Google Merchandize Store

- REVENUE PREDICTION
- MARKETING STRATEGY

AGENDA

1. Business Case
2. Feature Engineering
3. Modeling
4. Clustering
5. Strategy

Business Case

Problem & Objective

The 80/20 rule has proven true for many businesses–only a small percentage of customers produce most of the revenue. But major part of the revenue comes from only 20% of the customers.

MARKETING BUDGET ALLOCATION

Marketing budget can better be utilized If they target only those users who are most likely to purchase a product in the future.

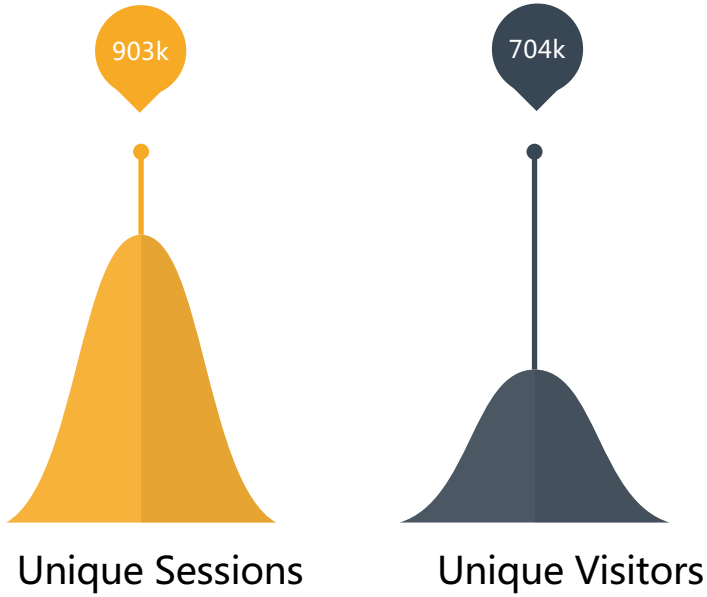
REVENUE PREDICTION

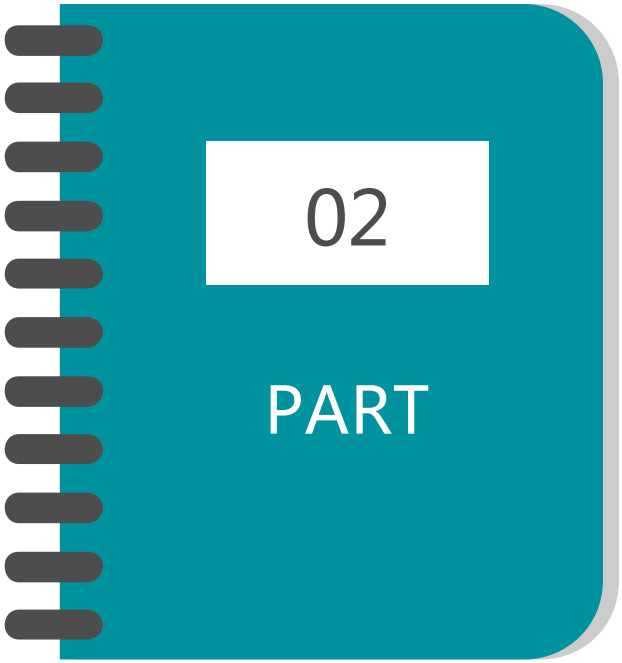
Analyze a Google Merchandise Store customer dataset to predict revenue per customer

Data Source

Our Data is from Google Analytics, GA is a tool for tracking online visitor behavior, such as their pageviews, how long do they stay and their transaction revenue.

The time spans of our data is from Aug 1, 2016 to Aug 1, 2017, it records 903,653 sessions, 704,353 unique visitors.



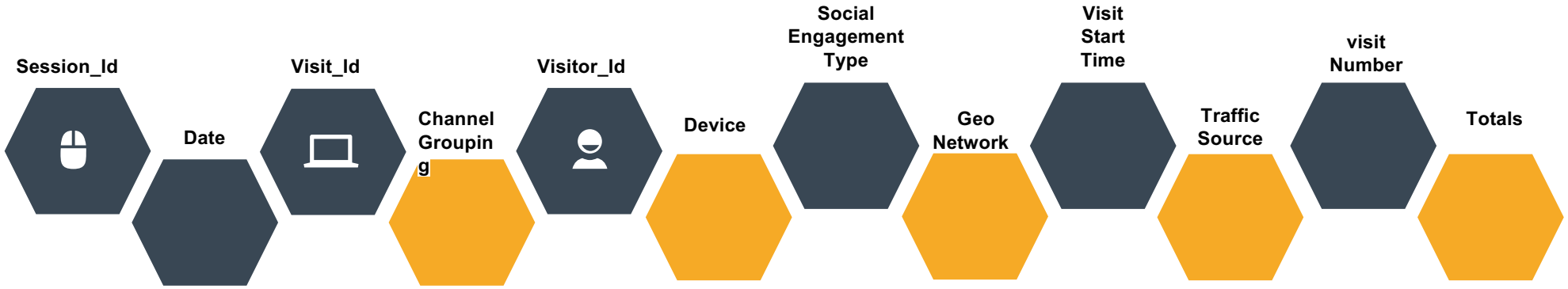


Feature Engineering

- Data Description
- Data Exploring
- Data Transformation
- EDA

Data Description

The raw data contains 12 columns, 5 of them are in the nested Json format.



- (Other)
- Affiliates
- Direct
- Display
- Organic Search
- Paid Search
- Referral
- Social

- Browser
- Operating system

- Continent
- Subcontinent
- Country
- City

- campaign
- source
- medium
- keyword

- Visits
- Hits
- Pageviews
- Bounces
- NewVisits

** full description is in the appendix*

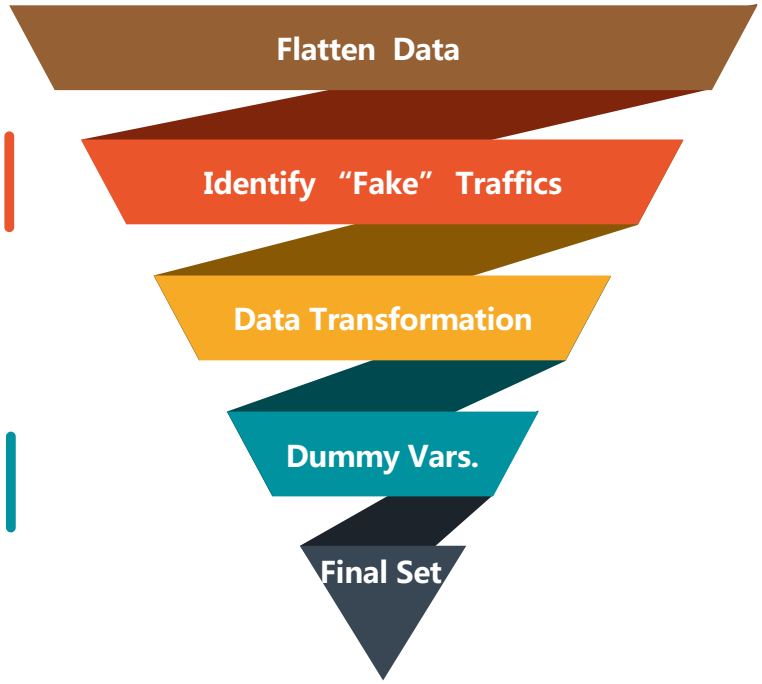
Feature Engineering

Identify "Fake" traffics

- Define "fake" traffics shows strong internal testing purpose, such as traffics from Google Analytics, Github, third party traffic tracking tools.
- Verify with transaction values
- Deleted 33k "fake" traffics

Dummy Variables

- Many of our columns carried categorical data, therefore, we created dummy variables.
- eg.: Weekdays, Visit hour, Subcontinent, Devices etc.
- Create campaign identifier if this traffic comes from Google Adwords



Session level dataset: 903,653 rows x 95 columns
Visitor level dataset: 704,353 rows x 95 columns

Flatten Data

- Flatten the JSON columns to 59 columns.
- Delete variables that has more than 75% missing values (expect Transaction Value).
- Drop variables that have same values.
- Reduced to have 29 columns

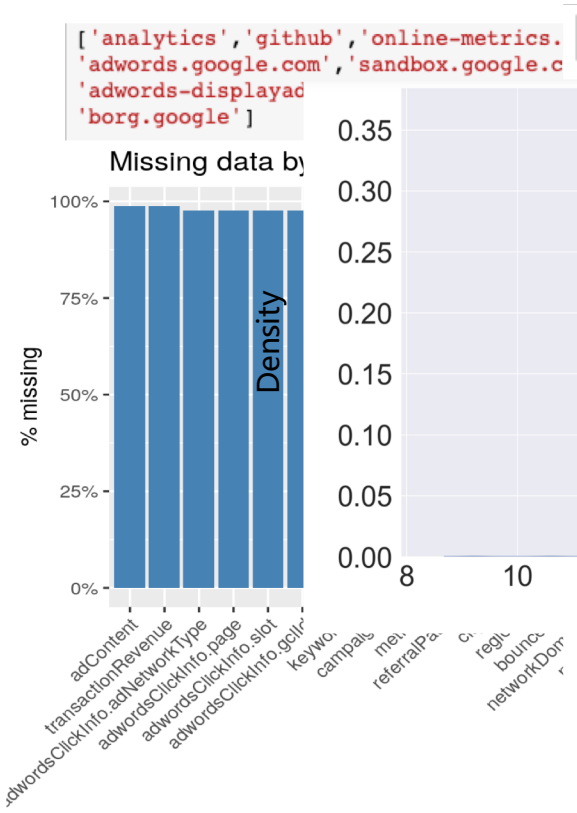
Data Transformation

- Filling missing values using mode. Such as filled Pageview with 1. Revenue with 0
- Construct time-related variables

Final Set

- Created session level dataset
- Created visitor level dataset group by visitor id, take corresponding aggregate functions to each columns

EDA - Feature Engineering




```
['analytics','github','online-metrics.
'adwords.google.com','sandbox.google.c
'adwords-displayad
'borg.google']
```

```
df_model.columns
```

```
its',
ch',
',
filiate',
ferral',
er_new_Edge',
icas',
',
',
',
o Wii',
,
```

3	0000			1.0
4	0000039460501403861	2.0	2.0	0.0	1.0

```
's_Thursday',
_hour_1',
'visit_hour_5',
'visit_hour_9',
', 'visit_hour_13',
'visit_hour_14', 'visit_hour_15', 'visit_hour_16', 'visit_hour_17',
'visit_hour_18', 'visit_hour_19', 'visit_hour_20', 'visit_hour_21',
'visit_hour_22', 'visit_hour_23'],
```

columns to 59 columns.
that has more than 75%
xpect Transaction Value).
at have same values.
29 columns

mation

alues using mode. Such as
with 1. Revenue with 0
related variables

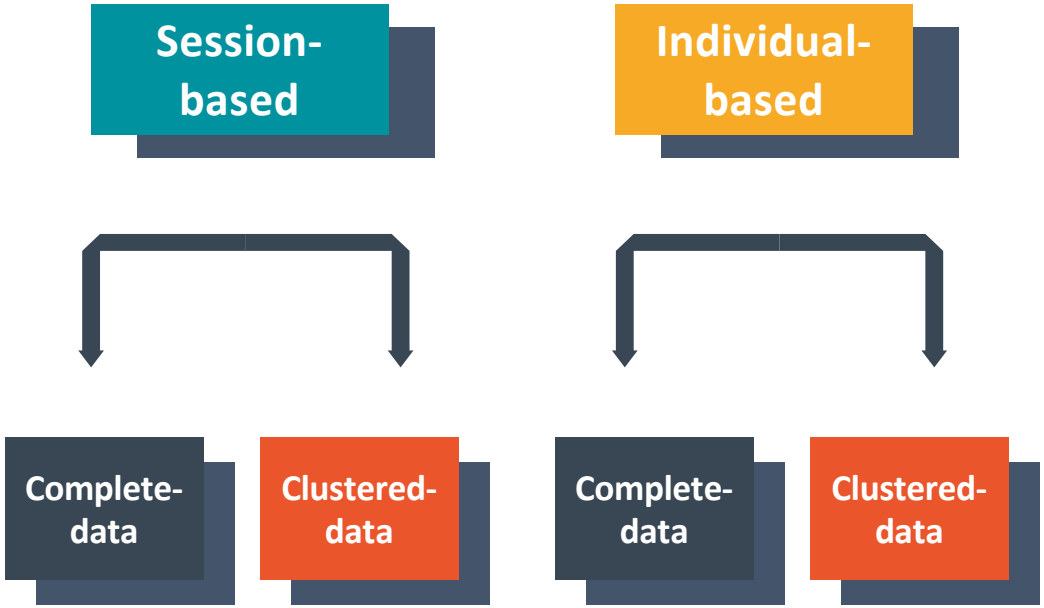
level dataset
evel dataset group by
orresponding aggregate
h columns

Business Case | **Feature Engineering** | Modeling | Clustering | Strategy

Modeling

- ML Models
- Clustered
- Session/Individual
- Balanced

Hierarchical Modeling Approaches



Approach-1:
Session-based vs. Individual-based Modeling
To serve for different purposes

Approach-2:
Complete-data vs. Clustered-data Modeling
To improve models' performances

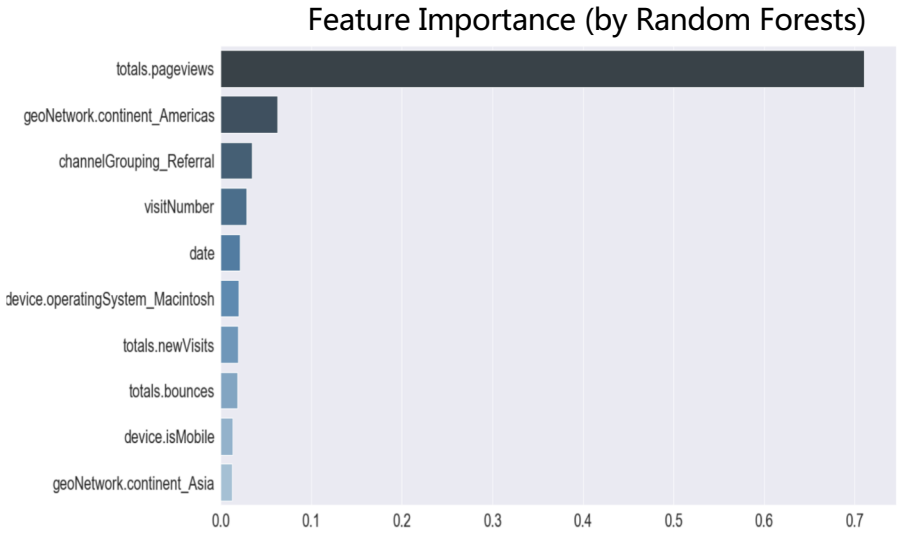
Approach-1: Session-based vs. Individual-based Modeling

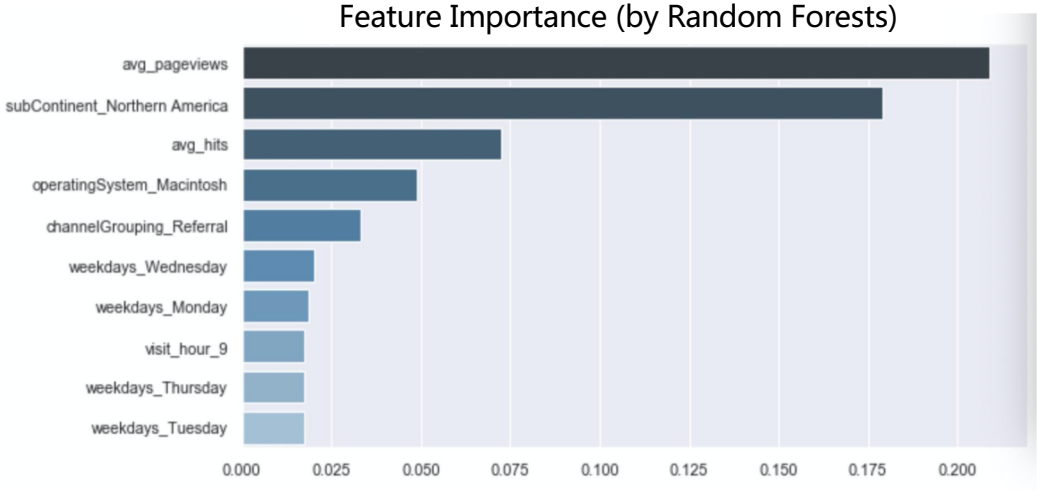
Session-based Modeling

- Original dataset to predict revenue for each session

Individual-based Modeling

- Aggregated features based on unique Individual ID (e.g. sum transaction level, average pageview)
- Most new created features' weights swell compared to original features that they are derived from





Candidate Models

	Pro	Con
Random Forest	Average result resulting in unbiasede result	Took time to tune the parameters
Gradient Boosting	Corrected errors made by each sample to enhance results	May result in over-fitting
XGB Boosting	Resolved over-fitting by puting a penalized terms	Time-consuming to deliver the result
Ridge Regression	Less time to finalize result and remove multicollinearity	Results are, most of time, not as good as Machine Learning Model
Lasso Regression	Less time to finalize result and remove multicollinearity	Like Ridge Regression, is parametric method and has assumption of residual
Linear Regression	Good interpretability	Several assumptions about linear regressions should be met

Models' Performance Assessment

Session-based Models Performances

	GradientBoosting	LinearRegression	RandomForest	Ridge	XGBRegressor
All_Customer (RMSE)	1.82	2.01	1.82	2.01	1.92

Individula-based Models Performances

	GradientBoosting	Lasso	RandomForest	Ridge	XGBRegressor
All_Customer (RMSE)	2.022375	2.370443	2.253886	2.263846	2.027313

- Comparison Between Session-based and Individual-based Models
  - Session-based models generally outperform Individual-based models, but the differences are tiny
  - It means a little amount of bias may be introduced with the creation of aggregated features
- Comparison Within Session-based or Individual-based Models
  - Tree models outperform GLM models
  - Boosting result outperforms Random Forest

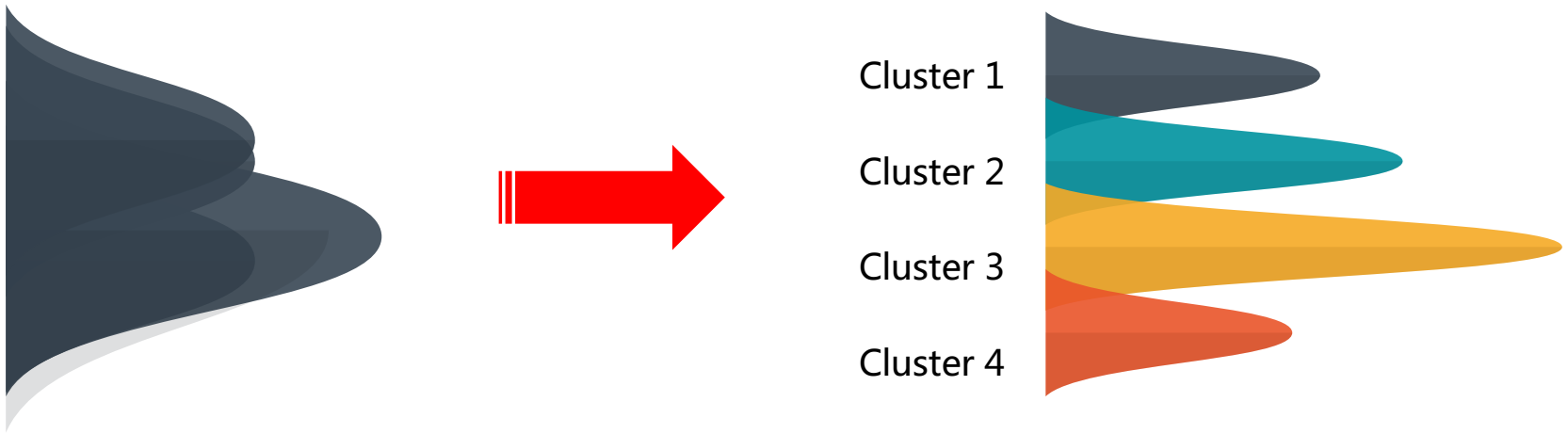
Approach-2: Complete vs. Clustered Modeling

Complete Modeling

- Complete dataset for all models
- **Pro**: Can be compared among candidate models directly using a single RMSE
- **Con**: Patterns of different sections of data may be hard to capture by a single model

Clustered Modeling

- Clustered datasets using the selected features
- **Pro**: Respectively modeling for different clusters to choose the best fit for each
- **Con**: Time-consuming and Heavily relying on how fit the data is clustered



Models' Performance Assessment

Complete Modeling

- (Same as the previous page)

- **Session-based models**

	GradientBoosting	LinearRegression	RandomForest	Ridge	XGBRegressor
All_Customer (RMSE)	1.82	2.01	1.82	2.01	1.92

- **Individual-based models**

	GradientBoosting	Lasso	RandomForest	Ridge	XGBRegressor
All_Customer (RMSE)	2.022375	2.370443	2.253886	2.263846	2.027313

Clustered Modeling

- RMSE varies a lot among clusters but changes litte within a single cluster
- Individual-based Clustering improves model fitting results, except Cluster_4
- In cluster_4, Standard Deviation of response is much higher than other clusters due to the small sample size of this cluster

- **Session-based models**

	Linear Regressor	Random Forest Regressor	Ridege Regressor	XGBRegressor	GradientBoostingRegressor
1	2.173270	2.001115	2.173344	2.116877	2.011489
2	1.035442	0.990795	1.035533	1.010749	0.984474
3	3.944547	3.680210	3.944839	3.811918	3.657345
4	13.468780	8.968768	13.355924	13.633525	10.950509

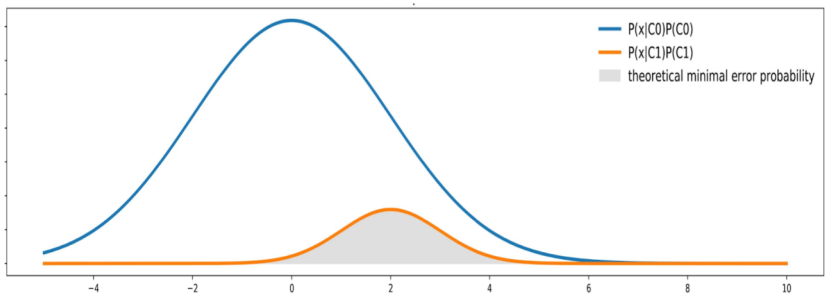
- **Individual-based models**

	GradientBoosting	Lasso	RandomForest	Ridge	XGBRegressor
Cluster1 (RMSE)	1.899737	1.939845	1.939845	1.900526	1.766069
Cluster2 (RMSE)	0.069714	0.137477	0.091165	0.000181	0.076158
Cluster3 (RMSE)	1.351296	1.458767	1.417745	1.422674	1.350185
Cluster4 (RMSE)	16.370706	15.842980	16.278821	16.031220	16.852300

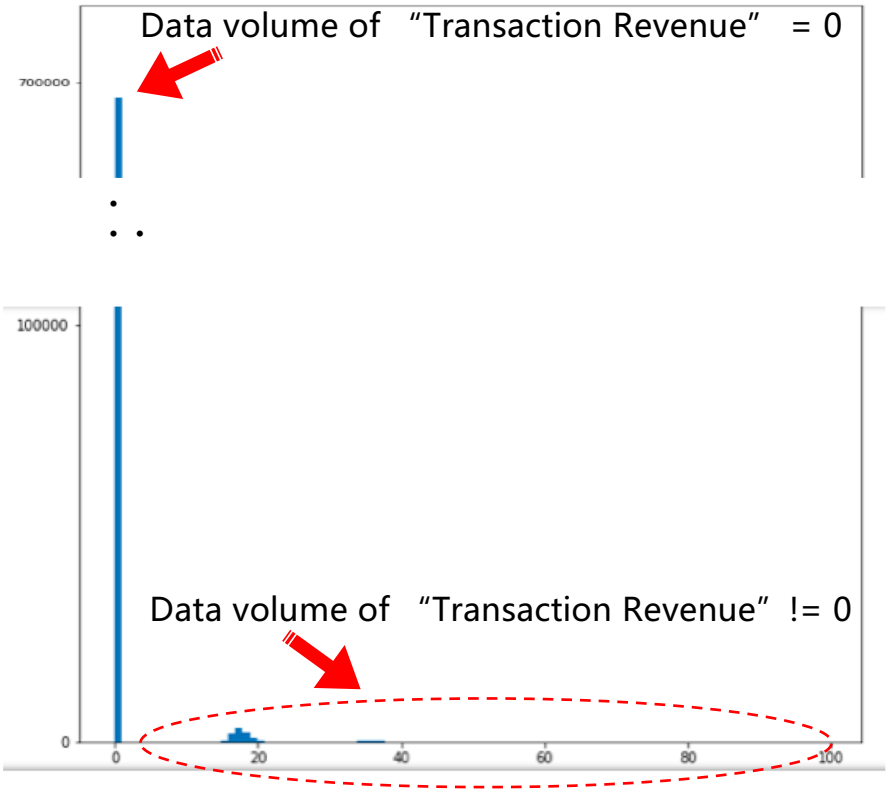
Unsolved Challenge : Data Imbalance

Problems caused by Data Imbalance

- Except the long tails, observations close to the Majority's mean are inclined to fall into its SD range
- Too small theoretical minimum error probability (area under the minority curve) to teach machines



	Zero	Non-Zero
Target Variable "Transaction Revenue"	**98.6%**	**1.4%**



Approach-3: Two-steps Modeling with Balanced Re-sampling

Step-1: Classification for Transaction

- To predict individuals who will have transactions
- Models trained with full dataset
- Given the extreme imbalance between Majority ("TR" = 0) and Minority ("TR" != 0), probability of prediction as Majority overweighs too much
- **Balanced Re-sampling is needed**

Step-2: Regression for Transaction Revenue

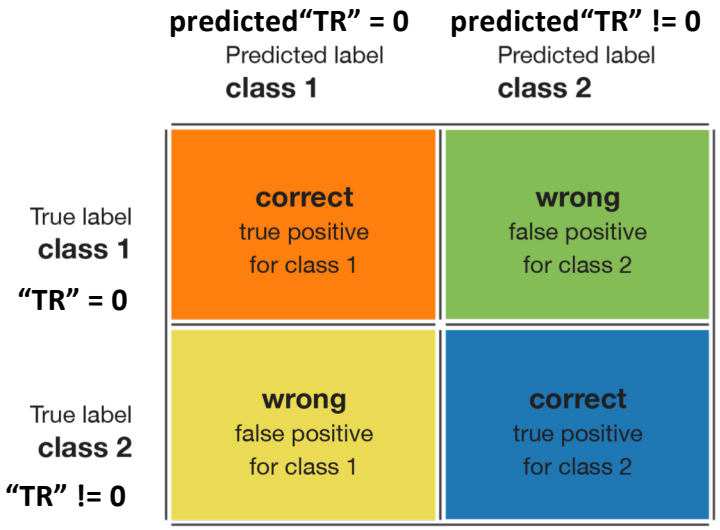
- To predict how much the transactions will be
- Models trained with sub-dataset of Minority ("TR" != 0)
- Process similar to the previous regression models

Comparison between Models

- Under-sampling better benefits model fitting than Over-sampling, but the diff is tiny
- LR and SVM overperform Tree models, although the diffs are not significant
- Overfitting is the root cause

	Over-sampling	Under-sampling
Random Forest	☑	☑
Decision Tree	☑	☑
SVM	☑	☑
Logistic Regression	☑	☑

Models' Performance Assessment

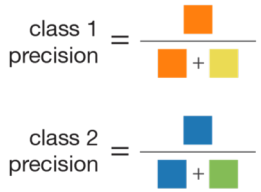


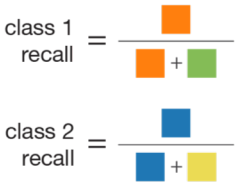
Model Interpretation

- **CONSERVATIVE for "TR = 0"**
  low Recall + high Precision : the model misses some real values, but those recognized values are highly reliable
- **LIBERAL for "TR != 0"**
  high Recall + low Precision : most (if not all) real values are recognized, but counterfeits are mixed inside

Tradeoff

- **Targetted Customers : "TR != 0"**

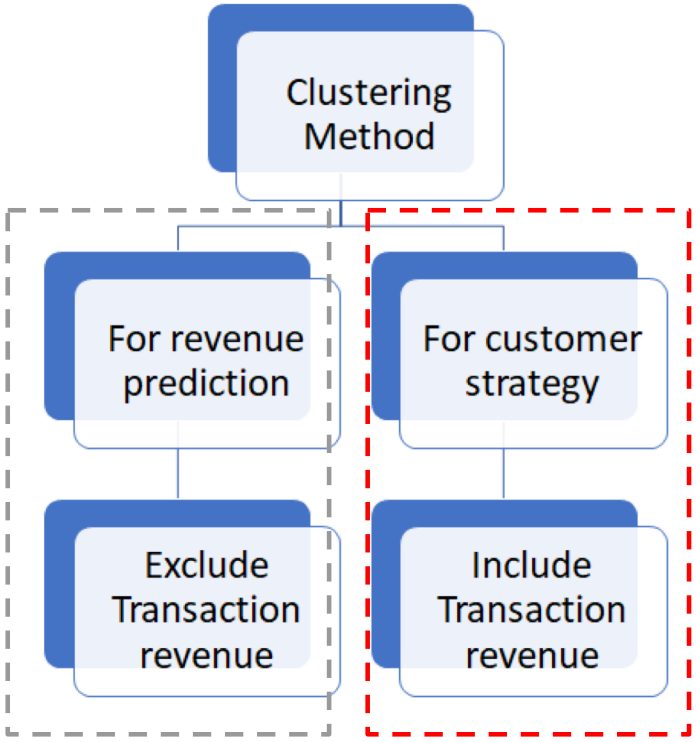


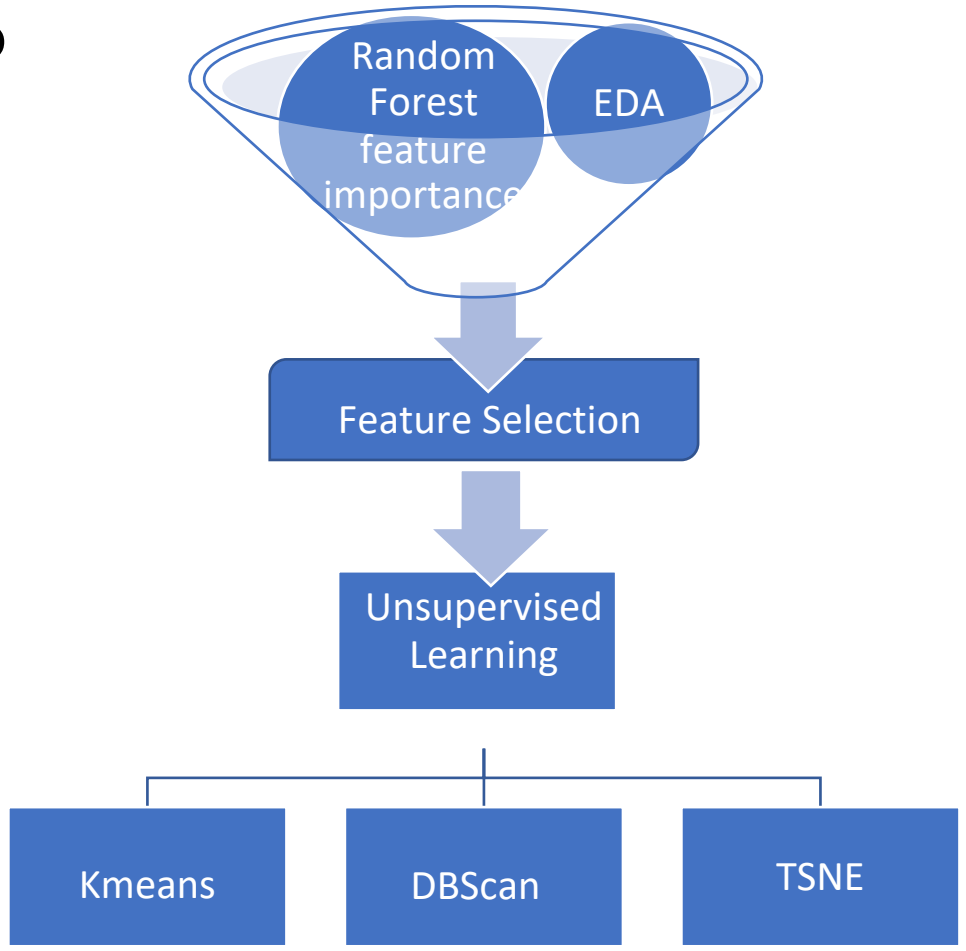


	precision	recall	f1-score
0	1.00	0.93	0.96
1	0.16	0.98	0.28
accuracy			0.93
macro avg	0.58	0.95	0.62
weighted avg	0.99	0.93	0.95

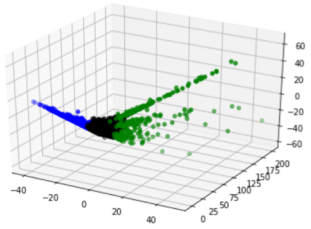
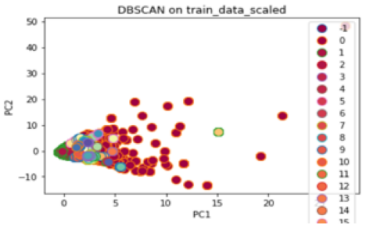
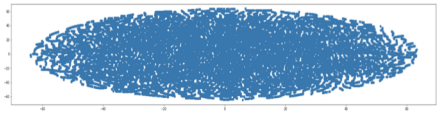
Clustering

Customer Clustering for Two Purposes



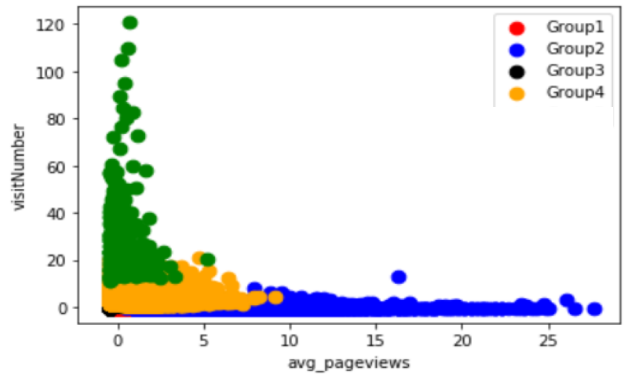


Clustering Method - Models Comparison

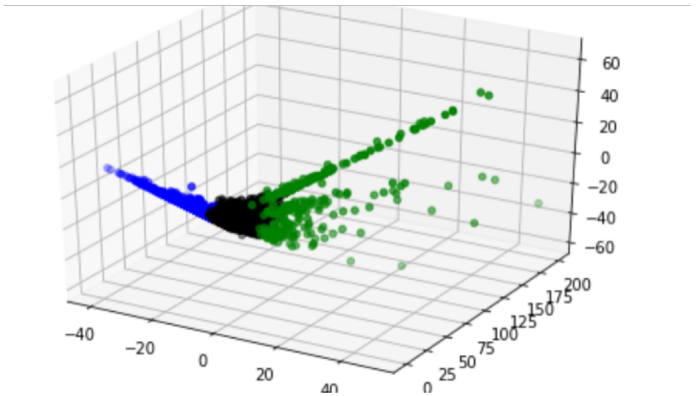
Metrics	Kmeans	DBSCAN	TSNE
Number of clusters:	5 clusters	> 20 clusters	1 cluster
Visualization:			
Conclusion:	Pro: - Clear clustering - Con: - need of determination of number of cluster first	Pro: - No need to determine the number of cluster initially Con: - Works better with non-round shape - Curse of Dimensionality	Pro: - Dimension reduction - Good for exploration purpose Con: - No clear pattern - Points are relative to each other without concrete interpretation.

Customer Clustering - Using the Cluster Centroids

Actual Features Graph -2D



Principle Components Graph - 3D



Customer Type	ColorCode	TransRevenue	Pageview	VisitNumber	Bounces	Referral
One-time buyer	Red/Blue	Low	Low	Low	Low	Low
Easy-be-influenced	Orange	High	Low	High	High	High
Budget buyer	Black	Medium	High	Low	Low	Low
Technophile	Green	High	Medium	Medium	Medium	Medium

Strategy

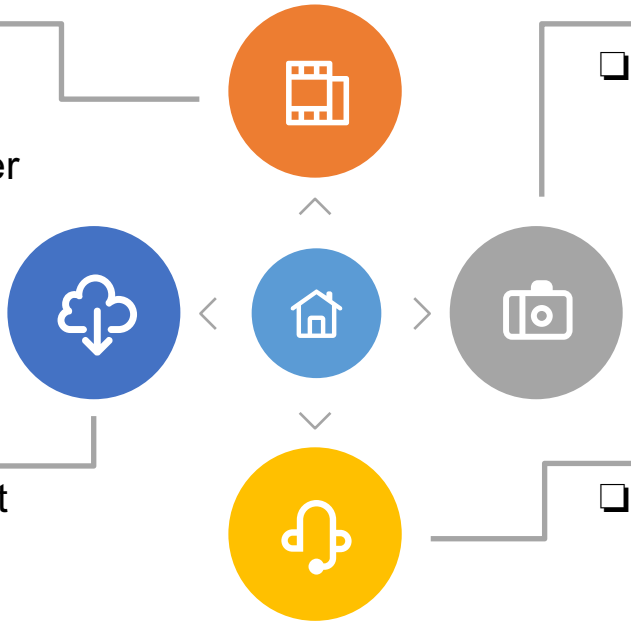
Segmented Marketing Strategy

Customer Segmentation	Strategy
• Technophile: generate a large amount of revenue with relatively low pageviews or hits	• Enhance their loyalty and encourage them to influence other customers • e.g., Start a membership program
• Easy-be-influenced: this group of people is mainly attracted by referrals	• Content driven. Establish the brand that google products are cool • Let them have more exposure to the brand
• Budget buyer: they browse the website a lot before making a purchase decision	• Price driven. Sending promotions and coupons
• One-time buyer: this group of people is the least interested in the product of our store. They wander in our store occasionally and rarely buy anything	• Boost awareness. Promote our store at special occasions

With Machine Learning techniques, Google Merchandise Store can

❑ Identify *likely* high-value *customers* by analyzing *existing* high-value customer purchasing behavior

❑ Predict revenue based on customer purchasing history

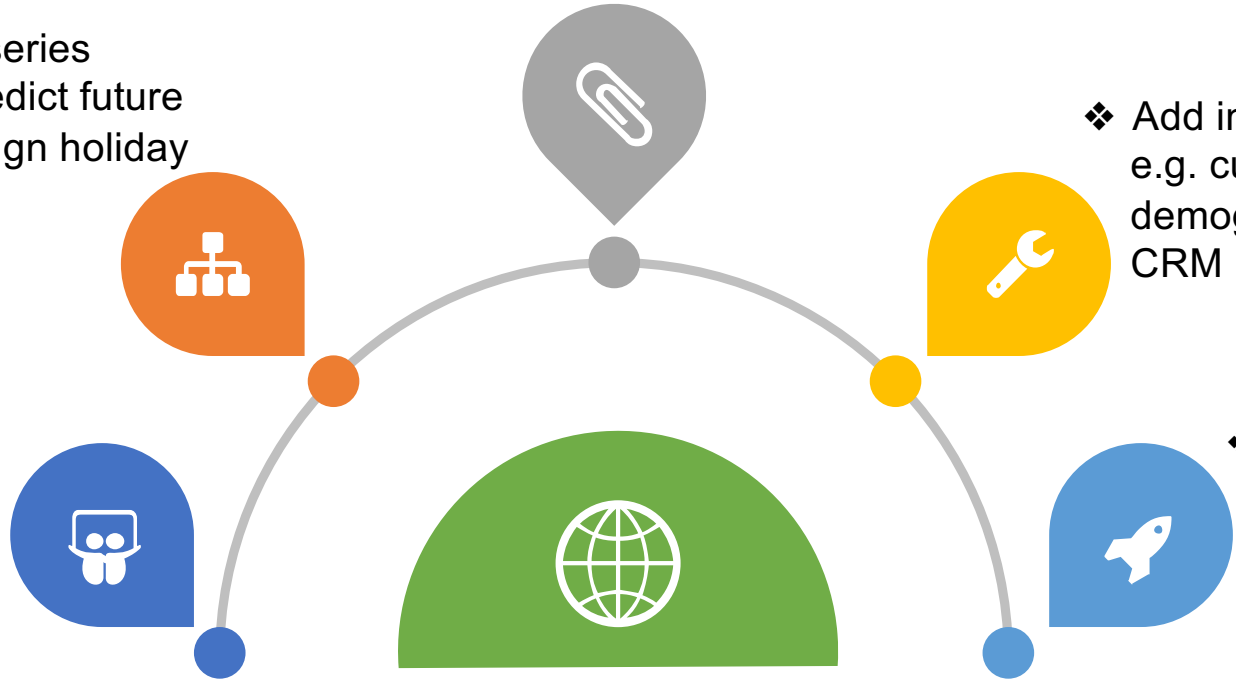


❑ Focus product development on high-value customer products

❑ Retain high-value customers by different marketing strategies

Future Plans

- NLP analysis for more information. e.g. ‘Campaign keywords’ to generate new features
- Apply time series model to predict future trend to design holiday compaign
- Add in more features. e.g. customer demographic from CRM
- A/B test to increase pageviews
- Handle imbalanced datasets



Business Case | Feature Engineering | Modeling | Clustering | **Strategy**

THANK YOU