GYM CHURN ANALYSIS

Oliver Xing

Introduction:

Gym churn rate prediction involves analyzing membership data to identify patterns and factors leading to customer attrition. There are two main goals.

1. By dividing customers into **distinct groups**, we can label them based on their likelihood of discontinuing their gym membership. This approach helps in proactively **identifying high risk customers**, **enabling targeted intervention strategies to enhance customer retention**.

2. By creating a **model**, we can input **all or several variables** to forecast **the churn status of a specific member**.

Back

Data Description

Data description

1. **Data type used**: CSV file
2. **Size of dataset**: 14 columns, 4000 rows
3. **Source of data**: Kaggle

Data description

4. Data name description:

Categorized variables(represented by 0 or 1):
gender: gender of the member
near: whether gym near home
partner: whether two members workout together
promo: whether the member has promotion to renew membership
phone: whether the member has phone number on file
group_visits: whether more than two members workout together

Numeric variables:
contract_period: period of contract member signed
age: age of the member
add_charge: extra charge customer paid except membership fees
month: remaining number of months in contract
lifetime: number of month since first establish the membership
frequency: member show up frequency in a week
frequency_total: member show up frequency in a month

Data description

5. Questions regarding the dataset:

Phone number:

I know phone number is must have while registering membership. So, it is strange that member does not leave a phone number. I will check the percentage of 0 in phone column

Contract period and remaining number of months in contract:

According to the dataset, there is no requirement for minimum contract period. So month carries different weight depends on member's contract period

In this case, I will create a column called month_percentage, which is calcualted by month / contract_period

Data description

6. **Data quality**

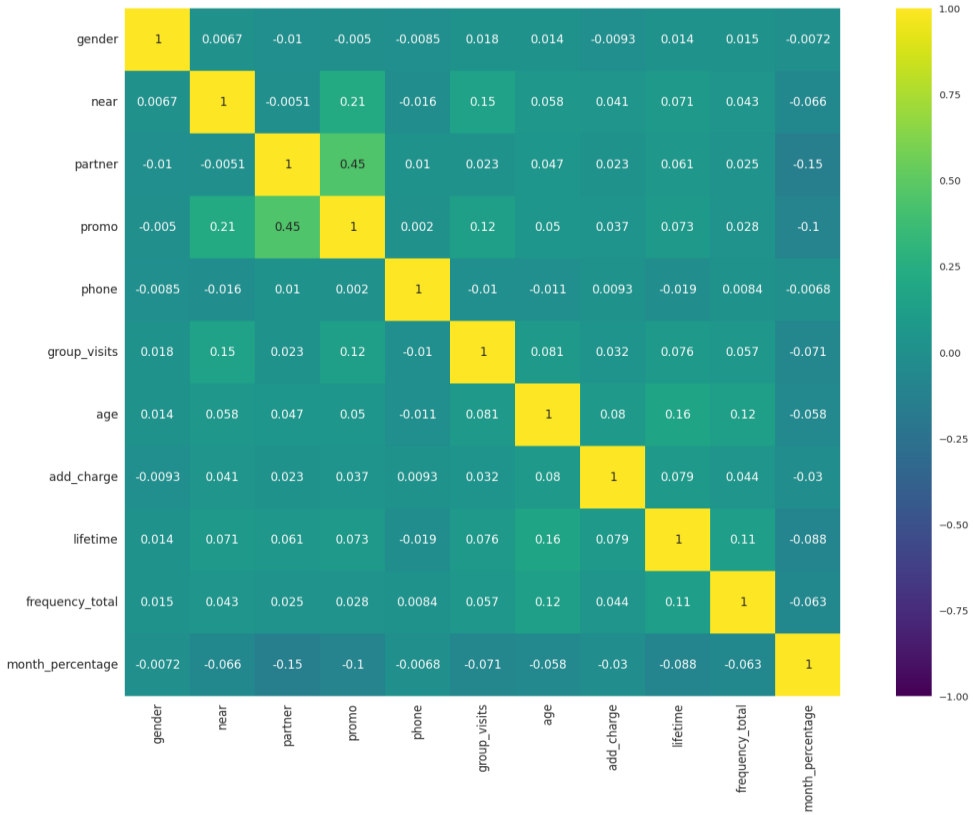
- There is no **missing values** or **duplicate values**
- All numeric variables are **within the range**.
- **Data types** are right

Index	Add Charge	Frequency Total	Frequency	Contract Period	Age	Month	Lifetime
max	552.59	6.02	6.15	12	41	12	31
min	0.15	0.0	0.0	1	18	1	0

Data description

7. **Suitable for ML?**

Only promo and partner have relative high correlation. The data should be fine for ML.



Analysis Methods

Analysis Methods

First part:

1. **Type of the problem**: unsupervised learning
2. **Method:** K-means

- Prepare the dataset. K-means can not work with categorized data. In this case, I will drop them
- Using **Elbow method** to choose the number of clusters
- Input the number of **clusters** and **random state** (Kmeans(n_clusters=, random_state=))
- Check the precision using **Silhouette Score**

Analysis Methods

Second part:

1. **Type of the problem**: Supervised learning
2. **Method**: Logistic regression, XGBoost

Logistic regression:

- Split the dataframe to **X_train**, **y_train**, **X_test**, **y_test**
- Select two interested variables to fit the **regression model**

```
sm.Logit(y_train,sm.add_constant(X_train[['month','frequency_total']])).fit()
```

- Select all variables to fit the **regression model**
- Check the **precision score**

Analysis Methods

XGBoost:

- Split the dataframe to **X_train**, **y_train**, **X_test**, **y_test**
- Search the best parameters for the model
- Input **X_train**, **y_train**

```
xgb_model.fit(X_train, y_train)
```

- Build a confusion matrix using **X_test**, **y_test**

```
pred_mgb = xgb_model.predict(X_test)
mat_mgb = confusion_matrix(y_test, pred_mgb)
```

- Check the **precision score**

```
classification_report(y_test, pred_mgb, target_names=class_names)
```

Results Content

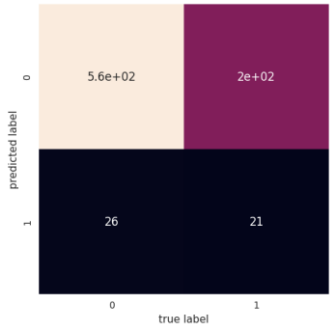
Logistic regression (two variables):

- Apply Cross-Validation on Training Data

0.88	0.86	0.88	0.89	0.89

The scores are relatively consistent, ranging from approximately 86.25% to 88.75% accuracy.

- Check confusion map



This part is quite strange. I read through the code several times but still produce this matrix.

- Accuracy score is 0.72, which is not good enough. I will add all variables too see if I can improve the accuracy

Results Content

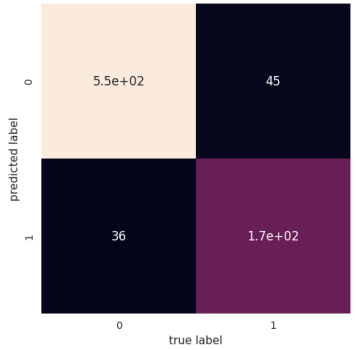
Logistic regression (all variables):

- Apply Cross-Validation on Training Data

0.88	0.86	0.88	0.89	0.89

The scores are relatively consistent, ranging from approximately 86.25% to 88.75% accuracy.

- Check confusion map

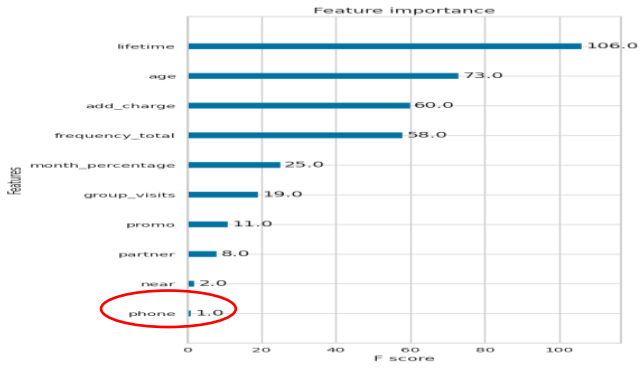


- Accuracy score is 0.899, which is quite good and improves a lot

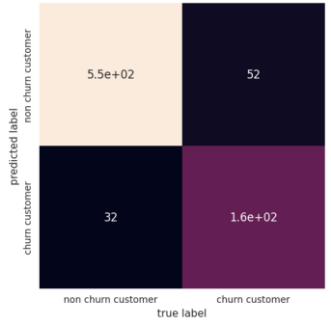
Results Content

XGBoost (all variables):

- Look “Feature importance”. In the Question section, I think phone should not influence the analysis because every member should leave a phone number or there is an error while collecting data. From the graph, phone importance is 1.0.



- Check confusion map

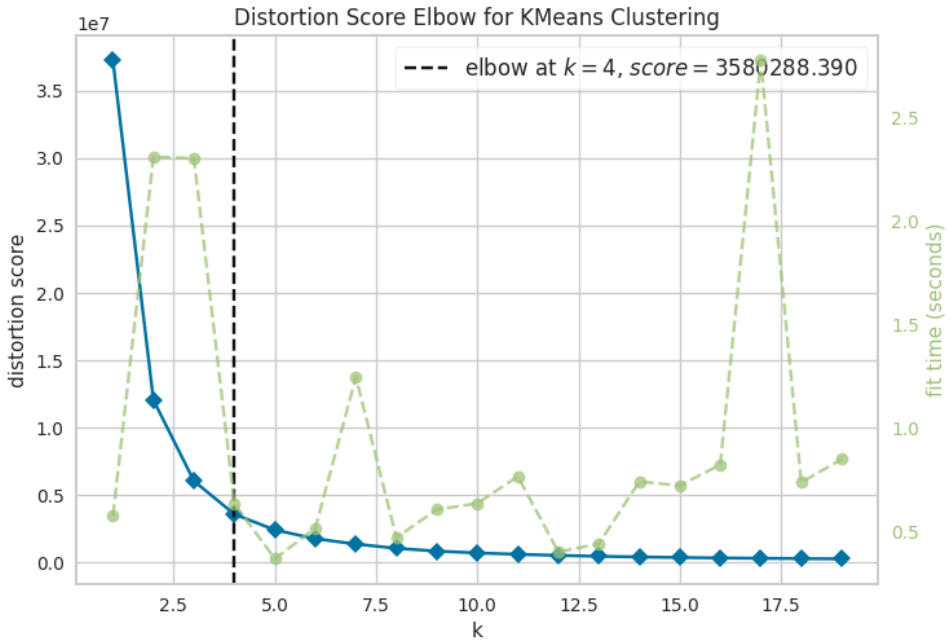


- Accuracy score is 0.90, which is higher compared to logistic regression

Results Content

K-Means:

- Choose number of clusters



The image indicates that **cluster of 4** is appropriate

Results Content

- After predicting clusters, I use Silhouette Score to check the accuracy (The silhouette score ranges from -1 to +1)

+1	A score close to +1 indicates that the data points are very far from neighboring clusters, suggesting excellent cluster separation
0	A score around 0 implies overlapping clusters, where data points are, on average, just as close to their own cluster as to the nearest neighbor cluster
-1	A score near -1 indicates that data points have been assigned to the wrong clusters

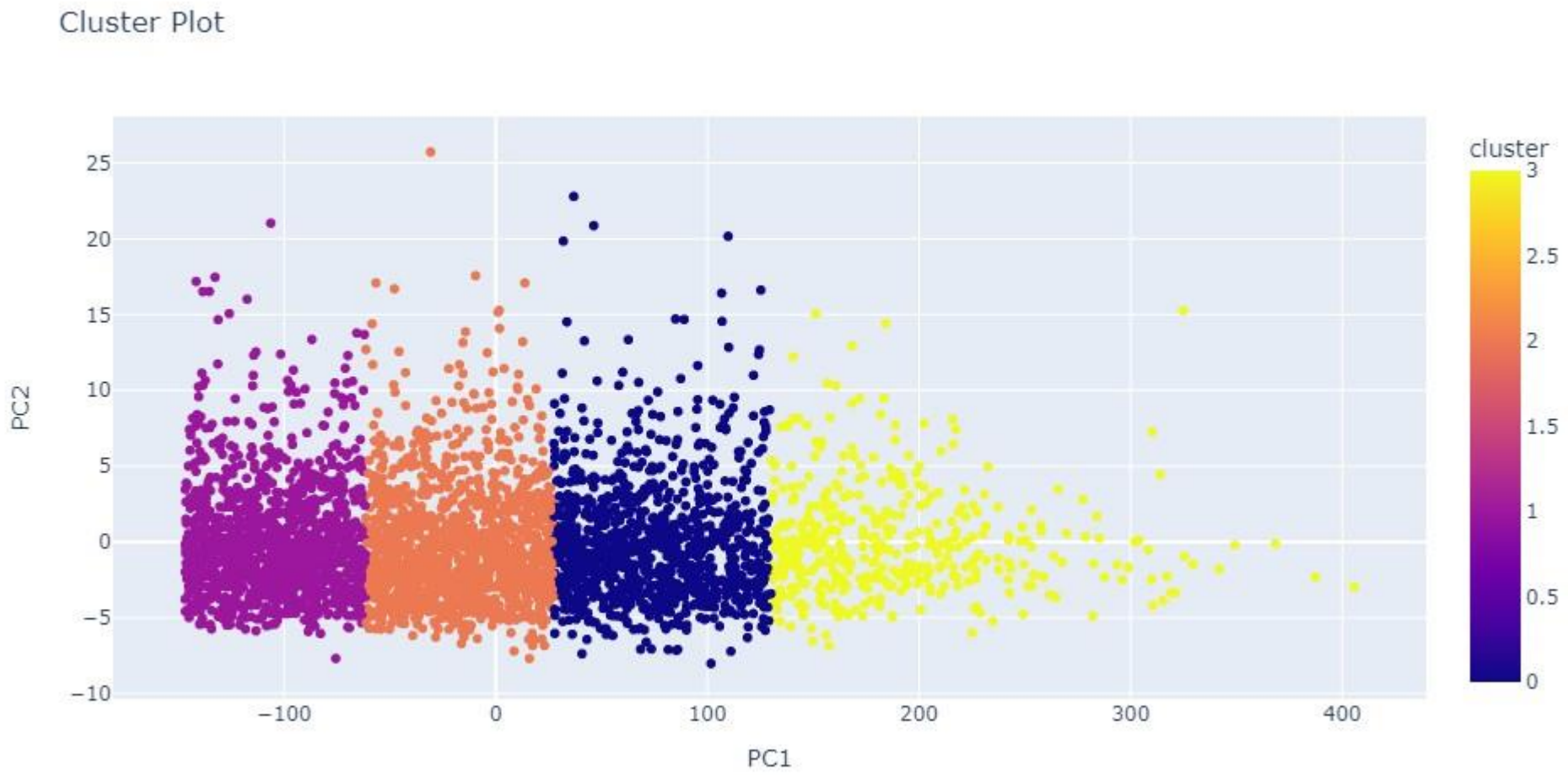
I got a score of 0.54. A silhouette score of 0.51 suggests some degree of separation between the clusters, but there may be some overlap or ambiguity in how the data points are assigned to clusters.

However, in our context (customer churn segmentation analysis), a moderate silhouette score may be acceptable if the goal is to identify distinct customer segments based on their behavior characteristics.

In this case, even if there is some degree of overlap or ambiguity between the clusters, the identification of distinct clusters may still provide valuable insights for the marketing and business strategy.

Results Content

- Utilize PCA to create a cluster plot



Results Content

- I set four clusters to find a benchmark of different variables. I use benchmark to personalize marketing strategy for members.

- Firstly, I find the churn rate for each clusters

0	0.197
1	0.347
2	0.302
3	0.077

1 and 2 have significantly higher churn rate. In this case, I'm interested in the characteristics of these two groups.

Results Content

- Secondly, I compare the mean of each variable for each cluster to the mean of each variable for whole dataset.

Cluster	Age (mean)	Add Charge (mean)	Lifetime (mean)	Frequency Total (mean)	Month Percentage (mean)
0	29.38	219.91	3.99	1.89	0.96
1	28.96	42.28	3.45	1.83	0.96
2	29.04	128.83	3.58	1.88	0.97
3	29.83	334.26	4.35	1.99	0.96

Variables means for each cluster

Index	Age (mean)	Add Charge (mean)	Lifetime (mean)	Frequency Total (mean)	Month Percentage (mean)
mean	29.184	146.944	3.725	1.879	0.963

Variables means for dataset

- Age, Add Charge, Lifetime, and Frequency Total values are smaller than the correspond means for the whole dataset

Conclusion Content

- XGBoost did the best at predicting. From highest contribution to the lowest: lifetime, age, add_charge, frequency_total, month_percentage, group_visits, promo, partner, near, and phone

- Suggestion:

1. Gym can create a special loyalty program for new members whose age is smaller than 29

2. Gym can develop promotions for add-on service if the median of add charge below 146.94

3. Gym can send motivation alerts to members whose monthly show up rate is less than 2

Conclusion Content

Shortcomings:

- The dataset has limited information compared to the real world. For instance, age range from 18-41. It is unsure if analysis (younger age tends to have higher churn rate) holds if we add age larger than 41

- There are many ways for classification. Some advanced methods might produce higher Silhouette Score

- I only compared scaling or not for preprocessing the classification data. Other preprocessing method might produce higher Silhouette Score

- The confusion map looks wrong though I checked code several times.