

**ISSS610 Applied Machine Learning**

**Predicting Conversion From Google Merchandise Store**

**Project Report**

By: Group 5

LIM SI LING EVELYN

LEE MENG YONG

JIANG NAN

HO WEIWEN DESMOND

**13 August 2019**

# Motivation

According to a joint report published by Temasek Holdings and Google[[1]](#footnote-1), South-East Asia e-commerce have hit an inflection point in 2018 and is projected to grow to a market size of USD 102 Billion by the year 2025. This reflects a 1,900 growth percent versus overall 2015 e-commerce market size of USD 5 Billion.

One of the main challenges facing e-commerce companies is to understand customers’ online journey and their intention to purchase. According to Invesp[[2]](#footnote-2), the average conversion rate of E-commerce website is 2.86% in Q2 2018. Every percentage point increase in conversion would greatly improve the overall revenue. By understanding customers’ online journey through the massive amount of data collected through web/app clickstream data, the company could formulate different strategies to each target group in order to maximize conversion and revenue, such as sending targeted ads or notification with promo codes.

In this project, we will use Google Merchandise Store clickstream data from Google Bigquery from the date range of 1 November 2016 to 30 July 2017. This project aims to build a machine learning algorithm to predict which visitors will make a purchase in the next 7 days.

# Data Source

The dataset from Google Merchandise Store’s Google Analytics contains 338 variables and is publicly available in Google Bigquery[[3]](#footnote-3). The dataset is nested at three different levels, 1. sessions level, 2. hits level, 3. product hits level.

To extract the dataset, we have to first create a Google Cloud Platform account. After which, the dataset can be extracted by incorporating SQL queries into the python codes in Google Colab.

One point to note is that the target variable (transaction) is highly imbalanced with only 3.9% of the non-bounce sessions making a transaction.

# Analytical Question

In order to extract meaningful data from the dataset, it is important to first understand and hypothesis what features will be important to predict transaction and concurrently, derive a meaningful analytical and business question. We tinkered with two analytical questions before arriving at the final one.

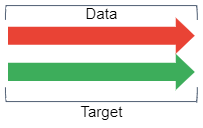
1. **Analytical Question 1**: Predicting if customers will make a purchase in period B, using data from period A

This analytics question is not viable as only 0.03% of customers who have activities in period A transact in period B (most customers will transact in the day itself). This leaves us with insufficient target variable to create a meaningful model.



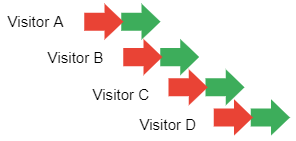
1. **Analytical Question 2**: Predicting if customers will make a purchase in period A, using data from period A

While this analytical question is a viable machine learning problem, the model built have very little business impact as business do not need a model to know who transact/did not transact.



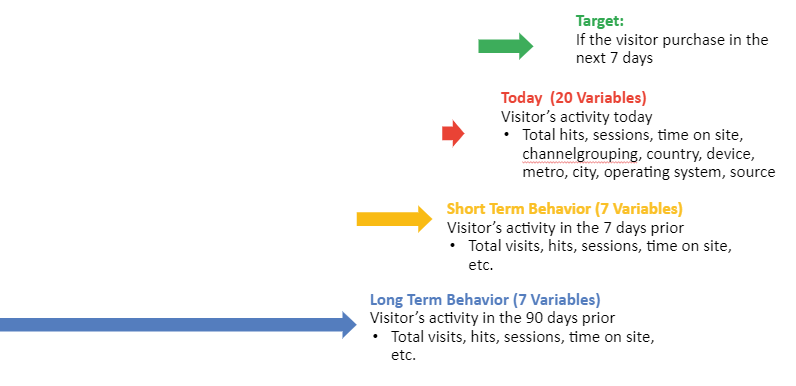
1. **Analytical Question 3**: Predict if customers will make a purchase in the next 7 days, using past data

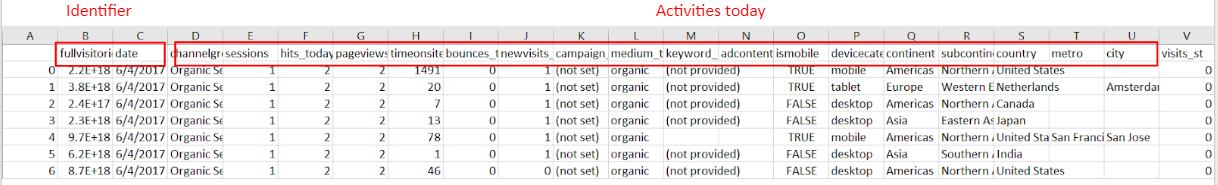
Finally, we settled on this analytical question, which is a workable machine learning problem with business value.

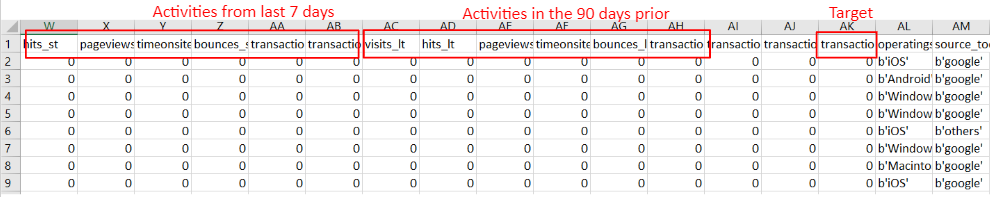


# Dataset

While the dataset contains a very large number of variables (338), we realized that most of the variables are either too granular or not relevant to the task. Using the AIDA (Awareness-Interest-Desire-Action)[[4]](#footnote-4) model as reference, we are convinced that a purchase decision is usually made over several days, and hence, short-term and long-term behavior of visitors will play an important part in conversion.

We settled on the short-term/long-term dataset architecture which include variables from the visitors’ session today, short-term behavior, long-term behavior, and using the 7 days forward looking conversion as the target variables. 

To create the dataset, we joined four different aggregations (Target, Today, Short-Term Behaviour, Long-Term Behaviour) together `fullvisitorid` and `date` as identifier. 



The dataset is then divided into train/validation and test dataset with two different time periods.

Train / Validation set: 1 March 2017 to 30 May 2017

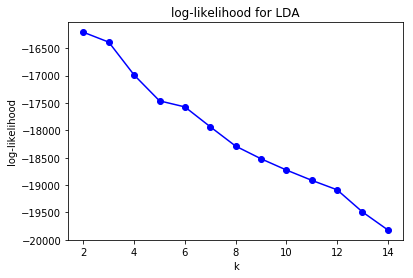
Test set: 1 June 2017 to 30 June

# Data Pre-processing

Under Data exploratory, it was noted only 3.9% of the customers in the training/validation data will make a purchase in the next 7 days which is relatively low.

As part of feature engineering, some categorical variables which have many classes were regrouped to reduce the computational time required for classification models as the matrix will be sparse after encoding. Data pre-processing was performed on three of such variables – ‘date’, ‘operatingsystem’ and ‘source\_today’. ‘date’ was changed from dd/mm/yy format to mm/yy format. ‘operatingsystem’ variable was regrouped into Windows, Macintosh, Android, iOS, Linux, Chrome OS (top 5 categories which account for most of the records) and the rest were classified as others. Similarly, ‘source\_today’ variable was regrouped into direct, partners, youtube, google, facebook and the rest as others. This helps to reduce on dimensionality.

Another variable ‘keyword\_today’ has 436 categories (including null) and due to the free text nature, it is difficult to classify based on top few categories. Topic Modelling on keyword\_today was performed using Latent Dirichlet Allocation (LDA) and with the aim to uncover similar topic clusters and reduce the dimensionality of this column.

LDA is a generative probabilistic three-level Bayesian model in which documents are characterised by random mixture over latent topics which follows Dirichlet distribution and each topic has a probability distribution over a set of words. The column was pre-processed by tokenisation, case-normalisation, keeping of alphabets using regular expression, removal of stopwords with NLTK’s stopwords and feature selection by removing words which have low frequencies (i.e. potential typo).

Elbow method was used based on log-likelihood (figure on the right) to determine the optimal number of clusters and 5 clusters were selected. An examination of the results show that the topic clusters are online targeting, t-shirts and stickers, google merchandise store and youtube. There are two clusters which are related to google merchandise store. The top words for each cluster can be found in appendix I. These reduced the number of categories from 436 to 5.

As many of the columns seem to have 0 values, Principal Component Analysis (PCA) was performed on the 23 continuous variables to further reduce the dimensionality. Minka’s MLE was used and the optimal number of components to keep was given as 19.

Encoding was performed on the categorical variables to convert them into dummy variables and minmax scaling was used to scale the data into range of 0 to 1. Train/Validation dataset was split into 70% train and 30% validation for binary classification.

# Performance Metrics

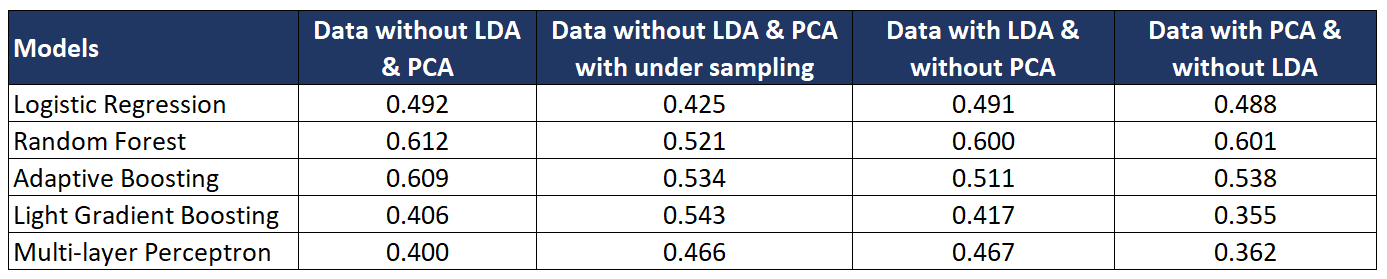
With the assumption that marketing cost for online marketing is relatively low versus the potential revenue made, classification models were evaluated based on F2-score. F2-score is a modification of F1-score where it considers both precision and recall in the measurement, but implicit computation will give a higher weightage to recall measure, i.e. the ability for the model to be able to capture as many potential customers as possible.

General formula of measure:

where for this project

# Model Performance

Five binary classification models – Logistic Regression, Random Forest, Adaptive Boosting, Light Gradient Boosting and Multi-layer Perceptron (Neural Network), and two outlier detection models – One Class SVM and Isolation Forest, were used for this project. For binary classification, k-stratified cross validation was used to avoid biasness and overfitting of the training data and randomized search was performed using a range of parameters to select the parameters which optimizes F2-score for each classification method (hyperparameter tuning). Regularisation was also used to prevent overfitting. i.e. L2 is used in logistic regression. As the conversion rate is relatively low at 3.9%, undersampling was also performed using ‘RandomUnderSampler’ from imblearn package. The table below shows the F2 score when the models were deployed on the Test dataset (i.e. 1 June 2017 to 30 June) for the various preprocessing methods.



# Model Evaluation

Random Forest with data without LDA and PCA obtained the highest F2-score for the test data out of all the combinations tried. The full results with precision, recall, accuracy, f1 and f2 scores for the Test Data can be found in appendix II. Generally, Logistic Regression has the highest recall but the lowest precision which explains for the lower F2 score for all the different data preprocessing methods. Random Forest and Adaptive Boosting did better in terms of recall while Light Gradient Boosting, a modified version of gradient boosting which uses techniques to eliminate instances with small gradients without compromising on accuracy, and Multi-layer perceptron did better in terms of precision. Also, it was noted that for some instances, multi-layer perceptron was unable to converge in 100 iterations which might also be a reason for the poor result.

The top important features from the winning models are number of hits, time on site and pageviews on the day of visit, coming from United States, number of hits in short term and long term, Macintosh operating system users, coming from direct source, using mobile device and visits in the long term. Interestingly, number of hits on the day of visit itself or during short-term and long-term periods (note that these three periods are independent) seem to be a higher contributing factor than number of visits. Also, even though PCA helps by reducing dimensionality, the resultant vector is not very interpretable and thus might not be as useful in terms of result interpretation, especially so when most of the important attributes are continuous.

# Anomaly Detection

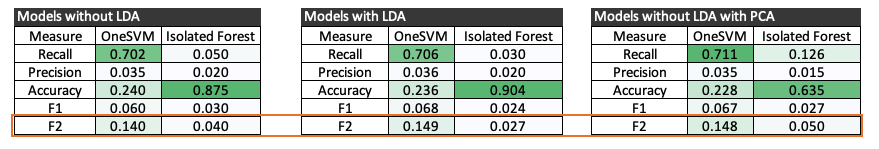
Due to the presence of imbalance of class values within the target data, whereby customers who bought items were the minority (3.9%), we wanted to investigate who they were anomalies.

We ran 2 anomaly detection models from sklearn: OneClassSVM which is an unsupervised outlier detection model and Isolated Forest which is a semi-supervised novelty detection model.

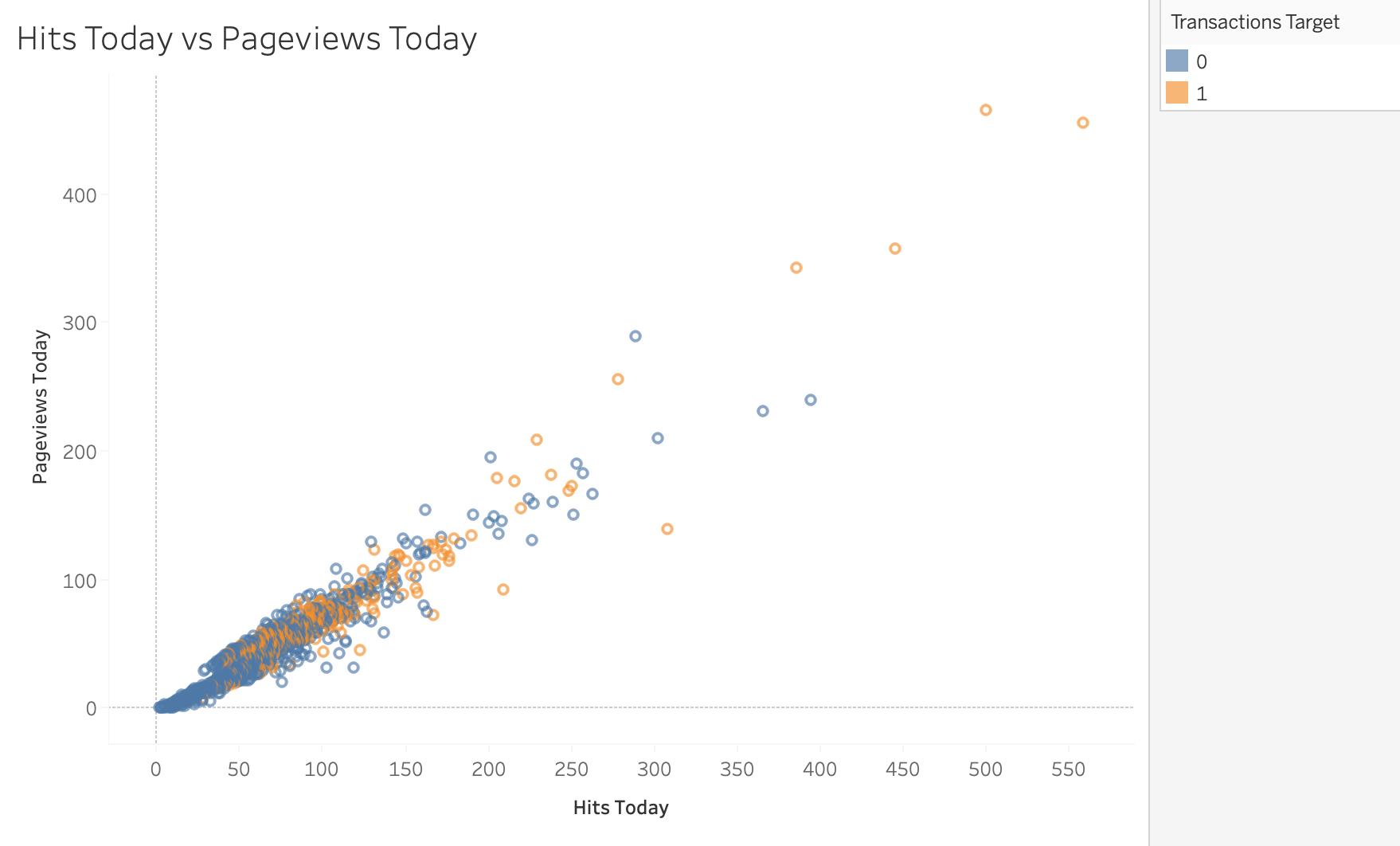
Anomaly detection aims to train on inlier data to obtain a model that can predict outliers. In our context, inliers are those customers that did not purchase anything (y=0) and outliers are those that made a purchase (y=1).

For the training data for anomaly detection, we combined training and validation set together and took the rows where y=0. Validation set was included as we thought more data would be better for the model. For the predict data for anomaly detection, we used test data where y=0 and y=1. To summarize, we trained on inliers and predicted on data that added in outliers.

We did not perform anomaly detection for the undersampled data. The results were not as good as the previous models, yielding an F2 score of at most 0.149. The breakdown can be seen below, where we tried without LDA, with LDA and without LDA with PCA. Furthermore, feature importance could not be extracted. It can only be extracted for linear kernels. As such we concluded that the customers who purchased were not anomalies.



To further investigate why the anomaly detection models failed to work, we plotted some of the best features (which was obtained from the other models) against each other. The majority of the orange data points which represent the y=1 class, are clearly clustered together with the majority of the blue data points which represent the y=0 class. Only a small minority of the orange data are outliers. This explains why the anomaly detection model did not perform well

.

# Error Analysis

To understand how well the winning model (Random Forest without feature engineering) perform, we did an investigation on the top 10 false positive results.

On the right shows an example of a false positive classified by the model. We observed that even though this particular false positive record is incorrectly classified, the session exhibited high intention to purchase, with visitor having visited the basket four times and was just a click away from completing his/her orders.

For the top 10 false positives, five sessions ended at the check-out page, two sessions eventually made a purchase after seven days, and three sessions ended without visiting the cart. We conclude that the model works reasonably well in estimating intention to purchase.

# Business Implication

This model can give e-commerce business a good insights on who are likely to make a purchase. Corresponding, business can leverage on this model to save digital marketing cost and increase revenue by sending targeted campaigns at customers.

**Digital Marketing** - With this information, businesses can narrow their targeting audience for digital marketing. Instead of sending remarketing advertisements to all visitors, if business only target visitors with more than 10% likelihood of making a purchase, they can reduce the total audience size by more than 90% and eliminate unnecessary marketing expenses.

**Actionable data at customer level** - By combining this output of this model with hits level data, businesses can understand visitors’ intention and send customized offers to individual visitors. For instance, if a visitor with a high likelihood of making a purchase ended the session at checkout page, the business can send a 10% discount promo code to this customers to push the customer down the purchase funnel.

# Future Work

Some future work we could pursue is feature engineering on the time component, for e.g. analyzing whether morning, afternoon, evening or night time will affect purchases. We can also try to cluster each country’s users into nationality, but that is subjected to availability of data. As most of our work was done at a session level, further analysis on the hits level can be done. Also, we could aggregate data from hits level and attempt to incorporate user’s web journey as part of feature engineering.

# References

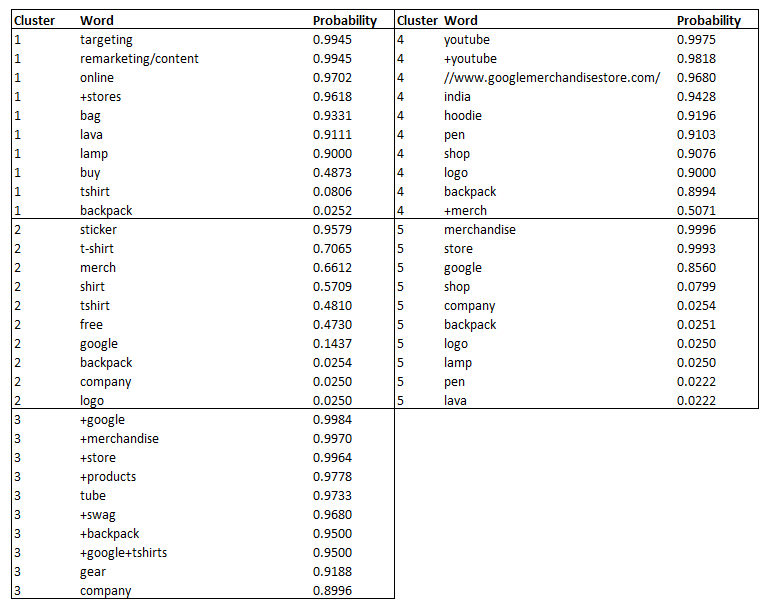
<https://scikit-learn.org/stable/modules/outlier_detection.html#outlier-detection>

<https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html>

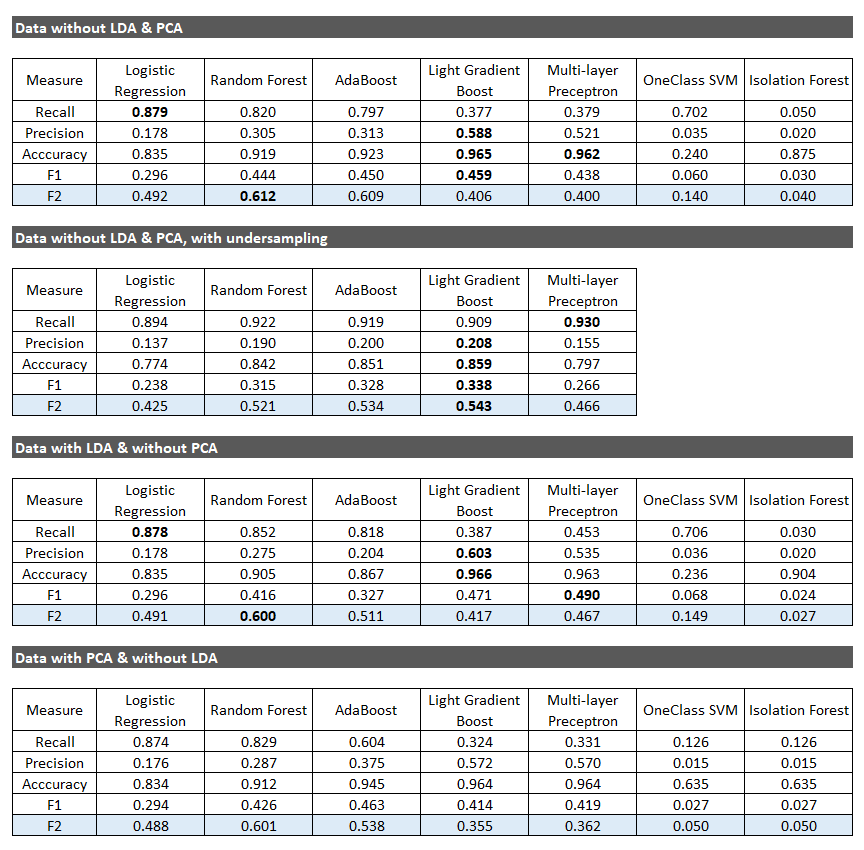
<https://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf>

# Appendix

1. Top words each topic clusters



1. Results for Classification and Anomaly Detection for various data pre-processing



1. <https://www.thinkwithgoogle.com/intl/en-apac/tools-resources/research-studies/e-conomy-sea-2018-southeast-asias-internet-economy-hits-inflection-point/> [↑](#footnote-ref-1)
2. <https://www.invespcro.com/blog/the-average-website-conversion-rate-by-industry/> [↑](#footnote-ref-2)
3. <https://bigquery.cloud.google.com/table/bigquery-public-data:google_analytics_sample.ga_sessions_20170801> [↑](#footnote-ref-3)
4. <https://belvg.com/blog/what-is-aida-and-why-online-stores-need-it.html> [↑](#footnote-ref-4)