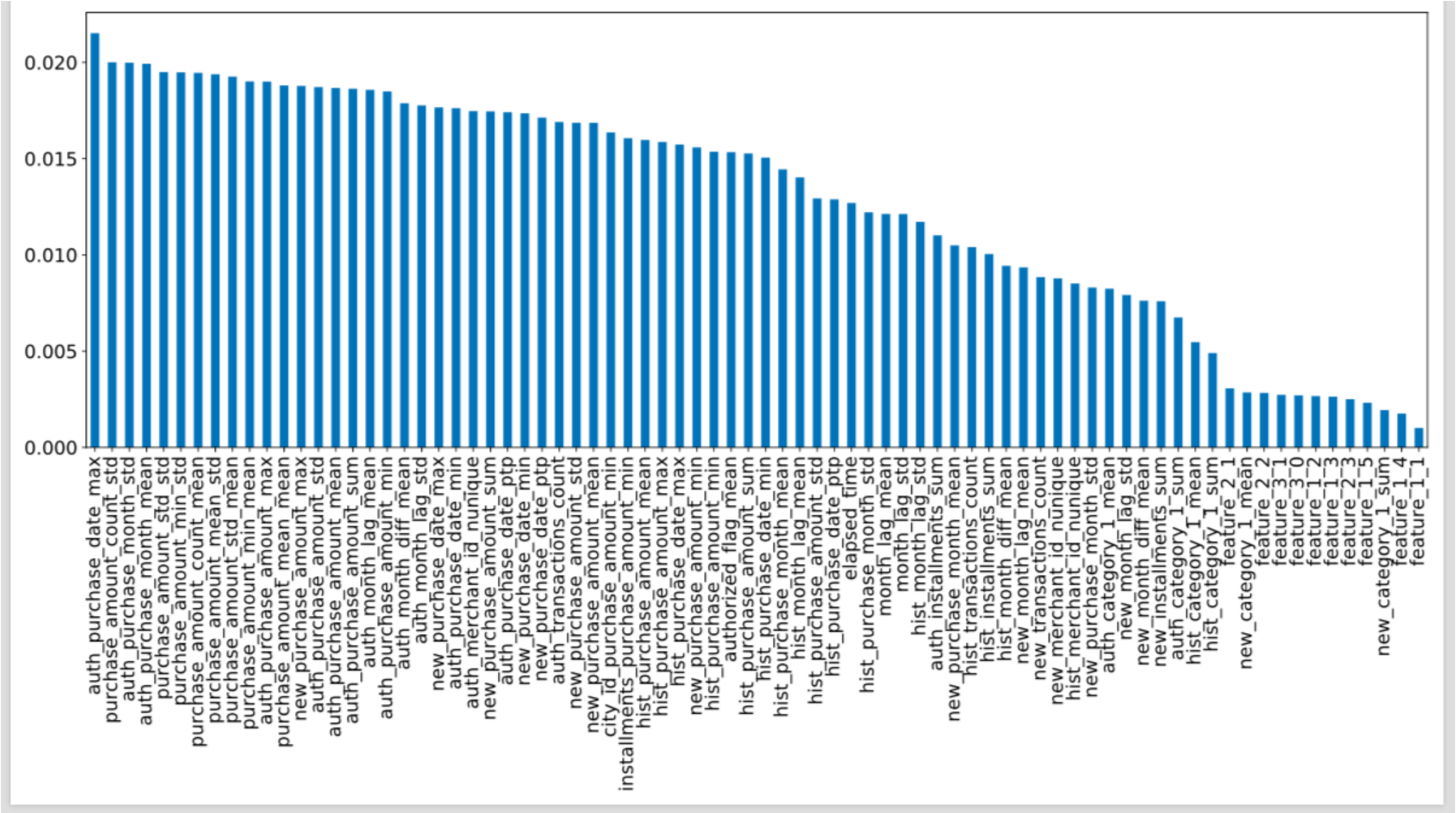
Accuracy: 34.33

MSE: 2.87



Random Forest

Features importance



Split data

To transform the variable from continuous variable into categorical variable, I divided integrals ranging from -12 to 12 into

- 24 groups with an interval of 1

```
bins = np.arange(-12.5, 12.5, 1)
names = np.arange(-12, 12, 1)
data['new_target'] = pd.cut(data['target'], bins, labels=names)
```

- 6 groups with an interval of 4;

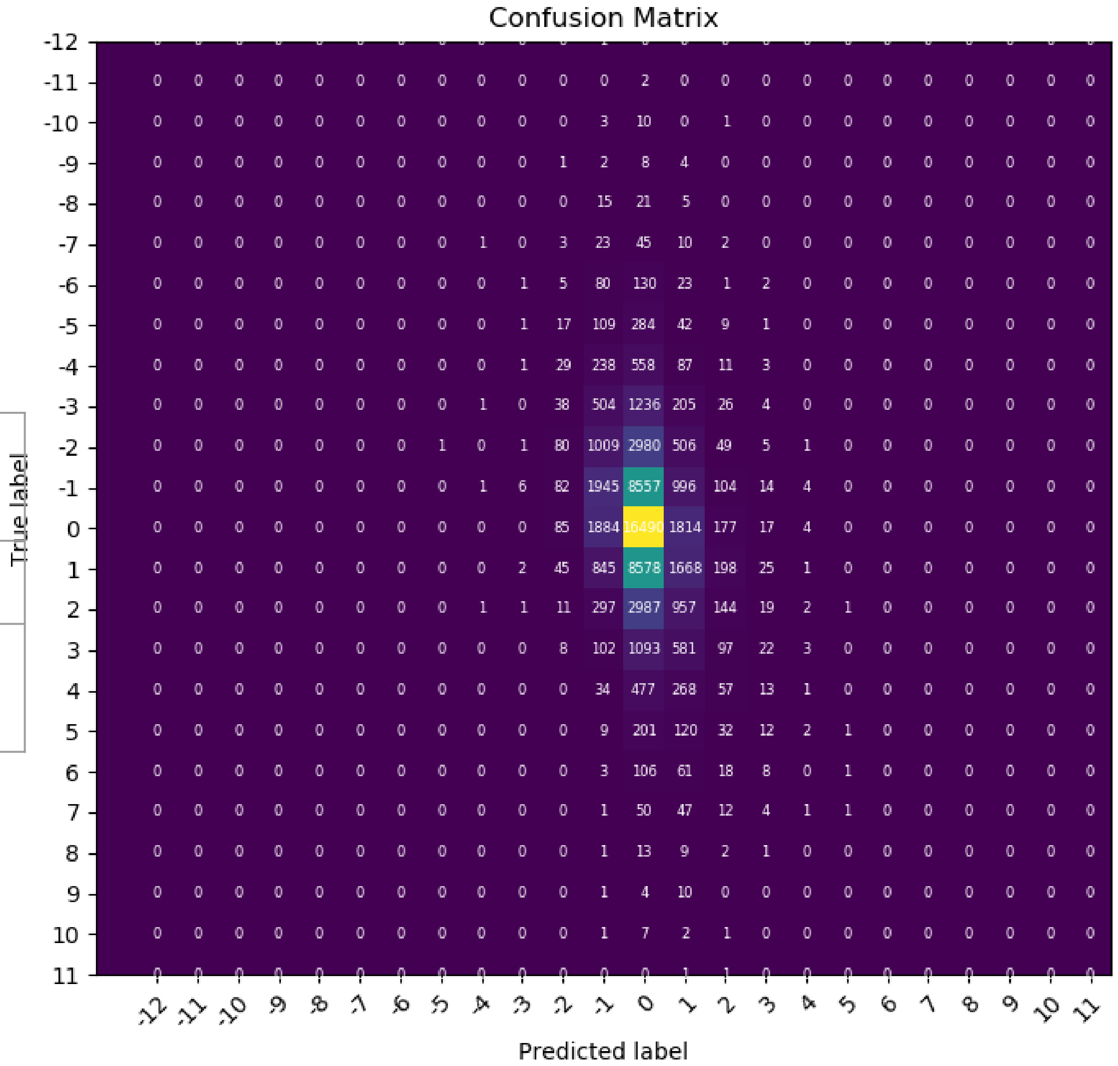
```
bins = np.arange(-12.5, 12.5, 4)
names = ['[-12, -8)', '[-8, -4)', '[-4, 0)', '[0, 4)', '[4, 8)', '[8, 12)']
data['new_target'] = pd.cut(data['target'], bins, labels=names)
```

- 3 groups with an interval of 8.

```
bins = np.arange(-12.5, 12.5, 8)
names = ['[-12, -4)', '[-4, 4)', '[4, 12)']
data['new_target'] = pd.cut(data['target'], bins, labels=names)
```

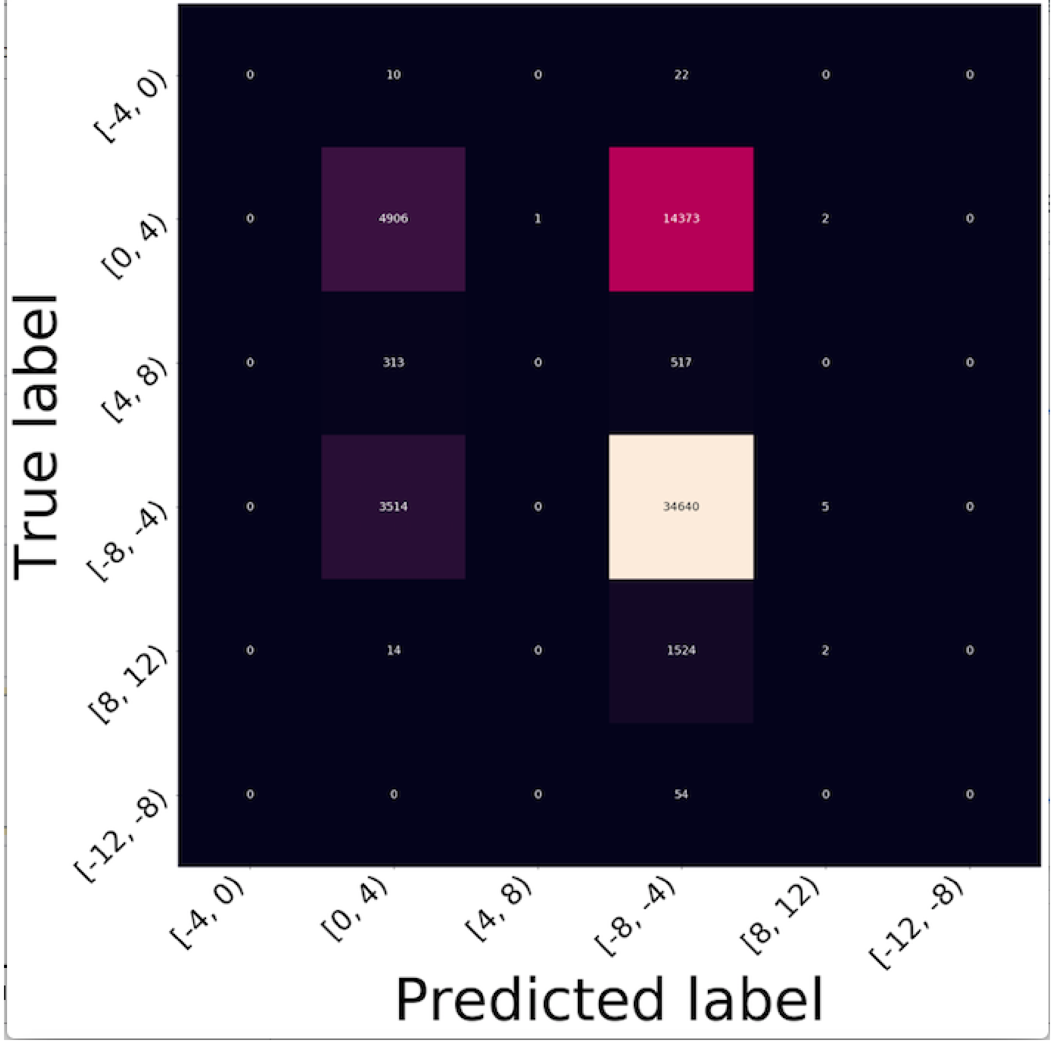
Random Forest

	Ranging by interval 1
Accuracy:	33.93%
Mean Squared Error :	2.89



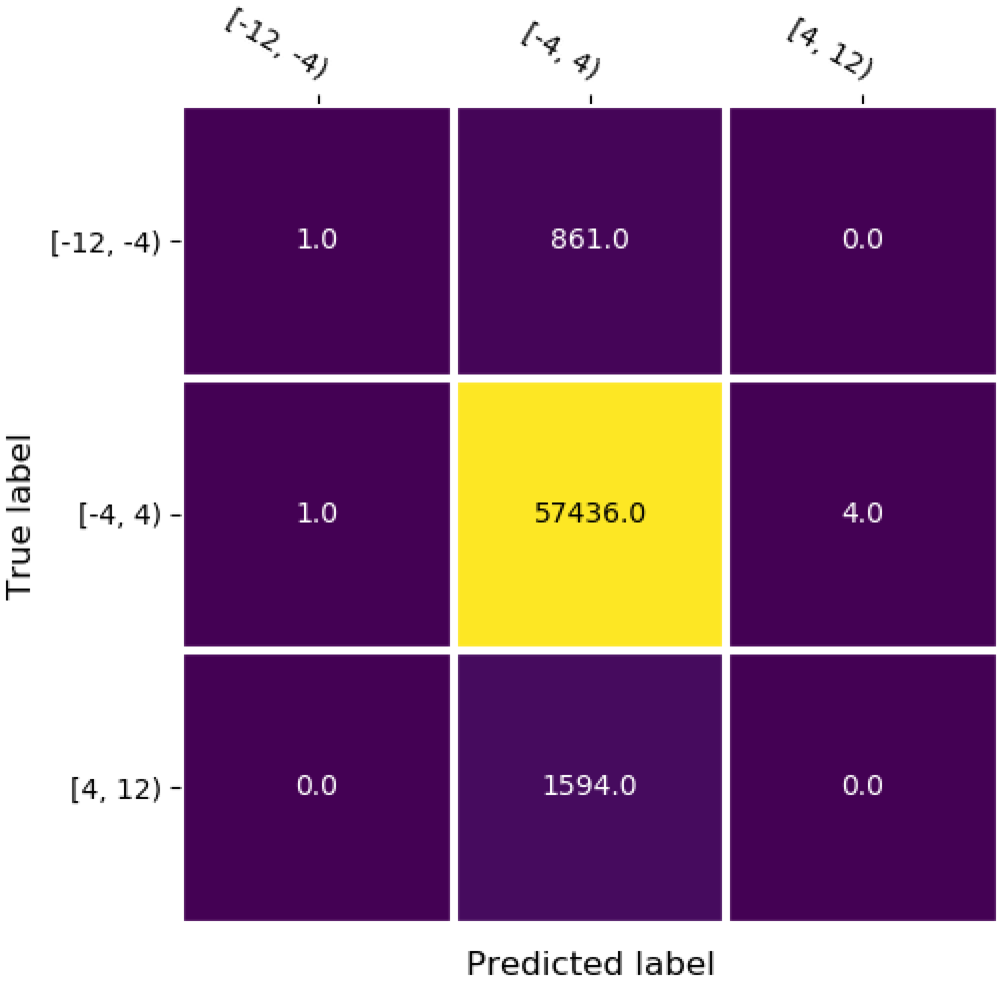
Random Forest

	Ranging by interval 4
Accuracy:	66.06%



Random Forest

	Ranging by interval 8
Accuracy:	95.89%

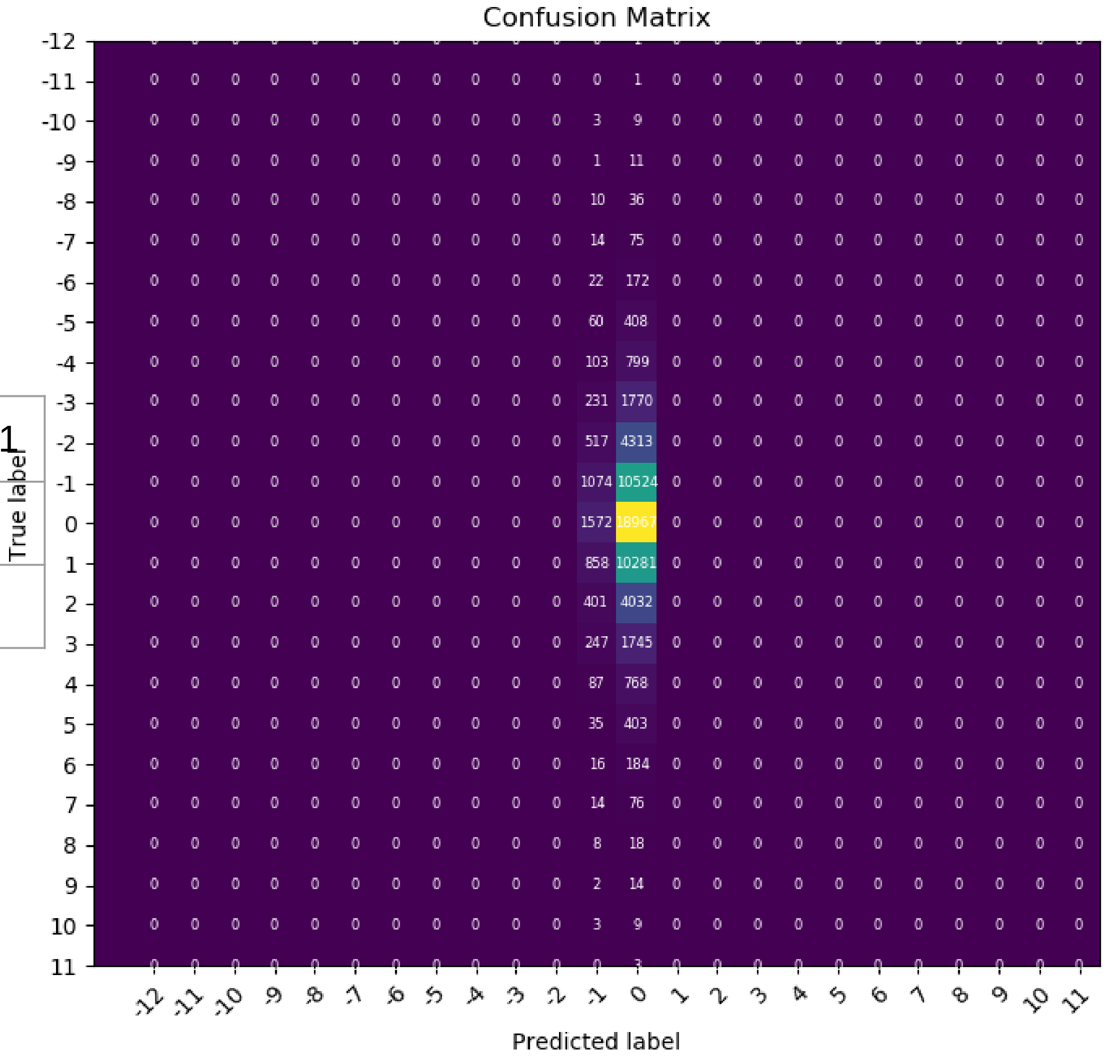


Random Forest

	Ranging by interval 1	ranging by interval 4	ranging by interval 8
Accuracy:	All features: 33.93% K features: 33.29%	All features: 66.06% K features: 65.52%	All features: 95.89% K features: 95.89%
Mean Squared Error :	All features: 2.89 K features: 2.89	All features: 5.96 K features: 5.96	All features: 2.63 K features: 2.63

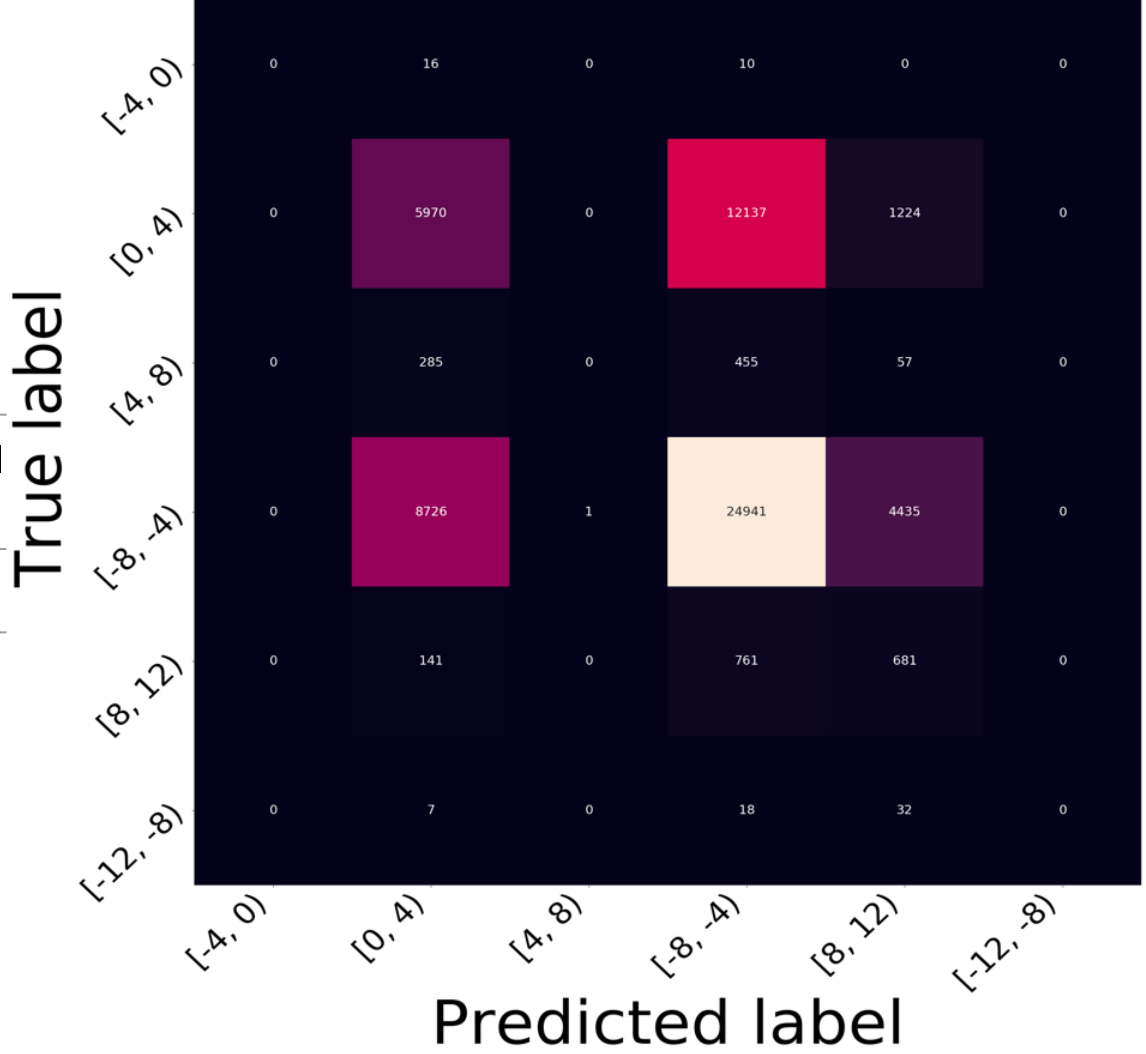
Naive Bayes

	ranging by interval 1
Accuracy:	33.46%
Mean Squared Error :	3.06



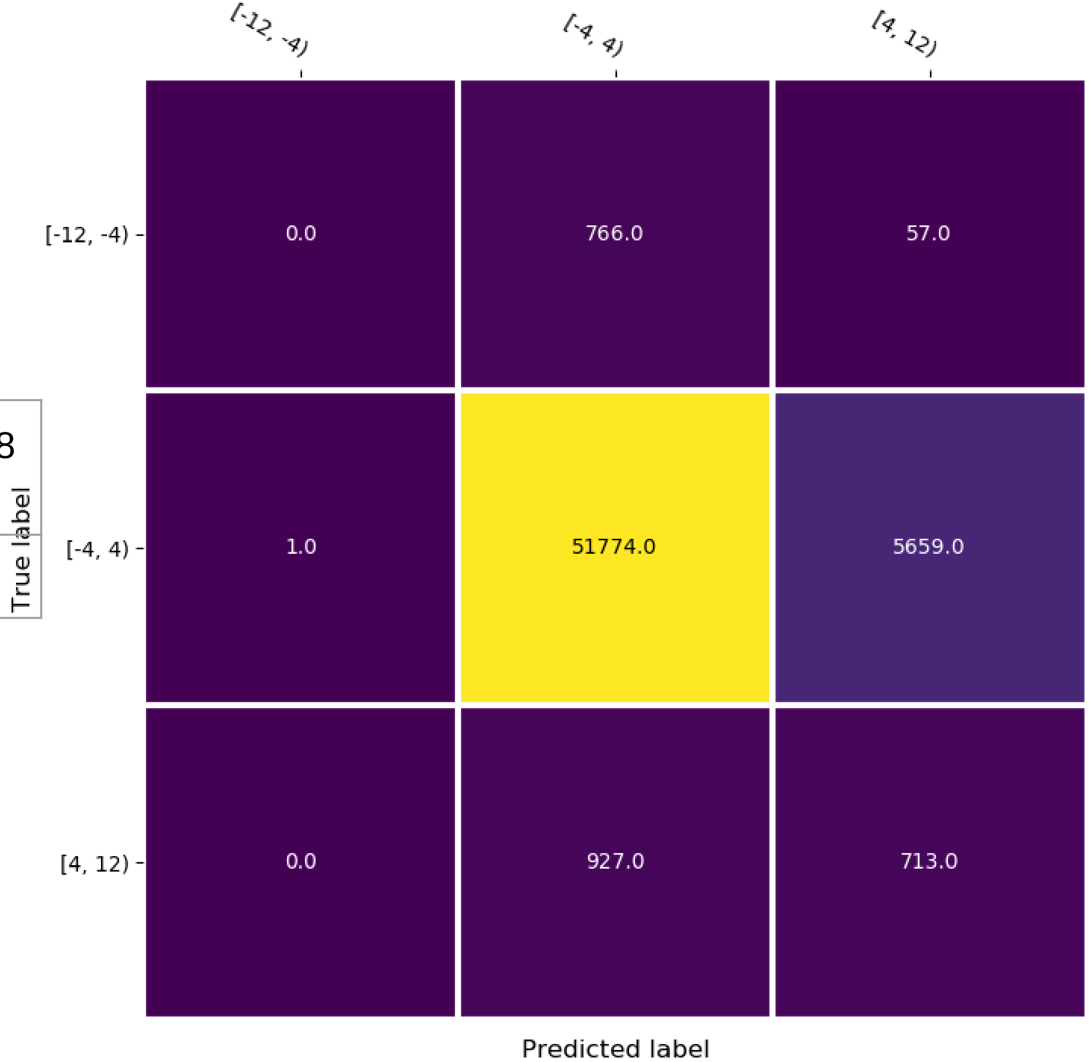
Naive Bayes

	ranging by interval
Accuracy:	52.74%



Naive Bayes

	ranging by interval 8
Accuracy:	87.63%

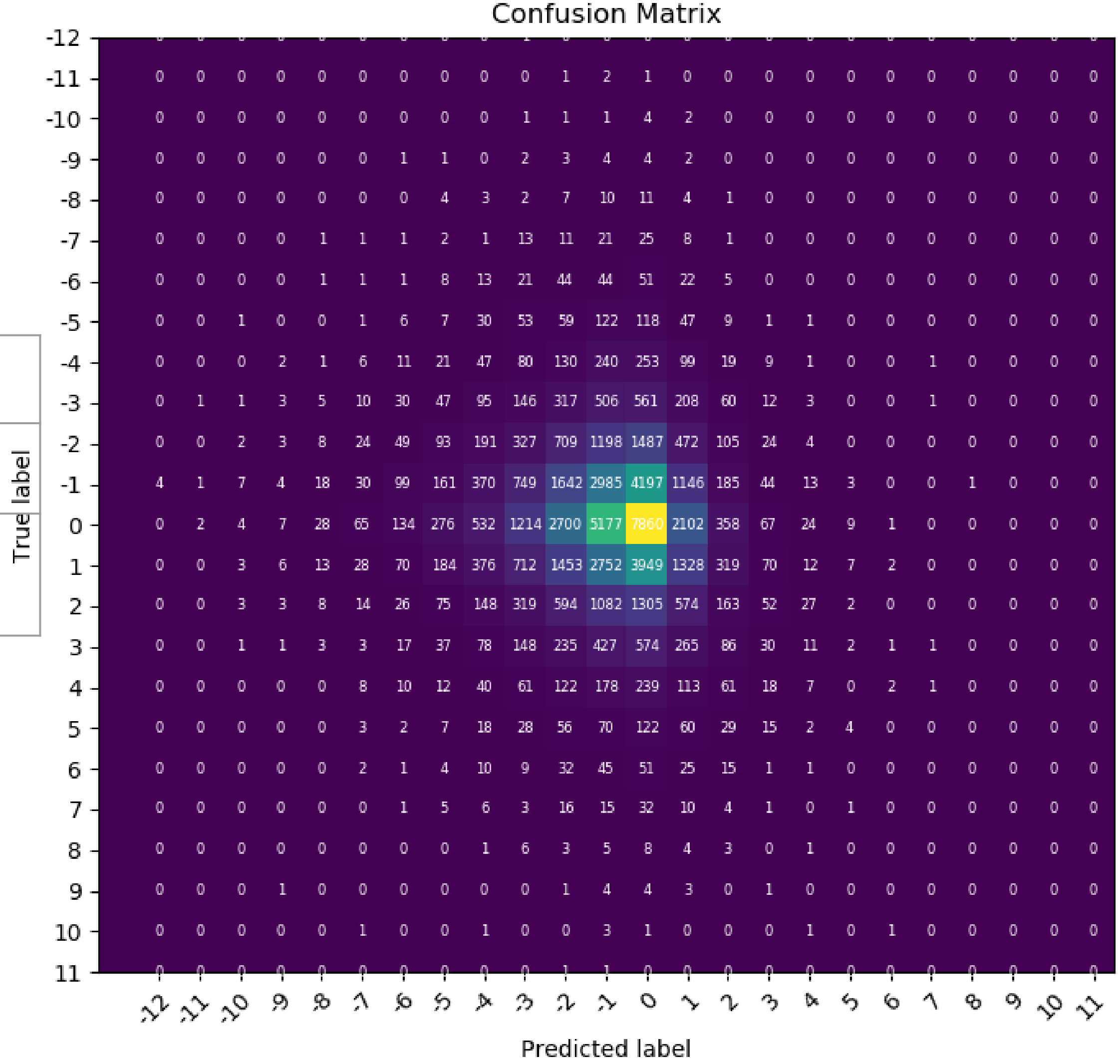


Naive Bayes

	ranging by interval 1	ranging by interval 4	ranging by interval 8
Accuracy:	33.46%	52.74%	87.63%
Mean Squared Error :	3.06		

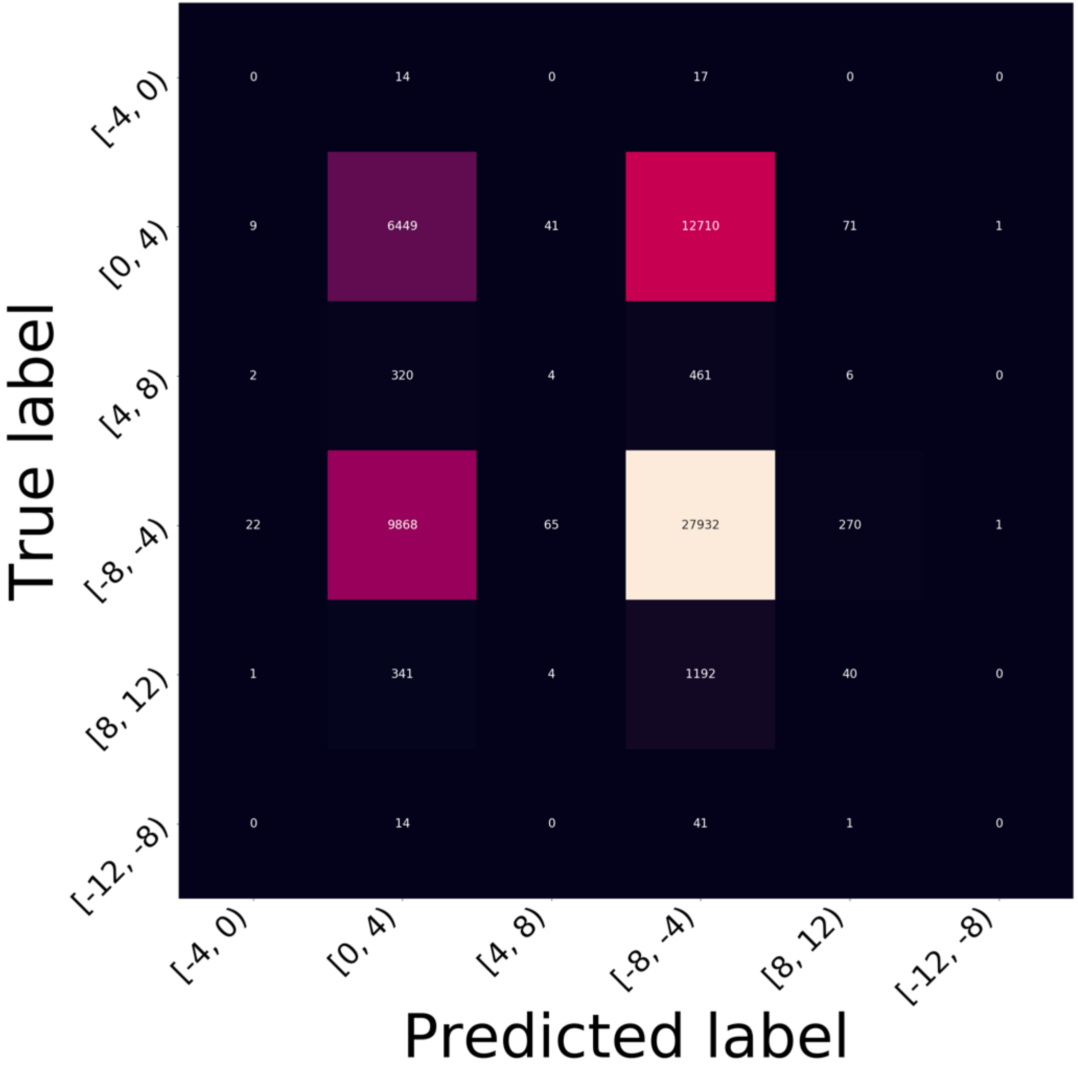
KNN

	ranging by interval 1
Accuracy:	22.18%
Mean Squared Error :	5.84



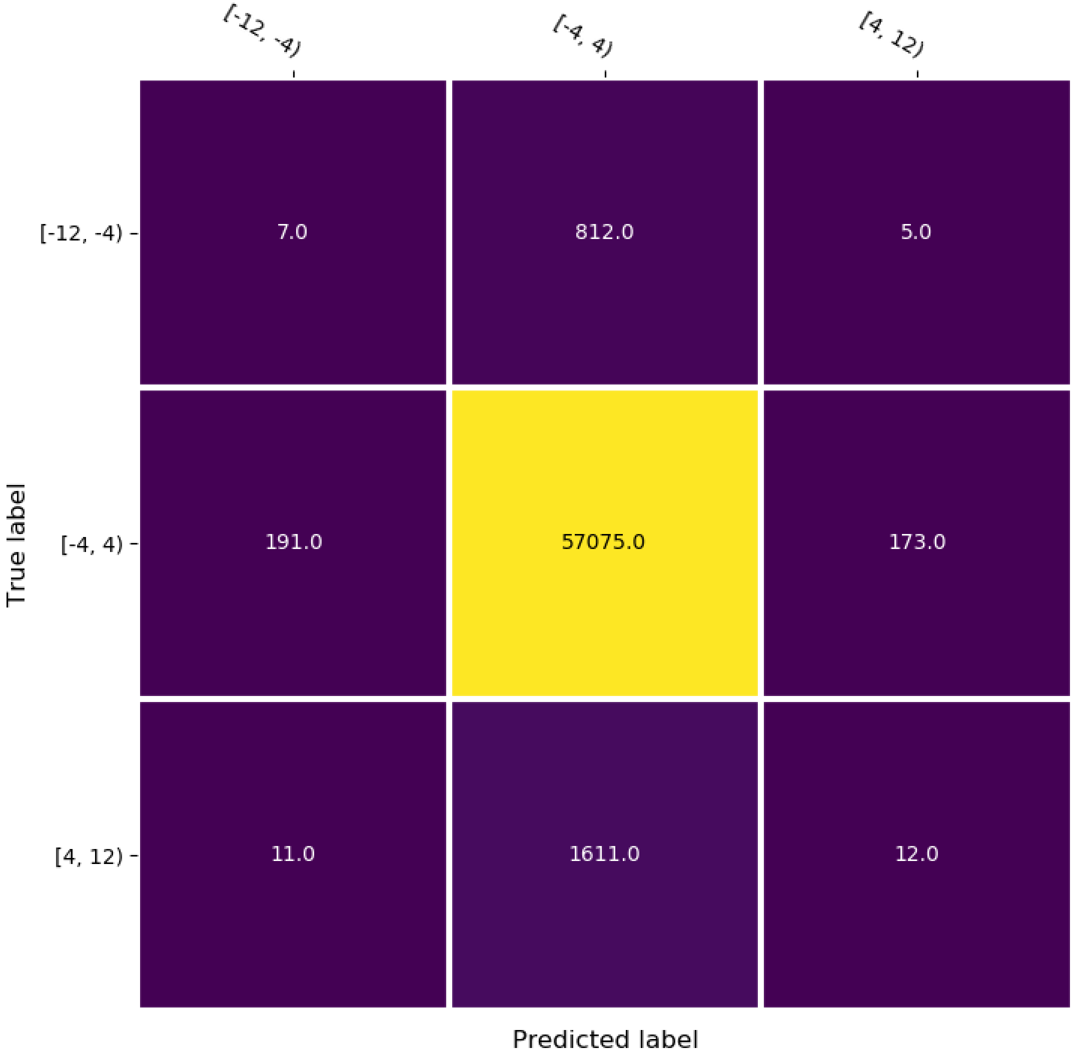
KNN

	ranging by interval 1
Accuracy:	56.96%



KNN

	ranging by interval 1
Accuracy:	95.32%



KNN

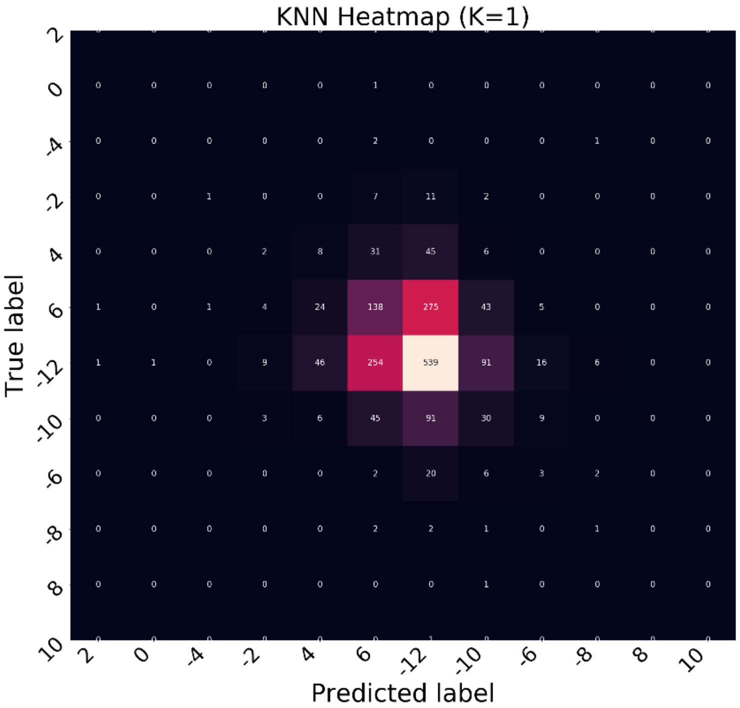
	ranging by interval 1	ranging by interval 4	ranging by interval 8
Accuracy:	22.18%	56.96%	95.32%
Mean Squared Error :	5.84		

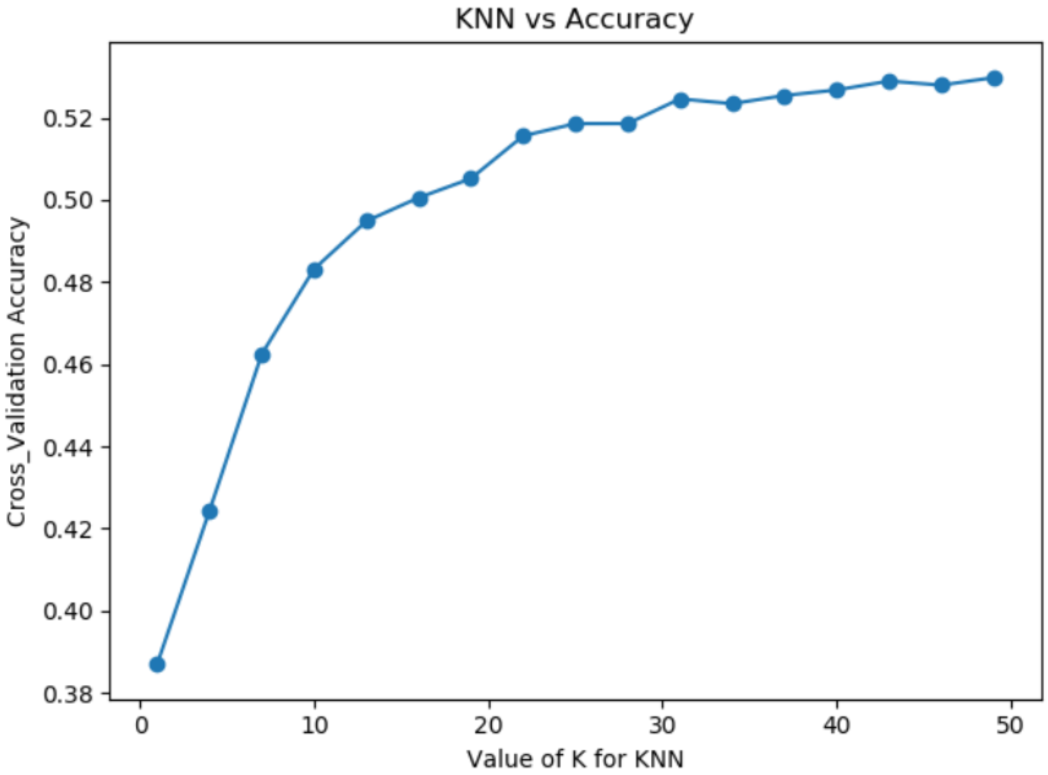
Summary and conclusions

- Accuracy
  - Decision Tree : 34.33
  - Random Forest : 33.85
  - Naive Bayes : 33.45
  - KNN : 22.67
- Mean Squared Error (MSE):
  - Linear Regression : 2.65
  - Decision Tree : 2.87
  - Random Forest : 2.89
  - Naive Bayes : 3.06
  - KNN : 5.86

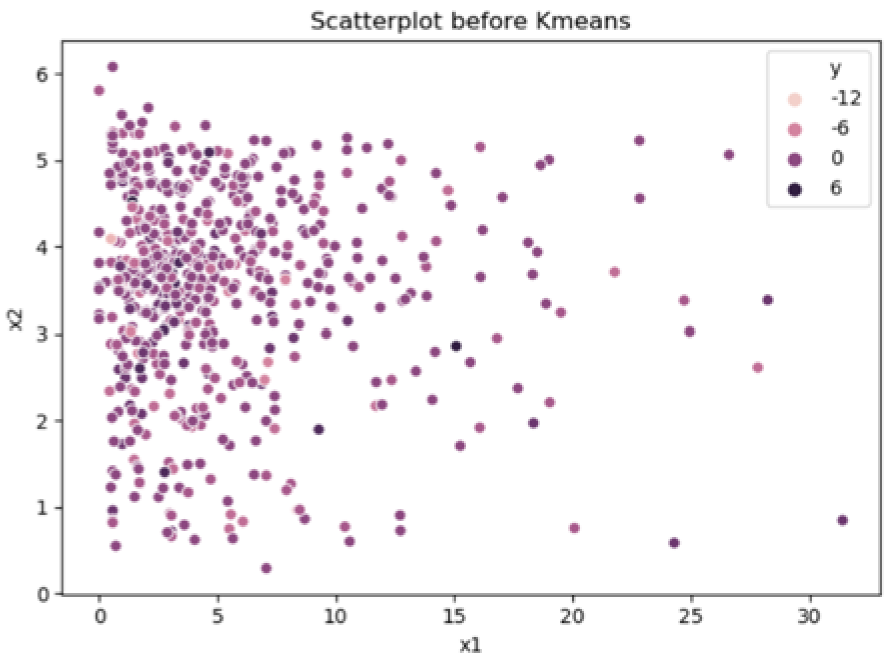
		Accuracy	MSE
Decision Tree:	ranging by interval 1	33.43	2.65
Random Forest:	ranging by interval 1	All features: 33.93% K features: 33.29%	All features: 2.89 K features: 2.89
	ranging by interval 4	All features: 66.06% K features: 65.52%	
	ranging by interval 8	All features: 95.89% K features: 95.89%	
KNN:	ranging by interval 1	22.18%	5.84
	ranging by interval 4	56.96%	
	ranging by interval 8	95.32%	
Naive Bayes:	ranging by interval 1	33.46%	3.06
	ranging by interval 4	52.74%	
	ranging by interval 8	87.63%	

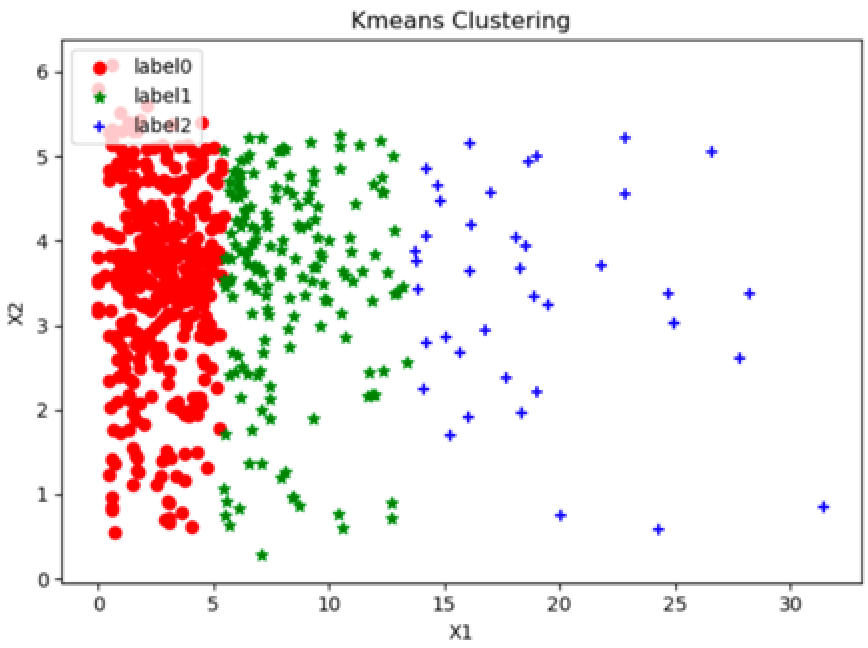
KNN


KNN Heatmap (K=1)
True label
Predicted label


KNN vs Accuracy
Cross_Validation Accuracy
Value of K for KNN

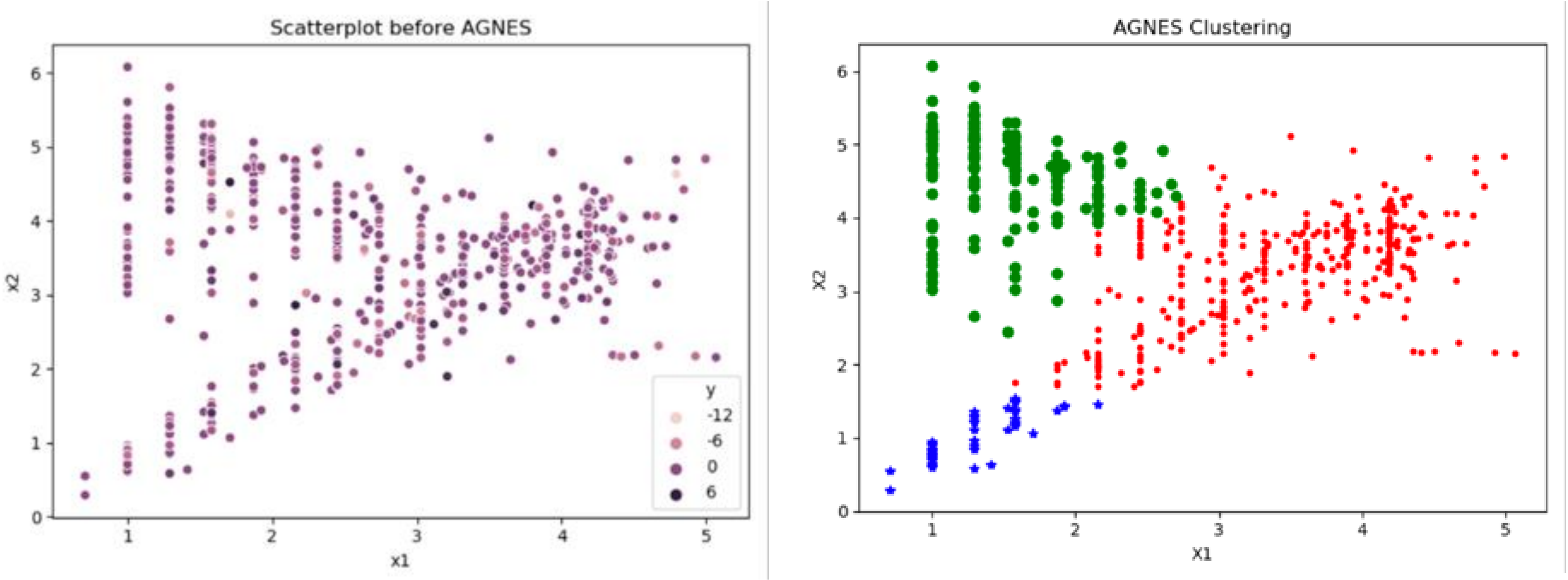
K-means





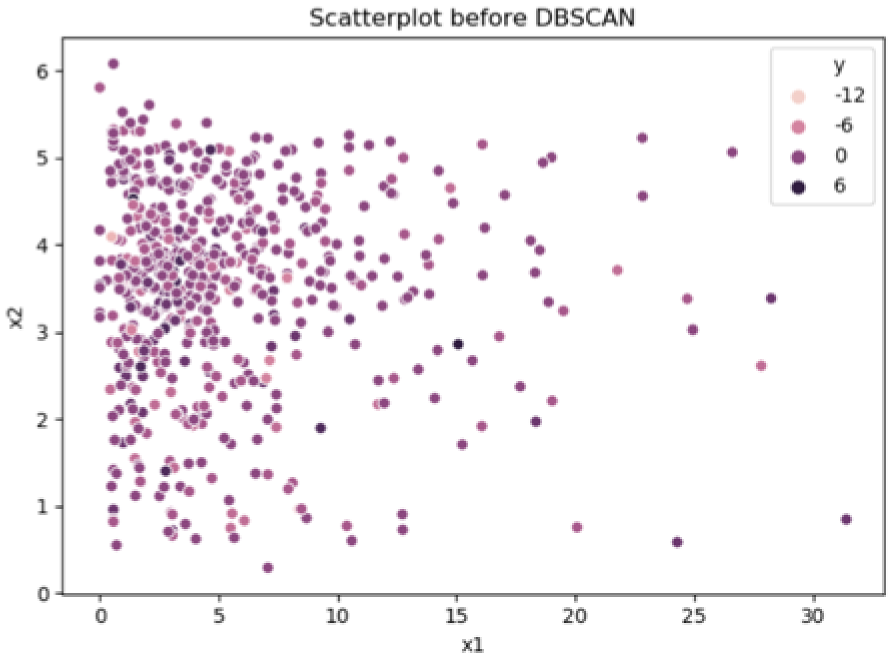
When features are selected as "purchase_amount_count_std" and "auth_purchase_month_std", scatter plots of raw data and result data after K-Means Clustering.

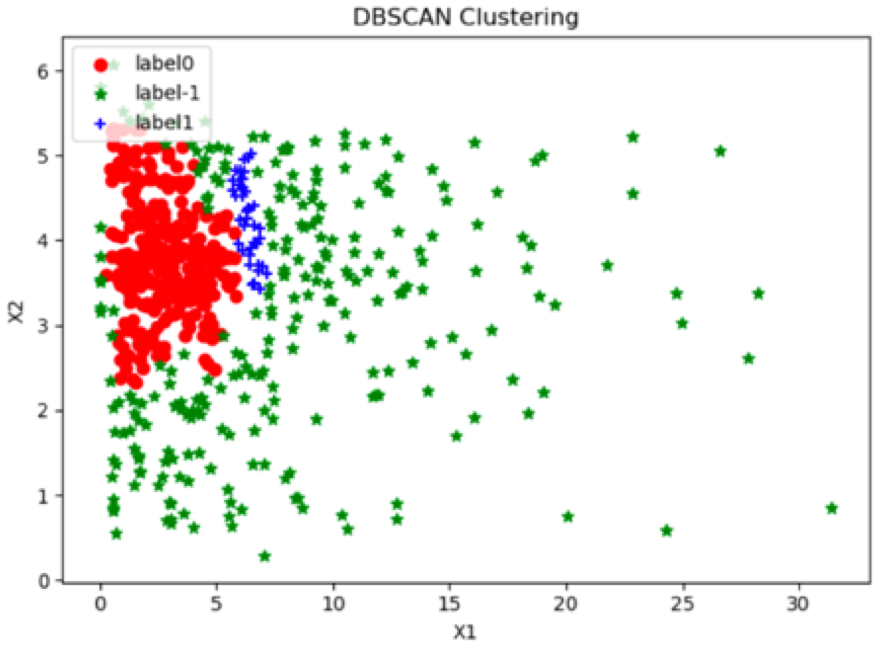
AGNES



When features are selected as "month_lag_std" and "auth_purchase_month_std", scatter plots of raw data and result data after AGNES.

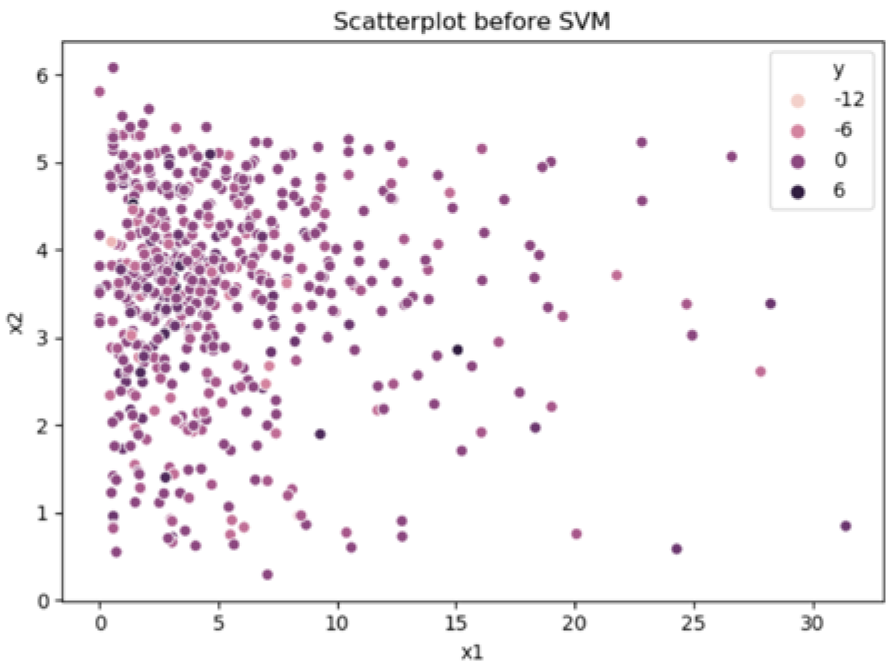
DBSCAN

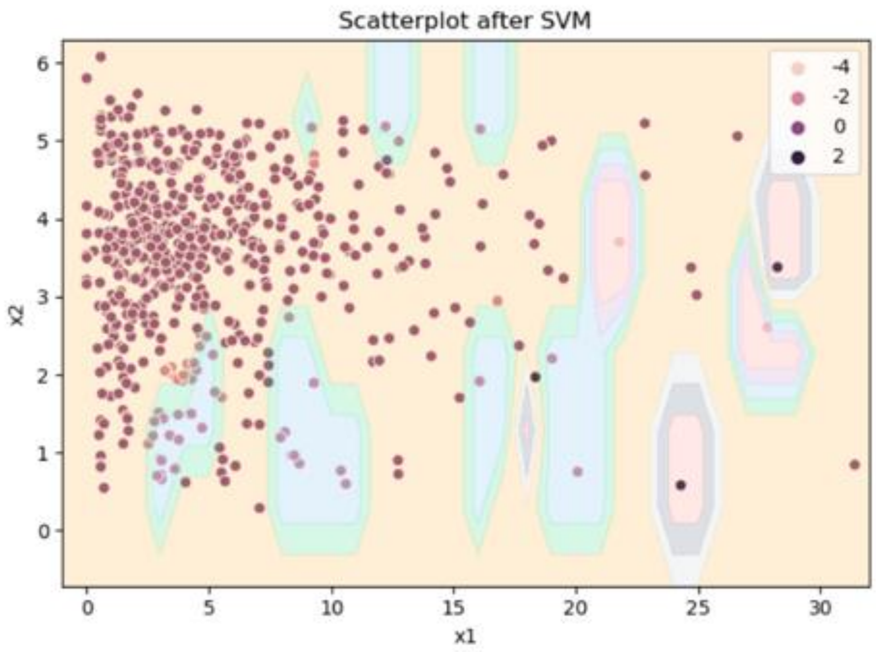




When features are selected as "purchase_amount_count_std" and "auth_purchase_month_std", scatter plots of raw data and result data after DBSCAN.

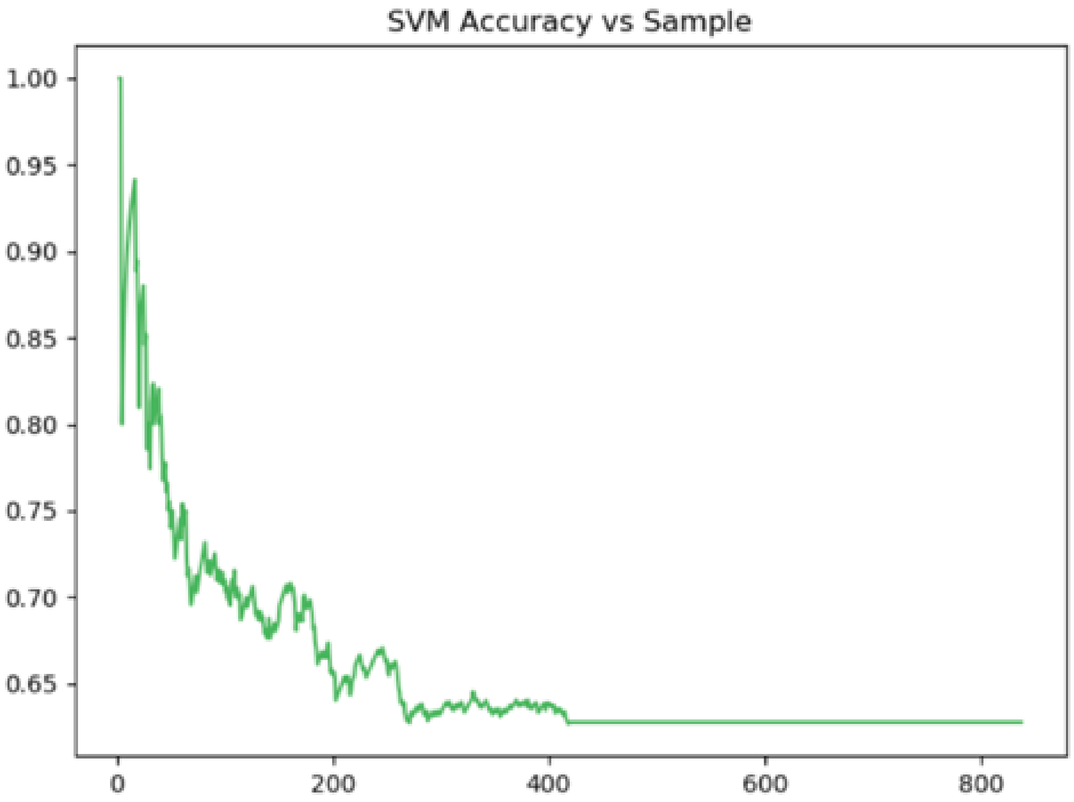
Support Vector Machine (SVM)





When features are selected as "purchase_amount_count_std" and "auth_purchase_month_std", scatter plots of raw data and result data after SVM.

Support Vector Machine (SVM)



The chart shows the relationship between sample size and accuracy in Support Vector Machine (SVM). It can be seen from the curve that as the sample size increases, the accuracy of the SVM gradually stabilizes, and finally remains at about 62.5%.

References

https://towardsdatascience.com/ways-to-detect-and-remove-the-outliers-404d16608dba

Elo Merchant Category Recommendation Help understand customer loyalty. Elo. (March, 2019). Retrieved from https://www.kaggle.com/c/elo-merchant-category-recommendation

Thank you!

Questions?