Assimilation of Machine Learning and Cloud Computing for Supply Chain Industry.

DBS

Dublin Business School

excellence through learning

Dissertation submitted in part fulfilment of the requirements for the degree of Master's in Business Analytics

At Dublin Business School

Submitted By

Tirth Girish Pipalia

Student id: 10524692

DECLARATION

I, Tirth Girish Pipalia do hereby declare that the dissertation entitled "Assimilation of Machine Learning and Cloud Computing for Supply Chain Industry" has been undertaken by me for the award of Master of Science in Business Analytics. I have completed this study under the guidance of Dr. Shahram Azizi Sazi, Dublin Business School, Dublin, Ireland. The information given in the report is authentic to the best of my knowledge. This project report is not submitted anywhere or published any time before. All the ethics, procedures and guidelines have been followed properly while preparing thesis.

Signed: Tirth Girish Pipalia

Date: 20/08/2020

ACKNOWLEDGEMENTS

This dissertation would be near to impossible without constant support and encouragement of several people.

First and foremost, I would express deepest gratitude and sincere thanks to my Supervisor and Professor Dr. Shahram Azizi Sazi. He has constantly provided his relentless knowledge and insights through is experience in the field of Machine Learning and Data Analytics at the cost of his time. The guidance and suggestions helped me to improve the quality of my research. He helped me to overcome several obstacles during the phase of dissertation and always supported to push my work to higher level. Without his caring nature this project would not have been reached to completion.

In the journey towards this degree I would not miss the chance to thank and appreciate the support of my Parents and family members along with my friends and classmates.

I would also extend my gratitude and warmest wishes for the Dublin Business School which has provided me the privilege to meet some great people. I would like to thank all the professors and management faculty for making this journey wonderful and cherish able experience of my life with deepest gratitude.

Sincere thanks to people and Government of Ireland for hospitality and caring attitude for international students.

ABSTRACT

During the era of IoT and Big Data as an emerging technology, it has major impact on all the business sectors. This technology results in generating massive amount of electronic data which contains valuable information. To extract and analyse this information at greater scale the only choice is Cloud computing and Machine Learning technology. The goal is to identify best process for Fraud Prediction and Sales prediction. Ten Machine Learning models are implemented for solving this business question. RandomizedSearchCV is implemented for hyper parameter tuning and time required by model for training and prediction is also evaluated. Even SMOTE is applied for imbalanced dataset. Classification models are validated based on Recall, F1, Confusion Matric ROC-AUC values. Regression Models are evaluated based on MAE and MSE score. Research is conducted using Amazon Web Services and python for predictive analytics by assimilating Machine Learning and Cloud Computing technologies.

Keywords: Amazon Web Services, SMOTE, RandomizedSearchCV, Predictive Analytics, Machine Learning

Contents

List of Figures

List of Tables

List of Equations

Acronyms

ML-Machine Learning

IoT-Internet of Things

AWS-Amazon Web Services

GBM- Gradient Boosted Model

LDA-Linear Discriminant Analysis

LASSO-Least Absolute Shrinkage and Selection Operator

MAE-Mean Absolute Error

MSE-Mean Squared Error

RMSE-Root Mean Squared Error

RR-Ridge Regression

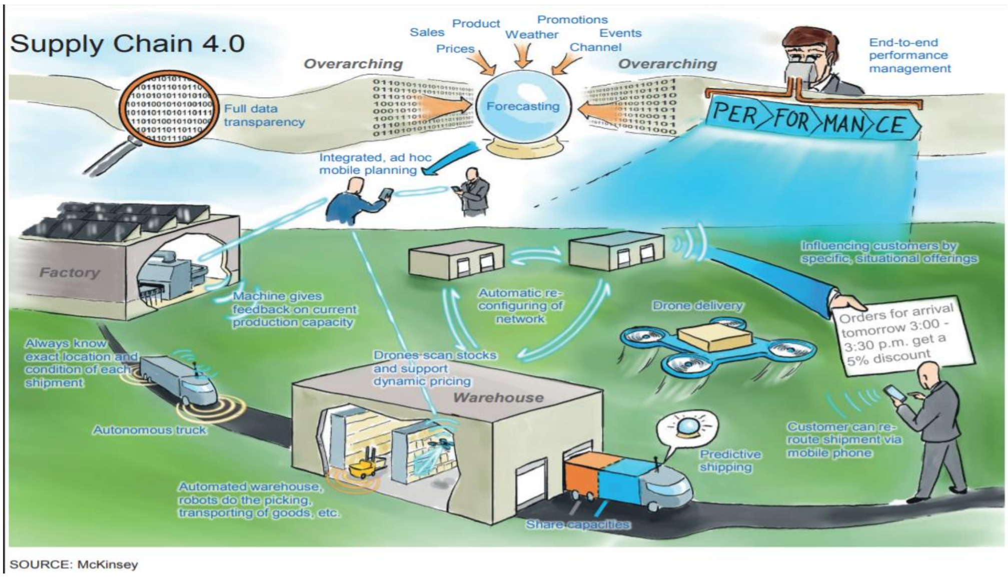
1. Introduction

1.1 Background

Last decade of the 21st century has addressed and witnessed the marvellous growth and expansion in all the major economy-boosting sectors into international locations, especially for apparel industries, computer, and automobile (Taylor, 1996). Supply chain no longer being a domestic phenomenon it transcends national boundaries, which increases the challenges related to it. (Meixell and Gargeya, 2005) has explained well by comparing various literature and research paper published to handle issues confronted by this industry, showing how significant it is for globalization. It provides a clear picture of the growth and vital role it will play for the overall development of emerging industries like electronics manufacturing, fibre and textile, apparel. Present nature of trade and commerce causes supply chain to be globally interconnected, distributed, organised service. This leads to highlight the single most vulnerability or weakness it faces that is the ability for having an adverse effect on activity which is thousands of mile away (Zage, Glass and Colbaugh, 2013). As described in this article (Alicke, Rachor and Seyfert, 2016) application of advanced robotics, Internet of Things, advance analytics of Big Data for Supply Chain Management which is implemented by automation of anything, placing sensors, analysing everything, creating brand network/bubble for improving customer satisfaction and performance of the company significantly. This will particularly be the biggest worry for the industry as the vast volumes of data will be generated when things are integrated with the mentioned technology. This data cannot be just stored to get drowned under the flood of data. Data needs to be analysed while it continues to grow at unprecedented rates.

This can be done by implementing computational methods, tools, and techniques. Therefore, (Constante-Nicolalde, Guerra-Terán and Pérez-Medina, 2019) predictive modelling analysis is implemented to predicts the sales and to predict fraud, so significant steps can be taken for the betterment of Supply Chain Management. As discussed by (Mojtahed, 2019) Fraud is defined as making a false representation so that unveiled information is relevant in such a way that the abuser gains misappropriate benefit or financial gain. These activities can range on a wide spectrum from insurance fraud, credit/debit fraud, online auction, to food fraud. As it can be remarkably diverse depending on the sector it is being occurred. However, there are still common and widespread themes that can be implemented to detect anomaly like Fraud in Transaction. Studies have shown the frequency of fraud is rising which leads to on an average 6.5% loss from companies expenditure and income (Blakeborough and Correia, 2017). Another important thing for the supply chain industry is to manage the delivery of goods at a huge scale and with complexity which can be nerve-wracking and stressful at times of peak demand. Machine Learning helps e-commerce organizations to manage supply chain efficiently. Also, to understand sales performance in e-business operations, which is the most important challenge so that the supply chain is managed efficiently (Chong et al., 2017). E-business plays a crucial role in today's economy as customers highly rely on the e-business marketplace this leads the organizations into the competition of surviving in the e-business environment(Li et al., 2016).

As the research continuous, exploration for solving the above issues for the betterment of the supply chain industry leads to the implementation of various machine learning and cloud computing technologies. This makes us divide the research into two major part 1) Fraud Prediction and 2) Sales Prediction using the same dataset. Due to the size and variables of the data, it is possible to extract several significant insights which will help to boost the business. The aim is to achieve the objectives with the implementation of Business Analysis and ML techniques on Big Data. So, to handle the big data we need infrastructure which can perform the computational task at a higher level and therefore as discussed ahead in the Methodology section Amazon Web Services (AWS) for Cloud Computing technologies is utilized. In Machine Learning technique for classification Random Forest, Decision Tree, AdaBoost, Light GBM, Linear Discriminant Analysis (LDA) is implemented and evaluated based on Confusion matrix, Accuracy, Recall, F1 score. For regression LASSO, Ridge, XGBoost, GBM and Linear Regression is implemented which is evaluated using metrics like MAE, MSE and RMSE. All this is performed using python programming language because it has some great packages and libraries which helps in implementing ML models and for Exploratory Data Analysis (Raschka, Patterson and Nolet, 2020).


Supply Chain 4.0
Sales
Product
Promotions
Weather
Events
Prices
Channel
Overarching
Overarching
End-to-end performance management
Full data transparency
Forecasting
PER>FOR>MAN>CE
Integrated, ad hoc mobile planning
Factory
Machine gives feedback on current production capacity
Automatic re-configuring of network
Influencing customers by specific, situational offerings
Drone delivery
Orders for arrival tomorrow 3:00 - 3:30 p.m. get a 5% discount
Always know exact location and condition of each shipment
Drones scan stocks and support dynamic pricing
Warehouse
Autonomous truck
Customer can re-route shipment via mobile phone
Predictive shipping
Automated warehouse, robots do the picking, transporting of goods, etc.
Share capacities

SOURCE: McKinsey

Figure 1 Supply Chain 4.0

1.2 Research Questions and Objectives

- To what extent cloud computing and machine learning techniques be effectively assimilated with supply chain industry to predict fraud and sales?
- What will be Obstacles and Opportunities for applying machine learning and data mining techniques in the supply chain industry?
- Which variables and features are important for Fraud Detection and Sales Prediction?
- Which model is the best fit for Fraud Detection and Sales Prediction?

These questions assist in defining the objectives:

- To discover common dataset for Fraud and Sales prediction so that inter-relationships between of features can be highlighted with respect to fraud and sales.
- Obtain critical and comprehensive literature review for related research work. From concrete sources like Journal Articles, ML Conferences, Books, Research Papers.
- Perform EDA (Exploratory Data Analysis) for generating preliminary insights from the dataset.
- Setting up the hyper-parameters for essential Machine Learning model using RandomizedSearchCV.
- Feature engineering to obtain the impact of important features in the dataset.
- Evaluate and validate results of ML models using metrics like Accuracy, F1, Recall Score and RMSE, MAE values.
- Setting up the services of AWS which will make the implementation of predictive analysis easier and efficient.
- Setting up AWS S3 permission and installation of essential packages for performing cloud integration task.

1.3 Dissertation Outline:

Thesis report comprises 6 chapters excluding **Chapter 1: Introduction** following sections can be identified as:

Chapter2: Literature Review

It briefs about the research and its topic by providing domain-specific knowledge, identifying subjects where research and findings are available. Also helps in preventing duplication of research and helps to give credit to other researchers whose work clarified research problem and acted as guidance throughout the research. The funnel-like approach is implemented in writhing of this section where it initially explains supply chain industry and narrows down to specific approaches taken in this research. It has summary and findings of various machine learning-based classification and regression papers.

Chapter3: Research Methodology

It explains the specific procedures or techniques followed to complete the research and to achieve the objective. Describing in detail tools and design followed in the experiment performed. Shows the implementation of cloud computing and integrating various tools like AWS S3 bucket, python packages.

Chapter4: Implementation

This part explains the outcome of the followed procedures in Chapter3. Provides the description of activities performed and its obtained result. Explains process like fetching the dataset, preparing dataset, and developing Machine Learning model for analysis, feature selection and model hyper parameter tuning. Throws light on methods undertaken to reach the obtained results and conclusions.

Chapter5: Analysis and Findings

This section helps in further churning the raw result into a business solution. Providing insights about weakness and strength of various model for achieving the desired goal. The outcome is then evaluated for selecting the best fit model for fraud detection and sales prediction. In this section solutions for the desired result is gleaned.

Chapter6: Conclusion

This part provides a summary of the whole dissertation and indicates the direction in which research can be further extended. And helps in identifying whether the research has fallen for the desired outcome of achieving the objectives.

2. Literature review

(Schoenherr and Speier-Pero, 2015) provides insights for the future potential of Supply Chain Industry concerning the implementation of Data Science and Predictive Analytics and Big Data. It aims to answer the following questions: 1) what is the underlying motivation for using predictive analytics on Supply Chain Management? What are the stakes (benefits and barriers) for successful implementation of SCM predictive analytics? The methodology used for the research was to conduct a large-scale survey among the SCM professionals to indicate their current status of whether (1) *No current use but plans for the future*, (2) *using at some extent*, (3) *they use analytics to a great extent*, (4) *they are not familiar with analytics*. As the focus was on the use of SCM predictive analytics for solving the business problem, respondents whose data indicated that they are not familiar or not interested in using this technique in future was removed. Based on the data collected for motivation to use SCM predictive analytics three groups where created 1) *No current use but plans for the future* 2) *To some extent* 3) *to a great extent*. Analysis of Variance (ANOVA) is used to calculate whether the means of three group are different and F-test to test the equality of means statistically. ANOVA uses the F-test to determine variability between group means. If the ratio is sufficiently large it concludes means of the group are not equal. The outcome of the paper has intrigue insight by encouraging the fact that more than 40% of the professional respondents actively use analytics in their routine business (Group 3). Whereas 8.7% plan to use this approach in the coming future (Group 2). And the most surprising fact was introduced that one third (28.4%) are not familiar with the analytics domain. The Primary objective was finding benefits and barriers of SCM predictive analytics: where benefits are decision-making capabilities, ability to improve supply chain efficiencies, enhanced demand planning capabilities, improvement in supply chain costs management, increase in work transparency and the creation of enhanced bargaining position with suppliers. And barriers are employees being inexperienced (so the need for training), time constraints, lack of integration with current systems, the costs of currently available solutions, change management issues, inadequate material of predictive analytics in SCM, along with the perception of SCM predictive analytics as an overwhelming and difficult task to be managed

Another research by (Lata, Koushika and Hasan, 2015) in which they compared several Fraud Detection Techniques in industries like Healthcare, Insurance, Credit Card, Banking, Telecommunications and Computer Intrusion. For Machine Learning based Fraud Detection, this paper compares Bayesian Networks, Markov Models (Markov chain and hidden Markov model), Neural Networks Fuzzy Logic Techniques, and Genetic Algorithms. Where in the healthcare industry Neural Network and Decision Tree is the most researched subject for fraud detection. Neural Networks are universally used in the healthcare industry due to its capability to handle complex data structure and non-linear relationships between the variables. Decision Trees which express dependent and independent variable in a tree-like structure and extracts the classification rules using IF-THEN expression.

As per the research conducted by (Ghatasheh, 2014) for the era of stringent and dynamic business environment Business analytics and the combined expertise of Machine Learning and Computer Intelligence, it makes the task of detecting fraud or risk for Banking and Funding Organization simpler. The German Credit dataset is used for research and Random Forest Trees is applied for the analysis. As Random Forest Trees are based on the predictions of several trees which tolerate more noise compared to 'AdaBoost' whilst utilizing random feature selection technique in splitting the trees. 10-Fold Cross-Validation method is implemented using different tools that are Keel, Heuristic Lab, and Weka. And for the benchmark Evolutionary Product, Neural Network for Classification (NNEP-C) is used in Keel. In Weka algorithms used are AdaBoost with C4.5, SVM using Linear Kernel, Real AdaBoost with C4.5, C4.5Decision Trees, Bagging with C4.5, Decorate with C4.5 and Dagging with C4.5. Similarly, for Heuristic Lab the algorithms are Genetic Programming, Neural Network Ensemble Classification (NNEC) and Multinomial Logit Classification (MN Logit). However main research was conducted using Heuristic Lab's Random Forest Tree modified for classification. For this modification 3 parameters are essential to be considered: (1) r - it is a ratio between 0 to 1 (2) m – which is number of attributes and (3) nT – number of trees. These parameters help to achieve an optimum result which is measure by evaluating actual target variable using Confusion Matrix. And therefore, for the result after tuning Random Forest tree where nT = 200, r=0.3 and m=0.5. Best Value for Sensitivity, F-Measure, Accuracy is 0.923, 0.857 and 0.784 respectively where Precision score for MN Logit algorithm at default tuning is best i.e. 0.883 and for Random Forest Tree it is 0.800. Hence showing that Random Forest Trees is a promising algorithm for solving Business problems.

Two types of credit card fraud which are: 1) Application-level fraud and 2) Transaction-level fraud are performed and to detect this, the feature selection method is implemented by (Singh and Jain, 2019). They have used J48, AdaBoost, PART, Random Forest, Decision tree and Naïve Bayes machine learning models. To compare the performance of this models 5 parameters are used namely Accuracy, Recall, MCC, Precision, Specificity, and Sensitivity. German credit dataset was used from UCI repository which contained 7 numerical and 14 nominal/categorical values where all the experiments on models focused on 10 folds cross-validation due to imbalanced dataset. Filter method and wrapper method were used for feature selection. The result showed that J48, Naïve Bayes, AdaBoost and PART classifiers had improved result when Filter and Wrapper method was applied however, the case for Random Forest was the opposite. Prediction accuracy for J48 ranged from 70.5 to 72.5 and 74.6, for AdaBoost from 69.5 to 69.6 and 74.5, for PART from 70.2 to 70.4 and 71.9 was increased by filter and wrapper method. Precision for J48 (from 0.76 to 0.782), AdaBoost (0.737 to 0.776) and for Random Forest (from 0.783 to 0.787) had enhanced. Therefore, concluding the outcome of research where Accuracy for J48 and PART classifier and Precision for J48 and AdaBoost classifier is significantly increased when information gain method and wrapper sub-select method is used.

(Hu, Chen and Zhang, 2019) has employed Tree-Based classification algorithms 1) Random Forest Tree 2) LightGBM to overcome the challenge to control the trade-off between False-Positive (miss detection) and Negative Rates (false alarm). Where in the finance industry False-Positive refers to not detecting fraudulent transaction leading to huge pecuniary loss and potential reputation loss as a result. Therefore, to control this tragedy means, asymmetric control is needed which can be implemented by adapting the Neyman-Pearson Classification Paradigm it prioritizes control of asymmetric errors. Hence after implementing Random Forest and LightGBM performance result under classical paradigm is 0.9995, 0.2256, 0.000106, 1.2560, and 0.9812 as of Accuracy, FRP, FNR, NPerror, AUROC respectively for LightGBM and 0.9994, 0.2744, 0.000141, 1.7441, 0.9750 as of Accuracy, FRP, FNR, NPerror, and AUROC respectively for Random Forest. This shows results for LightGBM is outperforming Random Forest. For most of the cases lower the AUROC value larger the NP error this is due to combined measurement metrics used for the performance of FPR and FNR. Also, NP error is robust against unbalanced classification results. This makes it best suitable option to evaluate different classifiers. Computation time for both the algorithm was also compared where 200 base estimators with a maximum of 10 levels on Mac laptop with 16GB RAM Intel Core i7 CPU at 2.7GHz for each algorithm excluding the cross-validation time over 10 runs. LightGBM: 3seconds and Random Forest: 154 seconds. Therefore, justifying LightGBM is the best algorithm cooperatively to Random Forest Tree when used under the Neyman-Pearson classification paradigm.

As discovered by (Rushin et al., 2017) where they explored the possibilities of algorithmic impact on the predictive power across 3 supervised classification machine learning algorithms: 1)Logistic Regression, 2) Gradient Boosted Tree, 3) Deep Learning(Neural Networks). The logistic regression, GBT, and deep learning models are compared across six different feature sets created using the two feature engineering methods (features created using domain expertise and the auto-encoder features). Performance for this models was gaged by k-fold cross-validation randomly separated in parts. The auto-encoder contained a single hidden layer with 50 nodes where reconstruction error was 0.0006. After searching through stipulated hyper-parameters deep learning model had 2 hidden layers with 50 nodes in each layer. After comparing the result of 3 models with all the 6 feature sets, study shows that based on 5-Fold Cross Validation AUC score of Deep Learning model has the largest value for the majority of the feature sets ranging from 0.875-0.773 whereas Gradient Boosted Tree has second highest values ranging from 0.864-0.769 for 6 feature sets. This also describes the importance of feature engineering using an auto-encoder and domain expertise because models can see the boost of 1-4% in AUC values. This tells creating features using domain expertise has more impact than the unsupervised method of feature engineering.

(Wei et al., 2019) has introduced a new model for solving the credit scoring based on backflow learning in which noise adapted 2-layered ensemble model is developed using five widely used base classifiers which are Random Forest, Gradient Boosting Decision Tree, Linear Discriminant Analysis, Extreme Gradient Boosting and Support Vector Machine. In which outlier score for each data value is measured to identify noise in data, one which is positive are then subsequently boosted into the training set through noise-adapted training set. They have evaluated the model on three different datasets obtained from UCI repository which has a different dimension of input space: Australian dataset has 15, Japanese has 16 and Polish has 65. Also, SMOTE is implemented for balancing the unbalanced classes of the training set.

Performance measures used were F1, Accuracy, Precision, and AUC. Base classifiers for which parameters where tuned had values as for SVM classifier kernel was set as Radial Basis function with penalty parameter C = 10 for soft margin. For Gradient Boosted Decision Tree number of tress and a maximum depth of tree was set as 0.1. For XGBoost number of trees was 200 where maximum depth for a tree was 6 and learning rate = 0.01. Random Forest classifier had 100 trees and each tree had the depth limit of 3. Best ensemble model was XGBoost base classifier which had an accuracy of 0.87 for Australian data, 0.89 for Japanese and 0.97 for Polish dataset. LDA was best base classifier when excluding ensemble models which had Accuracy values of 0.85 for Australian, 0.87 for Japanese and 0.95 for Polish dataset.

As the comparative study performed by (Alfaro et al., 2008) where the issue of forecasting corporate failure is addressed by evaluating the output of AdaBoost and Neural Network by emphasizing Type 1 error which in this case is when a firm which will possibly be failing in the future is classified as healthy. Along with that novel measure for the importance of variable to facilitate model interpretation is calculated. In training dataset 472 observation are present for each healthy firm and failed firm and for test dataset 236 firms with an equal number of healthy and failed firms. The AdaBoost classifier is built which has 100 trees with maximum depth=2 for pruning hence having 2.523% and 12.712%, Type 1 and Type 2 errors respectively for training set making overall 7.627%. For the test set 3.390% and 14.407% of Type 1 and Type 2 errors respectively combining to 8.898%. For ANN Type 1 and Type 2 errors are 4.025% and 17.585% respectively for training dataset accumulating to 10.805% and for test set Type 1 and Type 2 errors 7.627% and 17.797% respectively making 12.712% overall error per cent. AdaBoost strategy which implemented combining single trees achieved a 30% reduction in test error compared to an individual neural network. Which confirms that AdaBoost has outperformed Neural Network.

As proposed by (Cheriyan et al., 2018) where different models are applied and the result of which is summarized in terms of reliability and accuracy to efficiently predict and forecast the sales. Data implemented is obtained from an e-fashion store having records for three consecutive years from 2015-2017. Three different machine leering algorithms are used 1) Generalized Linear Model (GLM), 2) Decision Trees (DT) and 3) Gradient Boost Tree (GBT), based on prediction performance and empirical evolution 64%, 71% and 98% respectively is the best fit accuracy of models on the dataset. 100% accuracy can be possible for GBT to be achieved if implemented with further improvement by using models such as Grabit and Tobit to analyse. Precision for GLM=5.36, DT=11.24, GBT=50, Error Rate for GLM=36, DT=29, GBT=2, Recall score for GLM=0, DT=16.61, GBT=50 and Kappa for GLM=0, DT=.501, GBT=0.962. Showing clearly that Gradient Boosted Tree (GBT) stands out as a pioneer model with the highest accuracy and minimum error rate.

As implemented in the research paper by (Viktorovich et al., 2018) where classic machine learning algorithms are applied to The Ames Housing Price dataset (De Cock, 2011). Which is much more advance and modernized extended version of frequently cited Boston Housing Dataset. The gained solution of ML models is evaluated on Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price. Upon descriptive analysis 'Alley', 'FirePlaceQu', 'PoolQC', Fence, and 'MiscFeature' are the categorical variables having largest missing values hence indicating that majority of the houses do not have 2nd Garage, tennis-court that is covered by the ‘MiscFeature’, elevator, swimming pools, fence, alley access and shed also showing largest co-relationship between target variable(sales price) and house area ‘GrLivArea’. LASSO regression algorithm was used for which accuracy is significantly leveraged by regularization parameters that were set as α = 0.0003. For Elastic Net Regression which is like LASSO except using L2 penalty term along with L1 penalty term and for which α = 0.005 and γ = 0.13 is set. Also, Gradient Boosting of regression tree is applied where 100 trees with a maximum of 20 depth limit were implemented. Multilayer perceptron regressor implementation of Neural Network regressor with 3-layer perceptron (140, 70, and 25) neurons on each layer with Logistic Activation function and LGBFGs optimizer was implemented. Where the cross-validation score for LASSO=0.11139, XGBoost=0.13058, XGBoost with logit transform=0.12986, Elastic Net=0.11203 and Neural Network=0.11787 and the Ensemble of LASSO, XGBoost, Elastic Net, NN=0.111.

The research performed by (Ahn *et al.*, 2012) in which they have addressed two critical issues regarding ridge regression i.e. when to implement and how to improve the performance of Ridge Regression Model. To solve the confusing episodes of which model to be implemented when non-linearity present in the data. Ridge regression is coupled with a genetic algorithm named as GA-Ridge model. For evaluating the forecasting performance 3 forecasting models are used: Multiple Linear Regression (MLR), Pure Ridge Regression, and ANN where the performance of MLR and ANN was litmus rule for using GA-Ridge because Ridge regression is preferred when smaller β's value is expected. Neither ANN nor MLR excels the other model that significantly as mentioned in Remark 1 of the paper. Distance metric used for evaluating performance of 3 model show numbers as: 1) RMSE for GA-Ridge=0.0074, MLR=0.0104, Ridge Regression=0.0088, ANN=0.0110. 2) MAE for GA-Ridge=0.0055, MLR=0.0086, Ridge Regression=0.0069, ANN=0.008. 3) MAPE for GA-Ridge=239.66, MLR=304.91, Ridge Regression=244.72, ANN=291.05. Showing that GA-Ridge is superior compared to other models.

As researched by (Jain, Menon and Chandra, 2015) XGBoost, Random Forest Regression and Linear Regression is evaluated for forecasting sales on the data of retail chain shop of Rossmann –Germany's second-largest drug store chain. In the dataset top 5 highest relative importance variables are Store-11101946.0, Promo-1639790.75, DayOfWeek-713757.8750, Month-375840.531250, and CompetitionSinceMonth-270933.468750. The graph suggests that sales on Sunday are better compared to the rest of the week. Also showing a direct correlation between several customers and the store sales. RMPSE results after implementing three models on the test set are 0.15672 for Linear Regression, 0.13198 for Random Forest Regression and 0.10532 for XGBoost. Showing that XGBoost model performed best at the prediction of sales.

Researchers (Huang et al., 2020) have aimed to develop an intelligent model for the prediction of blood pressure during haemodialysis in which five algorithms were implemented and comparative study was performed. Two models were ensemble tree-based model Random Forest (RF) and Extreme Gradient Boost (XGBoost), other three are Support Vector Regression, Linear Regression and Least Absolute Shrinkage Selection Operator (LASSO). This was evaluated on the bases of R2, MAE and RMSE score where dataset used contained 7,180 and 2065 Blood Pressure records for training and test set, respectively. Also, correlation coefficients for the data showed 10 features as positive and rest 10 as negative. An important finding was that RF had the lowest RMSE and MAE in testing dataset 16.24 and 12.14 respectively when compared to XGBoost RMSE=17.65 and MAE=13.47 which performed well in the training set. Results showed that ensemble models performed well on training set where the value of R2 MAE and RMSE for RF and XGBoost is 0.95, 6.64, 4.90 and

1.00, 1.83, 1.29, respectively. SVR has R2, RMSE, MAE values as 0.78, 12.58, 8.57 respectively for Linear Regression and LASSO it is R2=0.59, RMSE=16.68, MAE=12.90 and R2=0.60, RMSE=16.92, MAE=12.87, respectively.

Cloud computing is a technology developed for enabling convenient, ubiquitous, pay-as-you-use, on-demand internet access to inter-connect or share configurable computing resources. (e.g. Services, Storage, Functions, Network, Applications, Servers) (Marston et al., 2011). This tools and services help organization or individual to rapidly provision and release the requirement to run and setup whole Internet network setup with minimal management effort or human interaction. (Mell and Grance, 2011) Cloud Computing technology comprises 3 service models: 1.Software as a Service (SaaS), 2.Platform as a Service (PaaS) and 3. Infrastructure as a Service (IaaS). With 4 deployment Models 1) Private Cloud, 2) Community Cloud, 3) Public Cloud, 4) Hybrid Cloud. This technology is useful for any organisation in three ways as said by (Velte, Velte and Elsenpeter, 2009) 1) Compute Clouds: Which will provide highly scalable, on-demand resources like Amazon EC2, Berkeley Open Infrastructure for Network Computing(BOINC), Google App Engine. 2) Cloud Storage: This allows user to synchronise and store the data to an online server without any need for File Management, Collaboration, Syncing, Administration and physical Security, File Sharing. Example of a few such services is Amazon RDS, Amazon Aurora, Azure Files, Azure Blobs, and Google. Therefore, combing Big data and Cloud Computing can provide benefits for Business Intelligence as mentioned by (Balachandran and Prasad, 2017) which are:

1) On-demand self-service: This helps to rapidly expand or shrink the deployed resources, without human intervention at a click of a button saving valuable time and human resources for basic unproductive actions.
2) Data and information over the net: For a huge multinational company having its establishments in various geographical locations, combine data from several locations can be centralized and synchronize. Which allows to access this data and resource from any remote location and providing the flexibility to use any device like Laptop, Mobile, iPad, etc. to access it.
3) Resource Pooling: Most profitable and advantageous option for any organization using multiple and huge quantities of storage, memory, graphics, VMs, servers, etc.

resources is to combine the billing and usage of this resource and operate as a whole single package.

4) Rapid Elasticity: Hardware and Software resources efficiency can be increased or decreased within a few minutes time instead of hours or days in some cases. Which provides the customer's independence for using the resource for any quantity and at any time.
5) Cost-effective: All the resource under the umbrella of cloud service provider can be monitored and set for a specific threshold which will indicate the user if that limit exceeds. Hence providing control on the budget and finance of the IT resource for an organisation.

Amazon S3 or Amazon Simple Storage System was launched on March 14, 2006, by AWS(Amazon Web Service) which is a subsidiary of Amazon ('Amazon S3', 2006). Amazon.com being one of the largest and advance e-commerce web application which is spread globally used scalable storage infrastructure which was implemented and launched commercially as Amazon S3. This allows customers and industries of all size to store their data without fretting about location, security, cost, availability of data. It is designe*d (Cloud Object Storage | Store & Retrieve Data Anywhere | Amazon Simple Storage Service (S3), 2020) for achieving 99.999999999% (119s) of durability and 99.95% to 99.99% of avaibality(Persico, Montieri and Pescapè, 2016). It stores data for millions of applications such as mobile applications, web apps, and websites, backup and restores data of servers, archive, big data analytics, IoT devices, Enterprise applications. S3 stores data in object form which are organized into buckets, this object has a unique user-assigned key. Buckets can be accessed globally using Amazon Management Console, programmatically using AWS SDK or by Amazon S3 REST API (Application Programming Interface). It can be integrated with several other services of AWS like Amazon RDS, Amazon Redshift Spectrum, Amazon Athena which helps to provide query-in-place services to customers which improve query performance up to 400% (Amazon S3 Features - Amazon Web Services, 2020).*

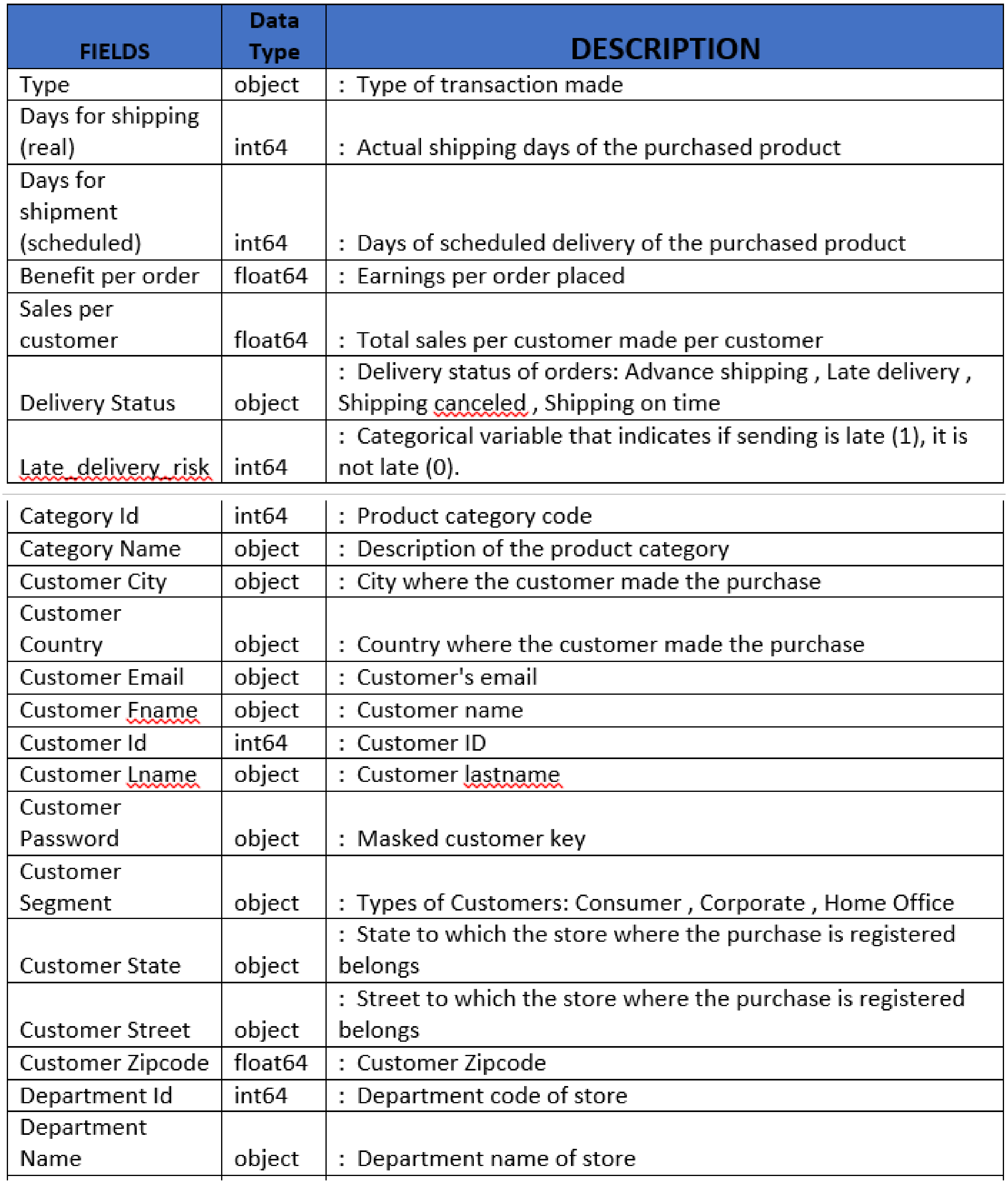
3. Research Methodology

3.1 Data Collection

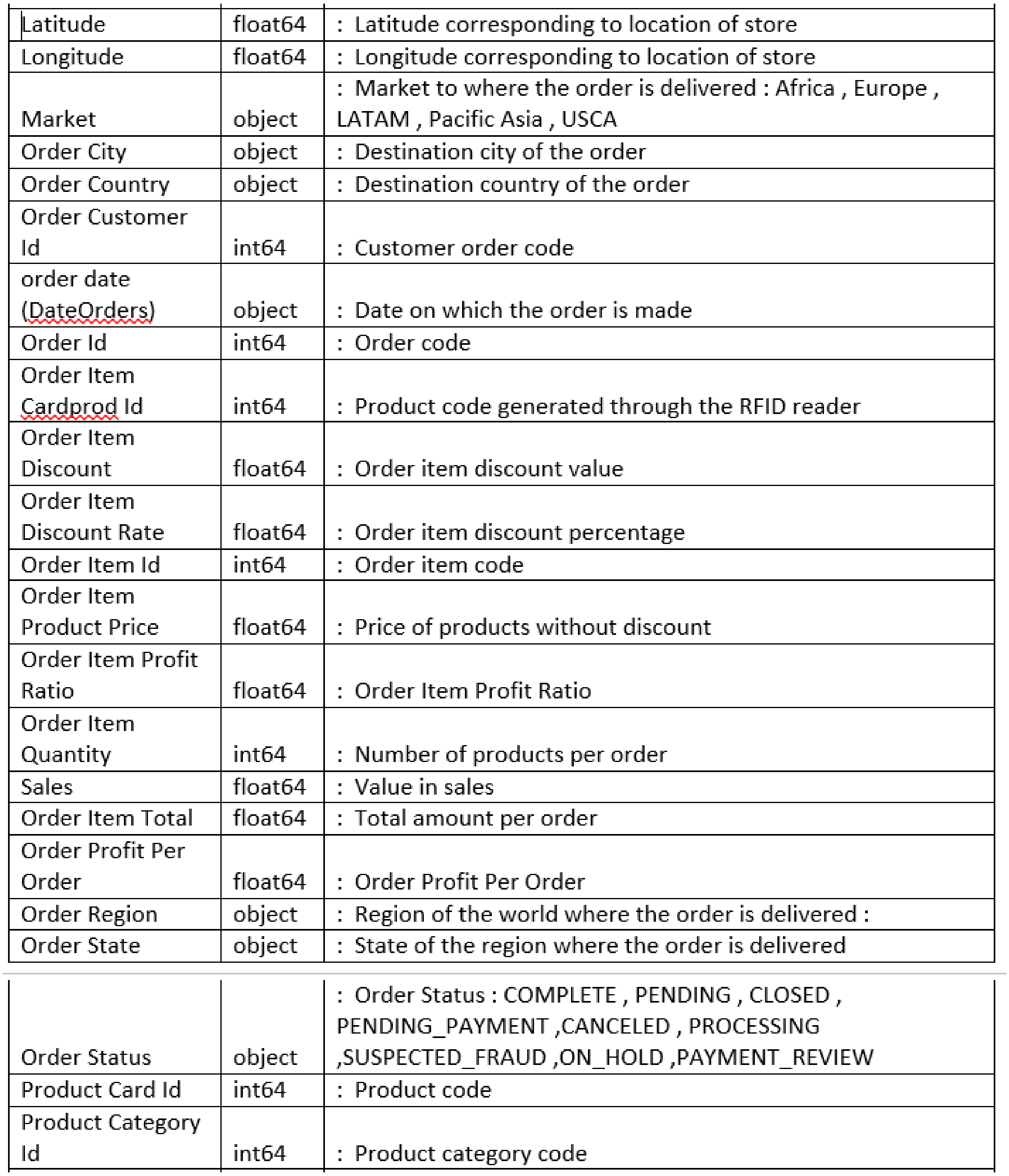
The dataset used for research is the fusion of primary and secondary data as it was collected by other people who are not involved in this research however data is utilized for achieving the objective of this research. Dataset was first uploaded on (Share & Manage Research Datasets - Mendeley, 2019) based in London UK, provides research and academic services developed by (Elsevier | An Information Analytics Business | Empowering Knowledge, 2008). (Constante, Silva and Pereira, 2019) are the contributors of the dataset who have uploaded 5 versions of dataset amongst which 5th version of the dataset is utilized for the research. Dataset has CC BY 4.0 licence and was collected to fulfil the requirements of Supply Chain for Big Data Analytics, Machine Learning Algorithms, and for areas of important registered activities of Commercial Distribution, Sales Production, Fraud prediction, as such.

3.2 Data Description

Data consists of 180,519 observations and 53 attributes which when analysed in real-time needs to imply Big Data analytics methodology. It has 24 object, 15 float64 and 14 int64 variables, in which later in data preparation and pre-processing some new variables are merged using mathematical calculations and some are dropped which does not have any significant impact on the target variable.

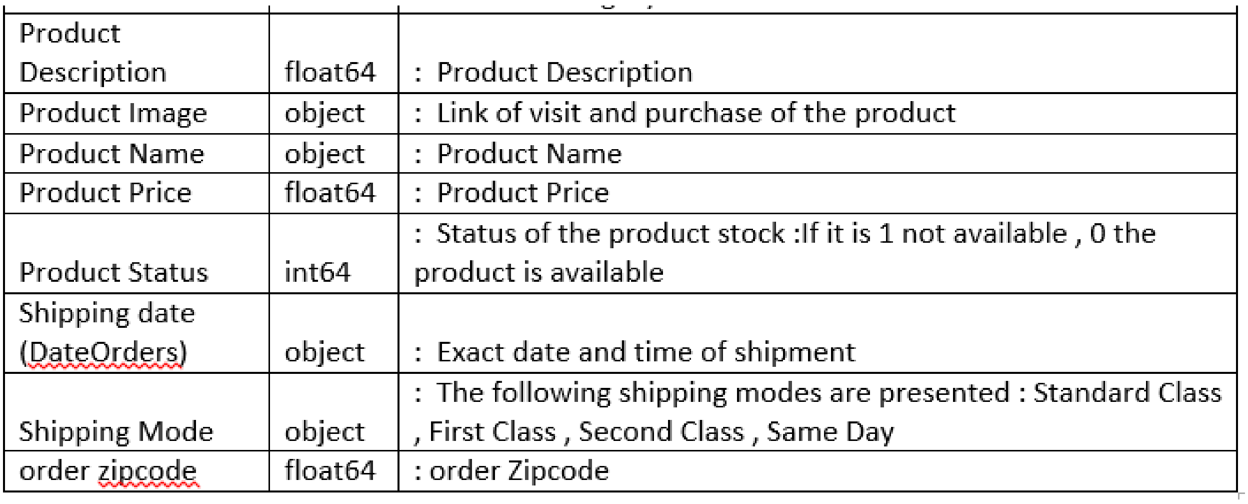


FIELDS	Data Type	DESCRIPTION
Type	object	: Type of transaction made
Days for shipping (real)	int64	: Actual shipping days of the purchased product
Days for shipment (scheduled)	int64	: Days of scheduled delivery of the purchased product
Benefit per order	float64	: Earnings per order placed
Sales per customer	float64	: Total sales per customer made per customer
Delivery Status	object	: Delivery status of orders: Advance shipping , Late delivery , Shipping canceled , Shipping on time
Late_delivery_risk	int64	: Categorical variable that indicates if sending is late (1), it is not late (0).
Category Id	int64	: Product category code
Category Name	object	: Description of the product category
Customer City	object	: City where the customer made the purchase
Customer Country	object	: Country where the customer made the purchase
Customer Email	object	: Customer's email
Customer Fname	object	: Customer name
Customer Id	int64	: Customer ID
Customer Lname	object	: Customer lastname
Customer Password	object	: Masked customer key
Customer Segment	object	: Types of Customers: Consumer , Corporate , Home Office
Customer State	object	: State to which the store where the purchase is registered belongs
Customer Street	object	: Street to which the store where the purchase is registered belongs
Customer Zipcode	float64	: Customer Zipcode
Department Id	int64	: Department code of store
Department Name	object	: Department name of store



Latitude	float64	: Latitude corresponding to location of store
Longitude	float64	: Longitude corresponding to location of store
Market	object	: Market to where the order is delivered : Africa , Europe , LATAM , Pacific Asia , USCA
Order City	object	: Destination city of the order
Order Country	object	: Destination country of the order
Order Customer Id	int64	: Customer order code
order date (DateOrders)	object	: Date on which the order is made
Order Id	int64	: Order code
Order Item Cardprod Id	int64	: Product code generated through the RFID reader
Order Item Discount	float64	: Order item discount value
Order Item Discount Rate	float64	: Order item discount percentage
Order Item Id	int64	: Order item code
Order Item Product Price	float64	: Price of products without discount
Order Item Profit Ratio	float64	: Order Item Profit Ratio
Order Item Quantity	int64	: Number of products per order
Sales	float64	: Value in sales
Order Item Total	float64	: Total amount per order
Order Profit Per Order	float64	: Order Profit Per Order
Order Region	object	: Region of the world where the order is delivered :
Order State	object	: State of the region where the order is delivered
Order Status	object	: Order Status : COMPLETE , PENDING , CLOSED , PENDING_PAYMENT ,CANCELED , PROCESSING ,SUSPECTED_FRAUD ,ON_HOLD ,PAYMENT_REVIEW
Product Card Id	int64	: Product code
Product Category Id	int64	: Product category code

Table 1Data Description



Product Description	float64	: Product Description
Product Image	object	: Link of visit and purchase of the product
Product Name	object	: Product Name
Product Price	float64	: Product Price
Product Status	int64	: Status of the product stock :If it is 1 not available , 0 the product is available
Shipping date (DateOrders)	object	: Exact date and time of shipment
Shipping Mode	object	: The following shipping modes are presented : Standard Class , First Class , Second Class , Same Day
order zipcode	float64	: order Zipcode

3.3 Pre-processing data

As it is famously known in Machine Learning community that Garbage In, Garbage Out (GIGO) hence to get good analysis it is vital to clean the data and make it adaptive for modelling. It is the process of transforming raw data into an interpretable and usable form for ML tasks. This is a process of cleaning the data by removing unnecessary variables which have high co-relationships, to replace the null value present in the samples, to drop some columns which have less valuable descriptive information that will just consume more compute resources and time. Some common characteristics of data pre-processing are to identify missing values, noisy data, incorrect data type, and incomplete data.

The task of handling Missing Data is performed by using the .apply() and .isnull() function from which it is known that 4 variables have missing values: Customer Lname-8, Customer Zipcode-3, Order Zip code-155679 and Product Description-180519. This indicates Order Zip code and Product Description need to be removed as most of its values are missing.

Explicitly 5 new features were created which are 'Total Price', 'fraud', 'Customer Full Name','late_delivery'. These columns are created to minimize the computation time and to provide more accurate and sensible data to the ML models. These columns are created by using mathematical and logical operations.

As the research is divided into two parts, for classification and regression the dataset is also needed to be arranged according to the task. This resulted in dropping some columns which were unnecessary for both the task as they were containing descriptive information.

Longitude and Latitude, Product Description, Product Image, Customer Email and Customer Password, etc. which in total are 12 as commonly expelled columns form dataset used for both the task. For classification with the help of heat map, the features with high co-relation were dropped. Regression task needed some columns to be replaced like ‘late_delivery’ and ‘fraud’ which resulted in dropping its corresponding related column.

Machine Learning models can only handle the numerical format of the data this makes the task of converting categorical values into numeric as a mandatory vital process. Categorical variables are converted into numeric data using LabelEncoder as it will not generate extra dimensions like OneHotEncoder which increases computation time and making the data more complex for the model. After converting all the data into a numeric value, the next step is to separate the target variable from independent variables. For classification ‘fraud’ variable of type, int is set as target variable and other 17 variables as independent features of which 8 feature are actual categorical in nature. In regression ‘sales’ is set as the target variable which is float and has 17 independent features in which 15 are categorical variables in nature.

Once all the manipulations needed for preparing the data is finished it is split into training and test set. Due to a fair number of observations available in data 70% is used for training and 30% is used for the test set instead of 20% this could possibly add more randomness for the model during prediction. It is necessary to split the data to avoid overfitting issue in the models. Once we generate a training set and test set it is important to standardize the independent variables. It is the process of transforming training set in such a way that the average mean is 0 and the Standard Deviation is 1. For classification and regression StandardScaler is implemented to standardize the data.

3.4 Handling Imbalanced Data

Imbalanced data can be justified when the dataset has more instance of a certain class when compared to the frequency of other classes. In the scenario of imbalanced dataset rare instance have less occurrence frequency, so classification rules developed by the model tend to predict less for the rare class. Therefore, this leads to pushing back the importance of rare classes which often have higher importance than contrary case (Sun, Wong and Kamel, 2009).

To solve this issue there are various techniques from which SMOTE (Synthetic Minority Oversampling Technique) is implemented for this research.

SMOTE works on the method of oversampling where it balances the original training set. It simply does not replicate and populate the training set with minority class but introduces synthetic examples. This is done by creating interpolation among several minority class instances which are within the defined neighbourhood. This justifies that technique is focused on “feature space” rather than on the “data space” which means algorithm values feature and their relationship instead of data points in whole(Fernandez et al., 2018).

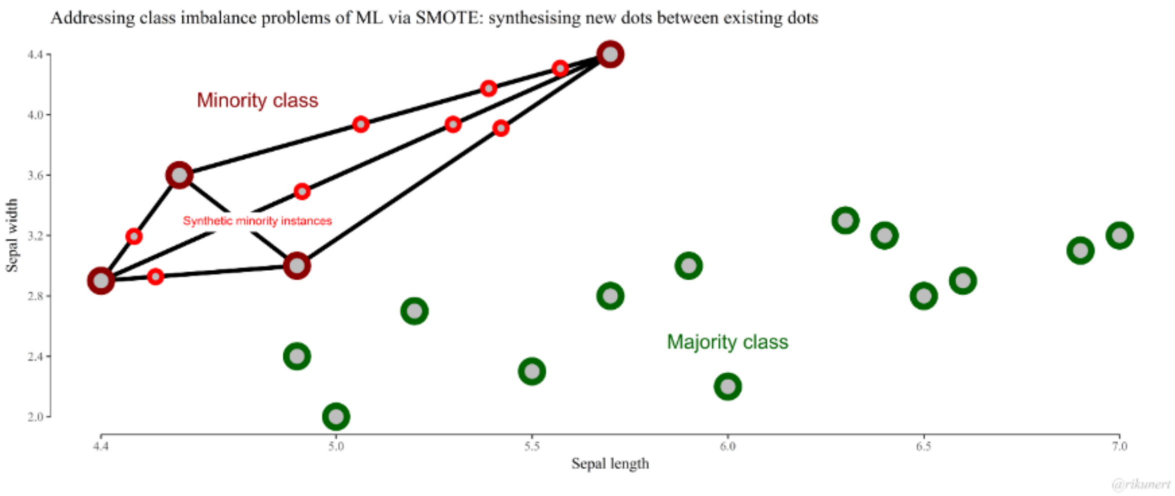


Figure 2 SMOTE Explanation

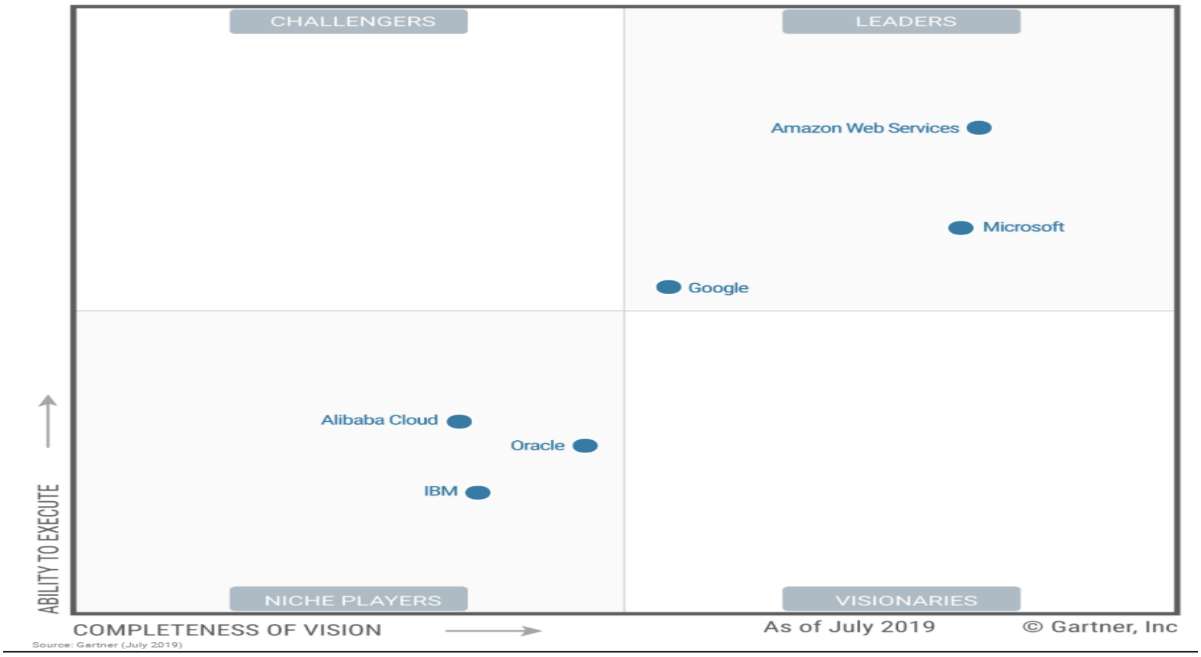
3.4 Methodology

Due to use of Big Data, it is necessary to use technology which can handle such data for analytics within the practical time frame, therefore, several different tools and technologies are assimilated so that core objective can be achieved and the business problem can be solved with the optimum operational expense. Hence, this is the stranding process where several elements are combined around a single axis which is to find the business solution.

3.4.1 AWS (Amazon Web Services):

AWS (What is AWS, 2020) is a platform which provides centralized service for IaaS, PaaS and SaaS it is the world’s most comprehensive and broadly adopted cloud platform. As mentioned in the report by (Gartner Reprint, 2019) where AWS is positioned in the Leaders Quadrant for Cloud Infrastructure as a Service(IaaS). This is achieved because of worldwide spread of its network, offering regions with multiple Avaibality Zone connected with low latency, high throughput, and highly redundant networking. To be specific (Global Infrastructure, 2020) AWS has 77 Avaibality Zones within 24 geographical regions around the globe. It is trusted technology among millions of customers from diverse domains and industries. To name few (Case Studies, 2020) Royal Dutch Shell, Expedia, Verizon, Hyatt, AXA, Nasdaq, Airbnb, Lyft, Coursera, FDZ and many more. This show how well-established AWS is due to the pioneer in Cloud Computing technology. This indicates that AWS has the potential to solve major business problem in Supply Chain Industry concerned to its electronic data generated in massive amount

Figure 1. Magic Quadrant for Cloud Infrastructure as a Service, Worldwide



Source: Gartner (July 2019)

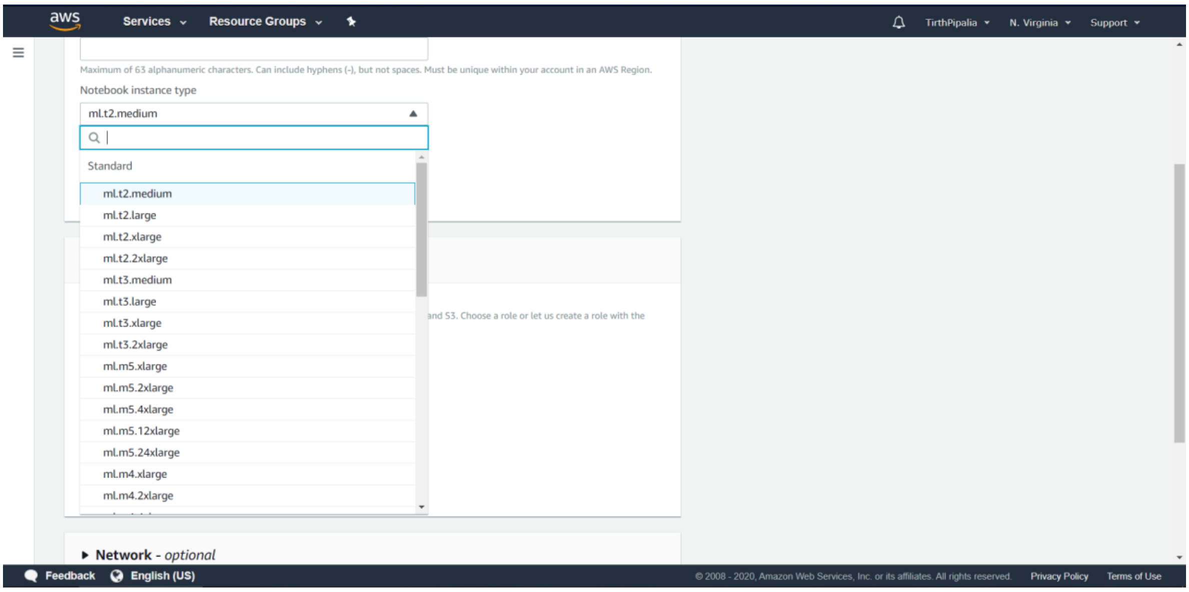
Figure 3 Best cloud provider

3.4.2 Amazon EC2

It provides scalable, secure, resizable computing capacity service in AWS designed to make web-scaling process effortless with less friction. As it provides IT infrastructure on using cloud computing there is no need for upfront investment in hardware which accelerates developing and deploying phase of an application. It provides complete control over security configuration, networking, and storage management. When used with auto-scaling and Elastic Load Balancer it helps to make fault-tolerant applications (Barr, Narin and Varia, 2011). Some impressing statistics according to (Amazon EC2, 2020) is, it has more than 300 virtual instance for every business need, 7x of fewer downtime hours compared to the next largest cloud provider. As there is a total of 8 different families' of instances in which each instance type provides a choice of CPU size, storage, GPU.

3.4.3 Amazon SageMaker

In the era of automation, it is necessary to develop a process which supports or uses tools and techniques which requires less human interaction so that more productivity is generated. This can be done in the field of Machin Learning by using SageMaker service from AWS (*Amazon SageMaker*, 2017). It is a fully managed service for the ML task where common ML algorithms can be optimized to run efficiently against extremely huge data even if it is a distributed environment. It provides an inbuilt feature called Notebook Instance which allows user to create Jupyter Notebook without any need for installing other dependencies. It also makes sure that basic essential packages and kernels are loaded in the Jupyter Notebook. It only charges for the time it was in use for computation. It provides options from several EC2 families on which Notebook should be launched according to the requirement. Once the configuration and setup process is finished we just have to activate the instance and launch Jupyter Notebook or Jupyter Lab. (Karnin Zohar et al, 2018) explains how cost-effective and time-saving it is compared to present process of implementing ML in finding business solutions.



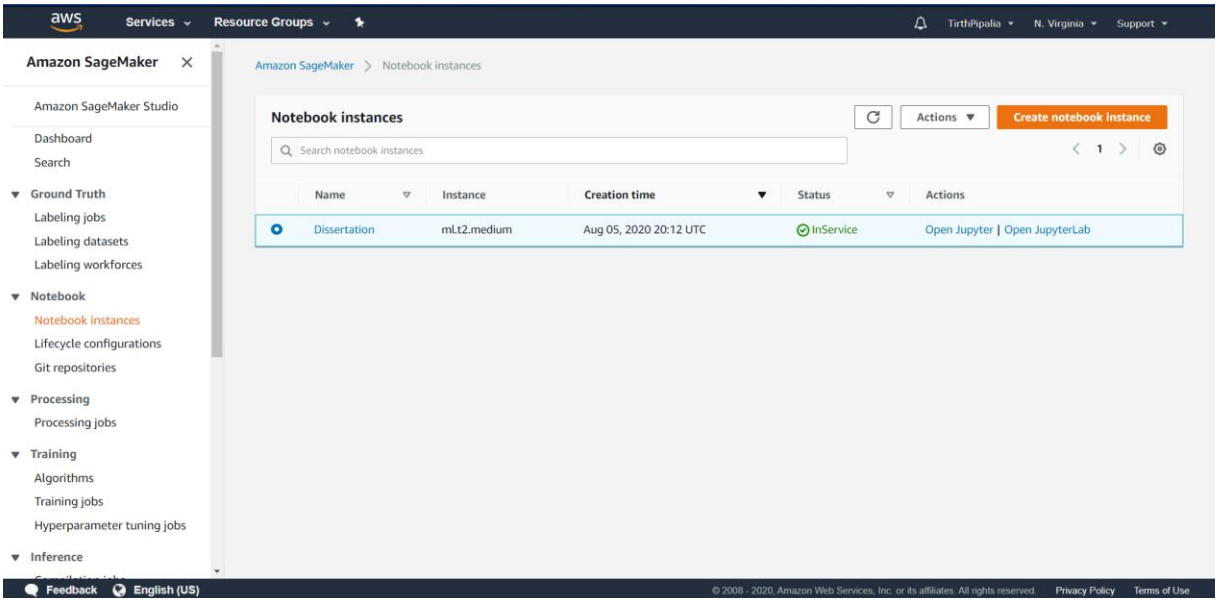
Maximum of 63 alphanumeric characters. Can include hyphens (-), but not spaces. Must be unique within your account in an AWS Region.

Notebook instance type: ml.t2.medium

and S3. Choose a role or let us create a role with the

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

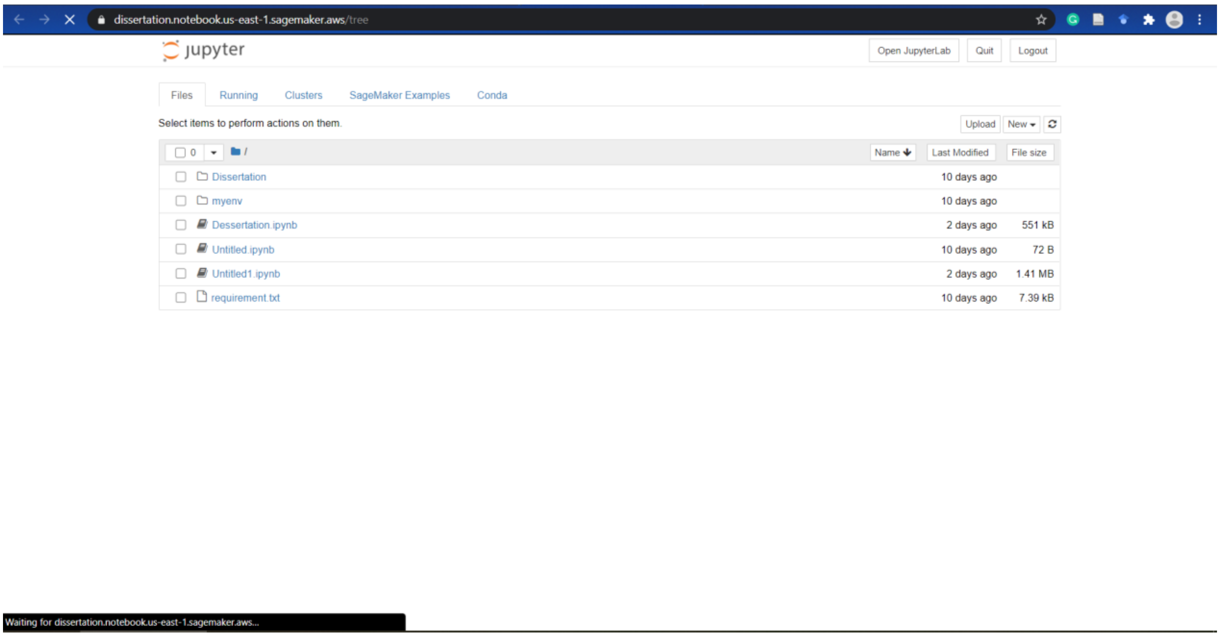
Figure 4 Instance selection SageMaker



Notebook instances

Name	Instance	Creation time	Status	Actions
Dissertation	ml.t2.medium	Aug 05, 2020 20:12 UTC	InService	Open Jupyter | Open JupyterLab

Figure 5 Notebook Instance Creation


dissertation.notebook.us-east-1.sagemaker.aws/tree
jupyter
Open JupyterLab
Quit
Logout
Files
Running
Clusters
SageMaker Examples
Conda
Upload
New
0
/
Waiting for dissertation.notebook.us-east-1.sagemaker.aws...

Select items to perform actions on them.

Name	Last Modified	File size
Dissertation	10 days ago	
myenv	10 days ago	
Dessertation.ipynb	2 days ago	551 kB
Untitled.ipynb	10 days ago	72 B
Untitled1.ipynb	2 days ago	1.41 MB
requirement.txt	10 days ago	7.39 kB

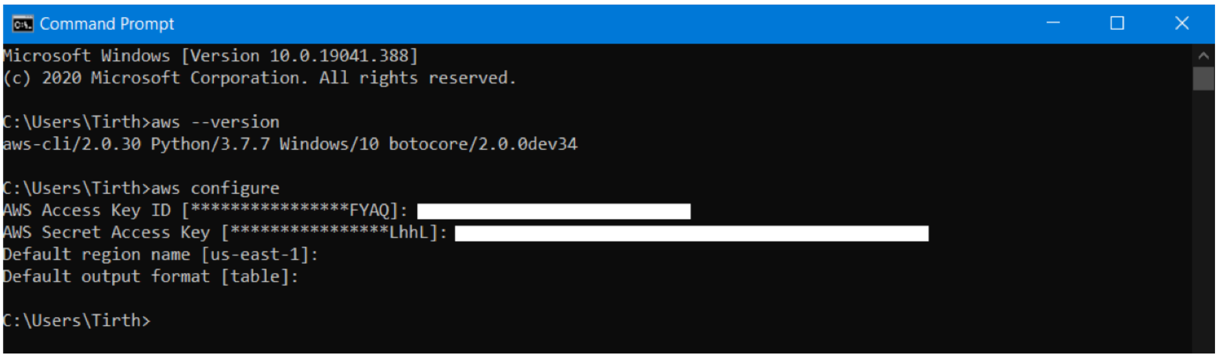
Figure 6 Launched Notebook Instance SageMaker

3.4.4 Amazon S3 and Jupyter Notebook Configuration

In faster-growing market due to globalization and technology advancements like Internet of Things, AI, Deep Learning, 5G, etc. immense quantity of structured and unstructured data is generated for each industry ranging from fish farming to commercial space agency. To cope up with this growth-rate of data and to benefit the business, Amazon

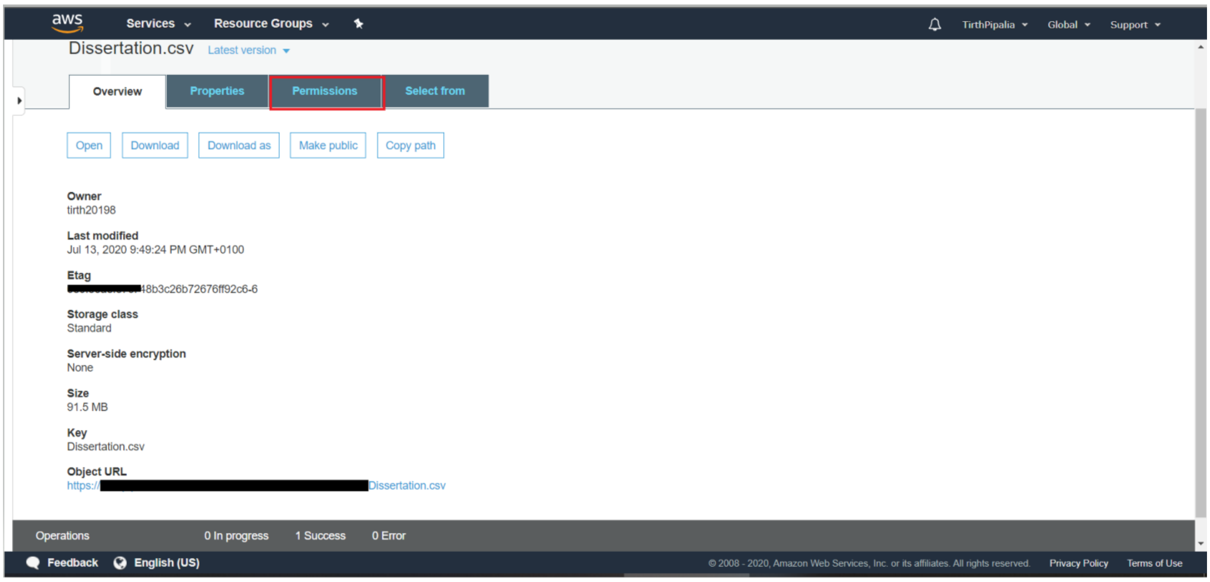
S3 is the cornerstone to be implemented in the Business Strategy. As discussed in section 2 the benefits of using Amazon S3 one of the services of AWS makes implementation and execution process simple and agile.

To integrate S3 service with python which will help to access the object stored on S3 we need to install 4 packages in Jupyter Notebook i.e. boto3, sys, os and open_smart. Once the object is uploaded in S3 to closest client region in this case Ireland. We need to change the permission of objects which by default is set as private. Then we need to configure AWS so that we do not have to add AWS Access Key ID and AWS Secret Access Key in our application code which allows implementing good security practices recommended by AWS. Once the configuration steps are completed, we just need to write a python script as shown in Figure 10. This will directly allow accessing the content of S3 bucket using a python programming language.



```
Microsoft Windows [Version 10.0.19041.388]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Tirth>aws --version
aws-cli/2.0.30 Python/3.7.7 Windows/10 botocore/2.0.0dev34

C:\Users\Tirth>aws configure
AWS Access Key ID [****************FYAQ]: [redacted]
AWS Secret Access Key [****************LhhL]: [redacted]
Default region name [us-east-1]:
Default output format [table]:

C:\Users\Tirth>
```

Figure 7 Configuring S3 permissions



Dissertation.csv Latest version

Owner
tirth20198

Last modified
Jul 13, 2020 9:49:24 PM GMT+0100

Etag
[redacted]48b3c26b72676ff92c6-6

Storage class
Standard

Server-side encryption
None

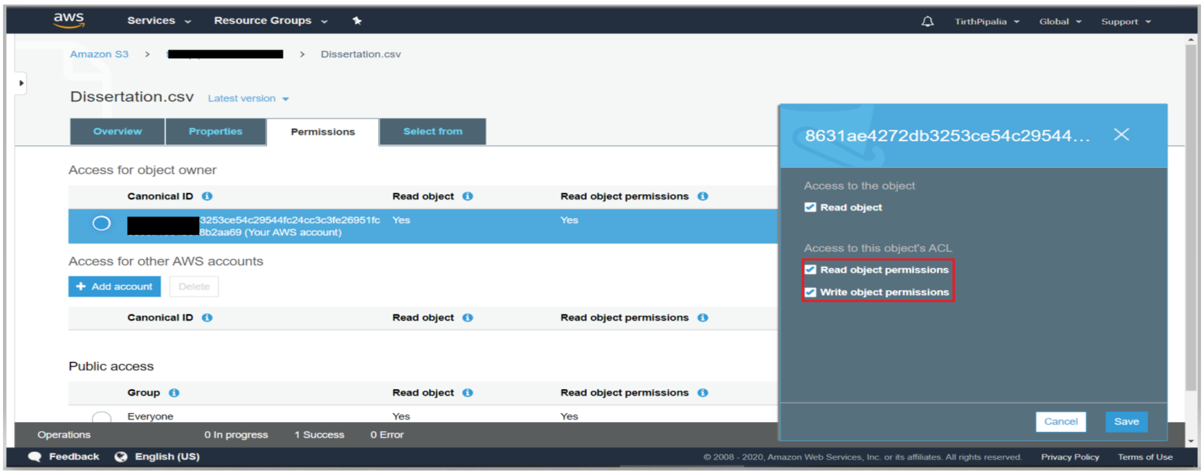
Size
91.5 MB

Key
Dissertation.csv

Object URL
https://[redacted]Dissertation.csv

Operations 0 In progress 1 Success 0 Error

Figure 8 Changing S3 object permission to Access



Amazon S3 > [redacted] > Dissertation.csv

Dissertation.csv Latest version

Access for object owner

	Canonical ID	Read object	Read object permissions
○	[redacted]3253ce54c29544fc24cc3c3fe26951fc 8b2aa69 (Your AWS account)	Yes	Yes

Access for other AWS accounts

Canonical ID	Read object	Read object permissions

Public access

	Group	Read object	Read object permissions
○	Everyone	Yes	Yes

Operations 0 In progress 1 Success 0 Error

8631ae4272db3253ce54c29544...

Access to the object

☑ Read object

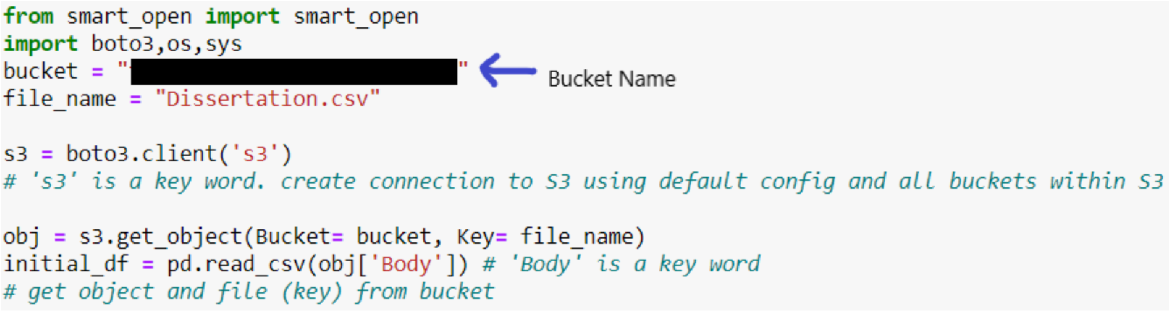
Access to this object's ACL

☑ Read object permissions

☑ Write object permissions

© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Figure 9 Enabled S3 object for Read and Write



```
from smart_open import smart_open
import boto3,os,sys
bucket = "[redacted]"
file_name = "Dissertation.csv"

s3 = boto3.client('s3')
# 's3' is a key word. create connection to S3 using default config and all buckets within S3

obj = s3.get_object(Bucket= bucket, Key= file_name)
initial_df = pd.read_csv(obj['Body']) # 'Body' is a key word
# get object and file (key) from bucket
```

Figure 10 Python code to access S3 object

3.5 Evaluation Metrics

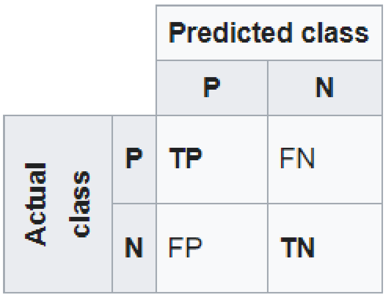
It is implemented for the qualitative measurement of the performance of machine learning models. These metrics are as a standard to measure the ML models performance. It is also essential to implement multiple evaluation metrics for single model because there can be a possibility that model shows satisfactory results for any individual metric. Therefore the output is essential to be compared amongst different metrics (M and M.N, 2015). As it is explained by (Srivastava, 2019) building a Machine Learning model works on constructive feedback principle where the model is developed and implemented. The result of the implementation is achieved and analysed using metrics. If the result matches the desired objective, model is selected or again goes for the process of tuning the model or trying a different model. There are several evaluation metrics and several ML models, so it becomes crucial to identify an appropriate metric to validate each model (Sammut and Webb, 2010).

Therefore, several common classification and regression metrics are used to evaluate the result. To avoid question and confusion like: How do I calculate accuracy for my regression problem? It is important to understand the definition of classification and regression problem.

3.5.1 Classification Problem

It is considered when values to be predicted are discrete/binary. Where the probability of predicted answer is limited to True or False. E.g. Delivery of product is successful or unsuccessful.

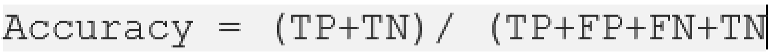
a) *Confusion Matrix*: Also, well known as accuracy paradox which truly goes by its name for confusing the decision-making process. Whenever there is a discrete outcome of the problem it is usually summarised using confusion matrix (Fatourechi et al., 2008). Later various evaluation metrics which are more specific to the task evaluates this confusion matrix which helps to identify and analyse the best model for a business solution. It is the tabular representation between the actual test data values and predicted values. There are 4 cells: TP=True Positive where both actual value and the predicted value is true, FP=False Positive where models predict as the true but actual value is false also known as Type 1 error, TN=True Negative where both predicted value and the actual value is false, FN=False Negative where the model predicts false and actual value is true also known as Type 2 error. Based on these different metrics are measured for the classification problem.



		Predicted class	
		P	**N**
Actual class	**P**	**TP**	FN
	N	FP	**TN**

Figure 11 Confusion Matrix

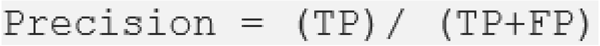
b) *Accuracy:* Based on the confusion matrix utmost ordinarily employed metric for classification is accuracy. The defined formula is the ratio of correctly predicted instances, also identified as a summation of diagonal elements of the confusion matrix. When a dataset is well balanced and not skewed it is the most optimum choice of metric.



$$\text{Accuracy} = (TP+TN)/\ (TP+FP+FN+TN$$

Equation 1 Accuracy

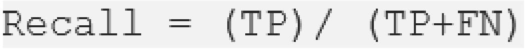
c) *Precision:* (Positive Predictive Value): It is used when we need to find the proportion of predicted true positives by model. It is a better option to implement when a business problem is to find the precise number of instance predicted by the model being correct.



$$\text{Precision} = (TP)/\ (TP+FP)$$

Equation 2 Precision

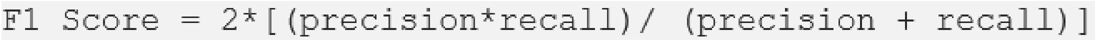
d) *Recall:* It is important when the stake for false negative is high and the consequences for false negative is devastating for business. This helps to capture the utmost True Positive as possible. Therefore, being a measure for information retrieval performance.



$$\text{Recall} = (TP)/\ (TP+FN)$$

Equation 3 Recall

e) *F1-Score:* When the business problem is to achieve both good recall and precision score this metric is unsurpassed choice to aid. It is defined as a harmonic mean of Recall and Precision. Even if it lies between Recall and Precision being closer to the smaller of these values. Therefore, a model with higher F1 has both good Recall and Precision score.



$$\text{F1 Score} = 2*[(\text{precision}*\text{recall})/\ (\text{precision} + \text{recall})]$$

Equation 4 F1-Score

f) *AUC_ROC Score:* It helps to find Area Under the Receiver Operating Characteristic Curve with the help of predicted score. This metric is used for binary classification problem. It measures the ability of the classifier to distinguish between the classes and implemented for summary of ROC curve. Higher the AUC value better the classifier model.

3.5.2 Regression Problem

It is considered when values to be predicted are continuous and models predict quantity. Hence the probability of prediction is reported as an error. E.g. Weather forecasting.

a) *Mean Squared Error (MSE):* It helps to identify the distance between the regression line and the set of points. It works by squaring this distance also known as errors, squaring is important so that negative values do not have separate weightage. Then an average is calculated for this measured set of errors. Which simply means prediction error as it is measured by calculating the difference between the predicted value and true value of an observation. As it relies on the differentiable methodology it can be optimized better hence making it a preferable choice.

$$MSE = \frac{1}{n} \Sigma \underbrace{\left(y - \widehat{y} \right)^2}_{\text{The square of the difference between actual and predicted}}$$

Equation 5 Mean Squared Error

b) *Root Mean Squared Error (RMSE):* One of the recognised and widely used metric for regression which is the square root of averaged squared distance between predicted value by model and actual target variable value in the dataset. In this metric errors are squared before calculating the average which can possess high penalty for large errors. This helps to indicate how concentrated data points are around the regression line. RMSE value should be lower for the better fit model.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

Equation 6 Root Mean Squared Error

c) *Mean Absolute Error (MAE):* One of the simplest metric to follow through. In this error (residue) is calculated for each data point by explicitly taking absolute values only which will remove negative values. Average of collected residues are considered as MAE. As this metric uses absolute values only it does not indicate over performance or underperformance of the model. Value of MAE closer to 0 indicates a better fit of the model.

$$MAE = \overset{\text{Divide by the total number of data points}}{\boxed{\frac{1}{n}}} \underset{\text{Sum of}}{\Sigma} \underbrace{\left| \overset{\text{Actual output value}}{\boxed{y}} - \overset{\text{Predicted output value}}{\boxed{\widehat{y}}} \right|}_{\text{The absolute value of the residual}}$$

Equation 7 Mean Absolute Error

3.6 Algorithms

3.6.1 Decision Tree:

One of the most approachable and distinct techniques for solving a classification problem. (Rokach and Maimon, 2005)As per its design and working it can be implemented for the decision-making process in several sectors like Machin Learning, Statistics, Medicine, and Visualization. It is a method of breaking down of the complex decision process into simple collective singular decision nodes which makes it easier to interpret and understand the flow of events (Safavian and Landgrebe, 1991). It starts with the root node which does not have any incoming edges. The node with no outgoing edge is called a leaf or terminal nodes also referred to as decision node because of this the final decision is obtained. All the nodes except the root node has exactly one incoming edge associated and except the leaf node, each node has two outgoing nodes also known as a test or internal node.

3.6.2 Light GBM:

It is an upgraded version of GBDT(Gradient Boosting Decision Tree) loaded with EFB(Exclusive Feature Bundling) and GOSS(Gradient-based One-Side Sampling) (Ke et al., 2017). It is a newly introduced model for Machine Learning but spreading it’s significance like wildfire as it reduces the computation speed when compared with conventional GBDT and

XGBOOST (Khandelwal, 2017). It was developed under consideration to improve the efficiency and accuracy of the GBDT model to handle Big Data. This is done by a straightforward approach of reducing the number of data instances and data features. It provides 3 sets of hyper parameters for tuning: 1) for best fit 2) for faster speed and 3) for better accuracy and several other important parameters to tune the model.

3.6.3 Linear Discriminant Analysis:

It is a combination of several methods from various algorithms like Analysis of variance (ANOVA), Principle Component Analysis (PCA), Linear Classifier, Regression Analysis. This model uses three steps to achieve the result (Tharwat et al., 2017): 1) To calculate the distance between separate classes which is also known as the between-class matrix or between-class variance. 2) Then calculates the distance between samples of each class and its mean also known as the within-class matrix or within-class variance. 3) Finally, it constructs lower-dimension space which minimises within-class variance and maximizes between-class variance. The best example of LDA implementation for classification comparison is shown in (Ghosh and Shuvo, 2019).

3.6.4 Random Forest:

It is one of the techniques of ensemble learning which is a hybrid combination of: bagging, random subspace method and decision trees for the core classifiers. (Biau, 2012) discuses about substantial gains for regression and classification when implementing ensembles of decision trees. And this tree is grown based on the random parameters assigned. It initially selects random samples from the datasets and then constructs a decision tree for each sample. The prediction result from each tree is then evaluated through voting from all the predicted result. Finally, the sample with the most voted predicted result is selected as the final perdition result. Trees generated are not subjected to pruning and enabling it to partially over fit the training dataset. Predefined random subset further diversifies each classifier by restricting the decision of which feature to split in the tree.

3.6.5 Ridge Regression: This is the method of ill-posed problems in regularization used to mitigate the problem of multicollinearity predominantly experienced in models having several numbers of parameters. When the parameters have a small effect, RR model performs well and prevents exhibiting high variance (Ogutu, Schulz-Streeck and Piepho, 2012). It does not force co-efficient to vanish therefore cannot choose the model which only has the most relevant and predictive subset of predictors. It reduces standard error by adding a degree of bias to the regression estimates (‘Ridge Regression’, 2020). This method helps in reducing near-linear relationships amongst independent variables in a dataset.

$$\hat{\boldsymbol{\beta}}\ (\mathrm{ridge}) = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\ \|\boldsymbol{\beta}\|_2^2$$

Equation 8 Ridge regressor

3.6.6 LASSO Regression: It is an abbreviation for Least Absolute Shrinkage and Selection Operator proposed by (Tibshirani, 1994) which performs both regularization and variable selection which also reveals there is no need for unique coefficient estimators if covariates are collinear. It shrinks some coefficients and sets them as zero which leads to retaining good features of ridge regression and subset selection. LASSO uses L1 regularization which accepts several co-efficient nearer to Zero and having small subset with larger co-efficient.

$$\hat{\boldsymbol{\beta}}\ (lasso) = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_1,$$

Equation 9 LASSO regressor

3.6.7 Linear Regression: One of the basic and widely implemented instance of regression which has a single explanatory variable. It helps to identify 2 core things from the dataset, 1) which set of variables plays a prominent role in improving the prediction performance? 2) Which variables/features are significant for a developed model? Linear is the name function of regressor with single predictor. This single predictor is selected based on the accuracy measured by its squared residual. The predictor whose squared residual is least is considered as best variable.

3.6.8 XGBoost:

It is one of the ensemble learning method designed to be highly portable, flexible, and efficient. It is the method which has been implemented in most of the ML Hackathon and most winning method on Kaggle (Chen and Guestrin, 2016). It grows fixed size decision tree sequentially like AdaBoost however these trees are larger than stumps. It is a better option than Gradient Boosting as it also combines regularization with it.

3.6.9 AdaBoost:

It is an ensemble learning technique which was developed for solving two problems: how to combine weak classifiers and how to adjust the training set in order to enable weak classifiers to conduct a training (Chengsheng, Huacheng and Bing, 2017). It has an upper hand when comparing speed and ease of operation which makes it easy to program. It is meta classifier which begins its process by fitting a classifier on a dataset and replicates this while providing weights for incorrectly classified instance(Pedregosa et al., 2011). This process is iterated until it completely fits the training dataset where there is no error, or it reaches the limit of maximum number of estimators. This adjustment helps in focusing on more difficult cases.

3.6.10 Gradient Boosted Tree:

It is another model which uses a boosting method in which weak learners are converted into strong learners where each new tree is introduced to a modified version of the original dataset. As it abides with boosting mechanism, it trains several base models in a sequential manner and increasing the number gradually and additively. It then identifies the shortcomings using loss function. The loss function is measured to indicate the performance of the model's coefficients on underlying data. For regression problem at each stage, the regression tree is fitted which provides a negative gradient of the developed loss function(Korolev and Ruegg, 2015).

4. Implementation

As the data collected is from several IoT devices and it is real-time data at the time of storage and whenever the entire system is running. Due to Big Data avaibality in Supply Chain Industry, it is the best industry practice to upload the data on cloud where it can be easily cleaned and maintained by AWS. Due to cloud-based technology, there is no need for extending and managing hardware storage and even better features and options for data management is provided by AWS itself. This aids in channelling spared time for uplifting the ML model and to achieve the desired outcome at a much faster rate. For this same reason, data is uploaded on S3 bucket which is universally callable object-based storage system service provided by AWS. Once the data is fetched from S3 bucket ML engineering and Data preparation can be commenced.

4.1 Setting up the environment

This section describes essential software and hardware tools and machine used for this research. Starting with the hardware AWS EC2 instance was used for computation on the cloud which was deployed and managed by SageMaker service of AWS. In which Jupyter instance was used which is pre-configured with basic tools and fully managed by AWS. There were several packages and libraries which were needed to be installed for analysis and machine learning modelling. To name a few XGBoost, lightgbm, time, Plotly, matplotlib, seaborn. Python 3 was used for the coding purpose as it has numerous libraries which support ML and DA also packages for accessing AWS S3 service were installed like boto3. No third-party application like H2O, Tableau was used for ML and EDA task. Minimum system requirement for this experiment is 2GB RAM, the 2.4GHz processor. Several other dependencies are installed and mentioned in the code file.

4.2 Fetching Data from AWS S3

To fetch data from S3 we need to pre-configure some permissions and security patches which is explained in Section 3.3.5. Once this configuration is completed with the help of boto3 which is SDK (Software Development Kit) for python language from AWS. It helps to connect, create, and manage various AWS service through programming without any need for login into AWS Management Console. It contains low-level access and object-oriented API for AWS services (boto3: The AWS SDK for Python, 2014). Another useful library smart_open is used which helps to stream large files from cloud storage like S3, Azure blob, GCS, SFTP, HTTPS, HTTP, or from the local file system. Which supports on-the-fly(de-) compression and transparent for several formats (Rehurek, 2015). The necessity to use smart_open is because when working with boto or boto3 their own method only works well for files of small-medium size as it loads the RAM completely without streaming. Smart_open shields from gotchas and boilerplate when dealing with large files by offering clean unified Pythonic API which results to write less code and aim for a lower number of bugs.

4.3 Exploratory Data Analysis

Heat map: It is one of the essential graphs which shows co-relation between different variables and helps to drop the feature with similar data values. Hence helping in reducing the computational time and improving the accuracy of the model because it is trained on values which are mostly unique and not repetitive. It is used as a tool for risk assessment (McKay, 2012) which highlights the features that are of no use for prediction and are just descriptive or co-relating data values.

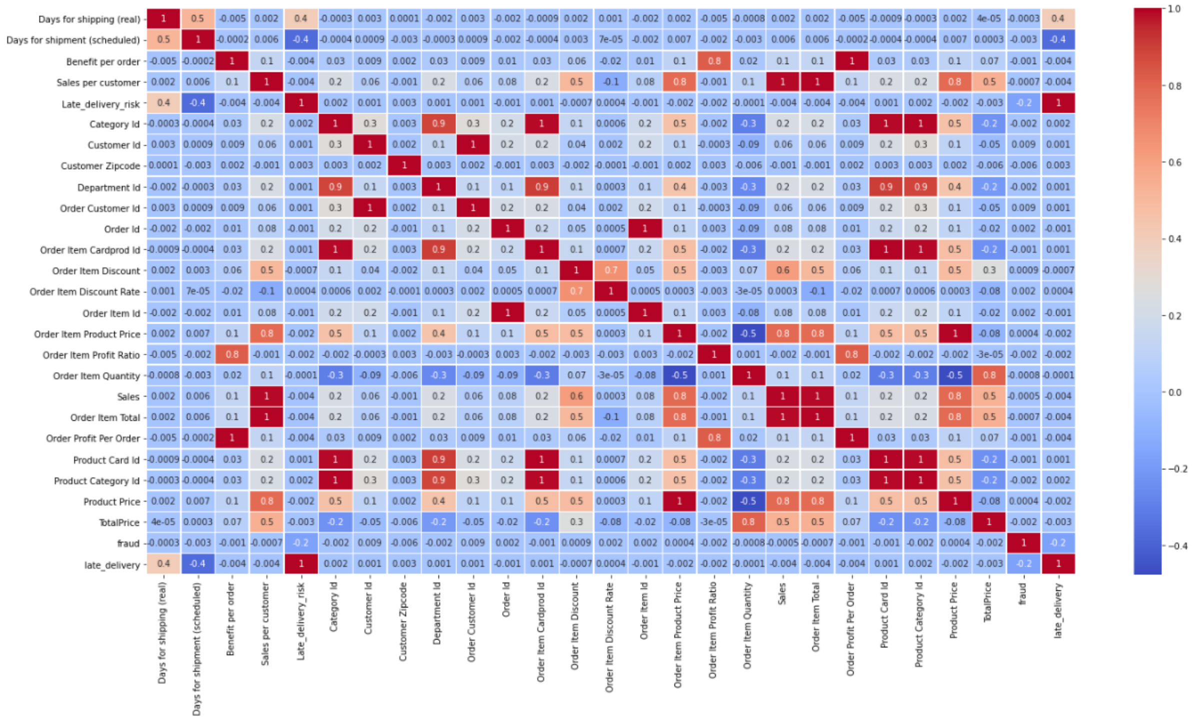


Figure 12 Heat Map with all the features of Dataset

Different Types of Payments used in All Regions (Figure 13): These visualization highlights Debit transaction is the most used method in all the region and Cash is the least preferred method for payment which increases the chances of committing fraud, which can be due to loopholes in the online payment system.

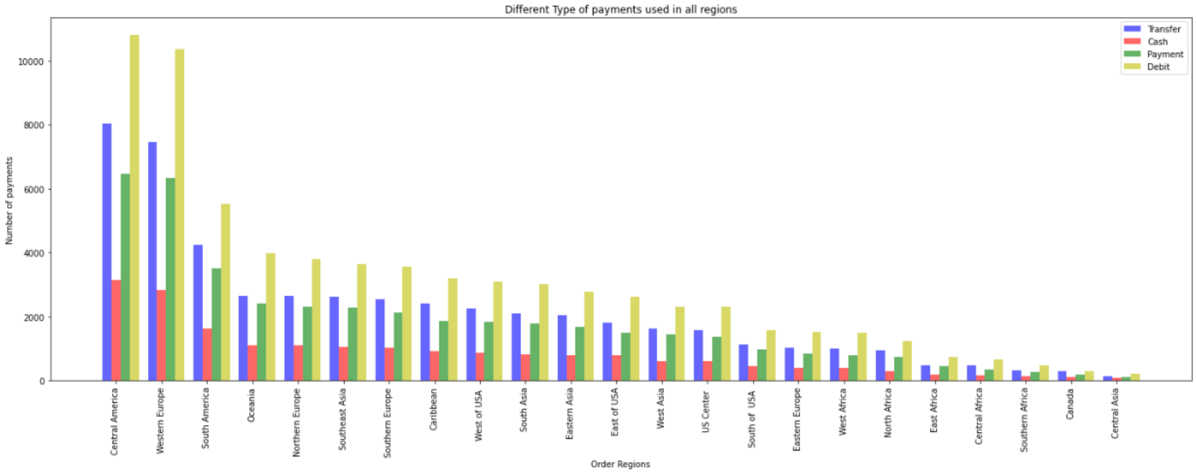


Figure 13 Most used method for transaction

The Region with the Highest Fraud (Figure 14): Pie Chart helps to indicate that Western Europe is the region with the highest number of fraud followed by Central America and South America. It helps by indicating that the focus for taking some measures in this area should be a priority rather than starting from regions with least fraud.

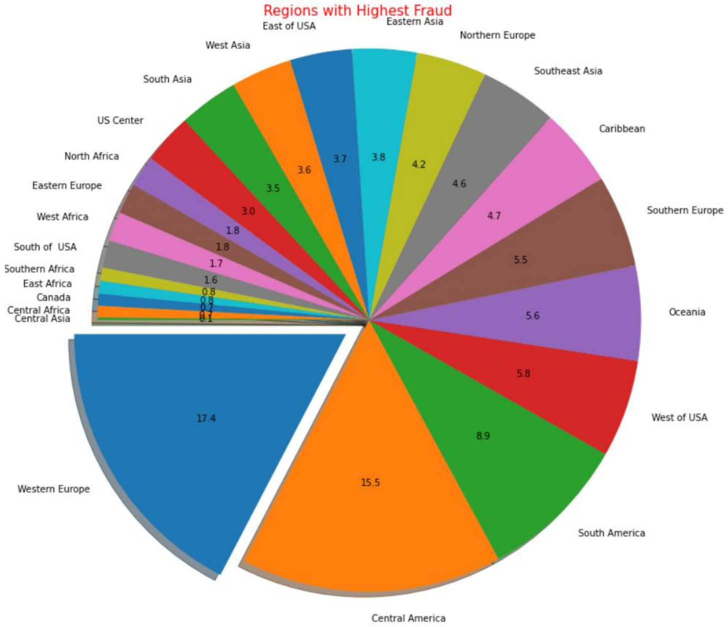


Figure 14 Region with maximum fraud

Total Sales for all regions (Figure 15): This plot shows the sales distribution for all the region s which clearly shows that Western Europe, Central America, and South America as the top 3 region for business.

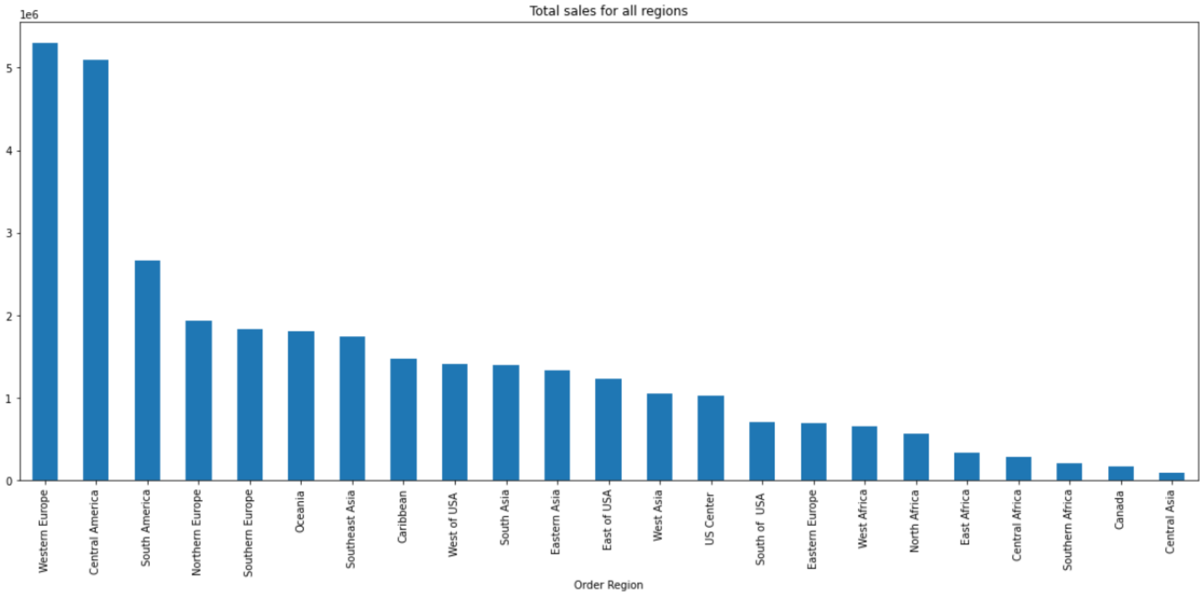


Figure 15 Total Sales of each region

The Region with most Loss(figure 16) : Indicating that Central America which has higher frau d frequency leads to the highest loss in revenue also showing a direct relationship between f raud and loss in that region followed by Western Europe and South America. Total loss of re venue is **-3883547.345768667**

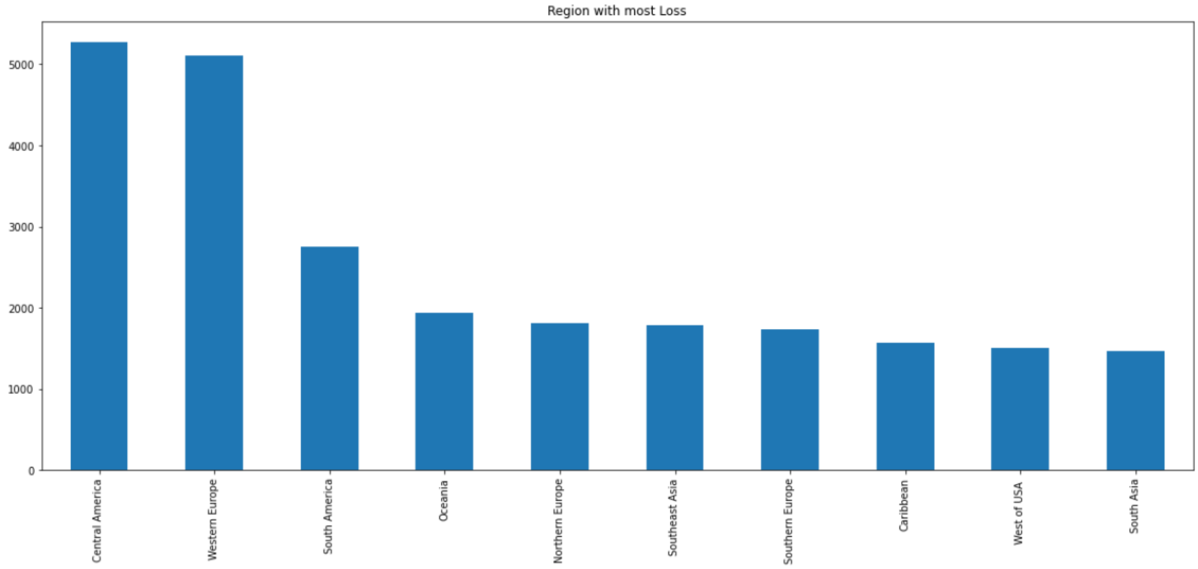


Figure 16 Region with total loss in revenue

4.4 Feature Engineering

It is an application of domain knowledge which will help in selecting essential features as a base to build a model upon it. This process is fundamental which can be expensive and difficult to implement (Zaidi, 2015). With the help of preliminary exploratory analysis, it was revealed that the target variable has several other classes which cannot be useful. Therefore, data related to the suspected fraud and fraud was only selected using feature engineering and several columns which were created and added in the dataset as explained earlier is also an example of feature engineering. A heat map is also used for feature selection and it is significant for feature engineering as explained earlier. Once the classification and regression model are fitted feature_importance and coef_ importance for each model is generated. This value helps in knowing which variables are essential for the model and what are the key features of the best performing model. All these insights will help in the decision-making process when some measures are needed to be taken.

4.5 Implementing SMOTE on imbalanced Data

Below figure shows 123540 values for classes Not Fraudulent and 2823 values for Fraud whic h is minority class here.

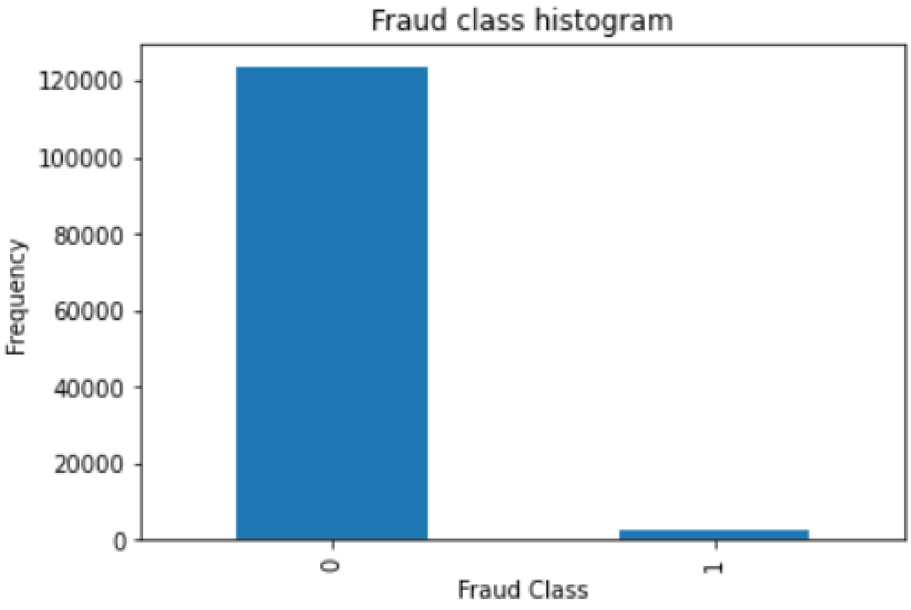


Figure 17 SMOTE unbalanced classes

Below figure show balanced training set after implementing SMOTE where both Non-Fraudulent and Fraudulent transaction class has same frequency value i.e. 123540.

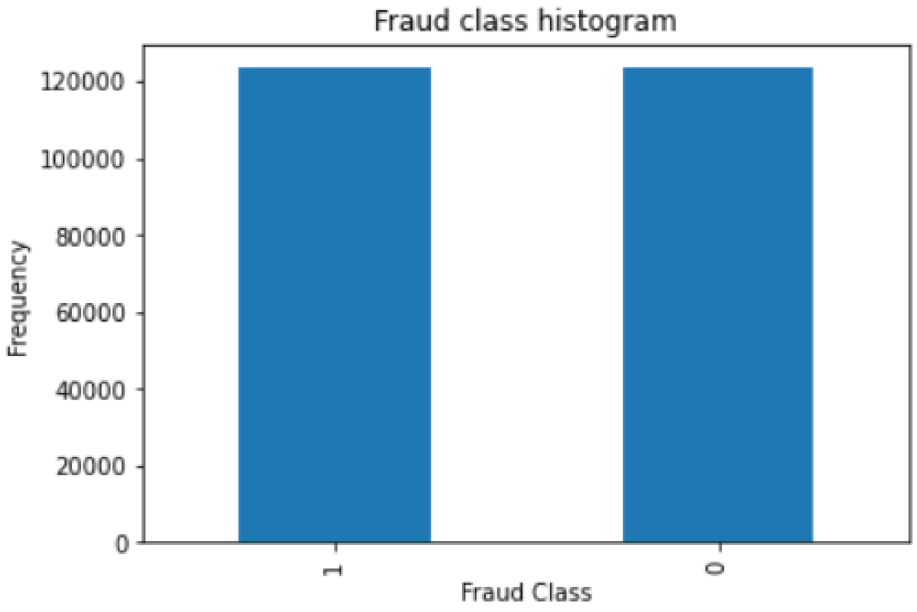
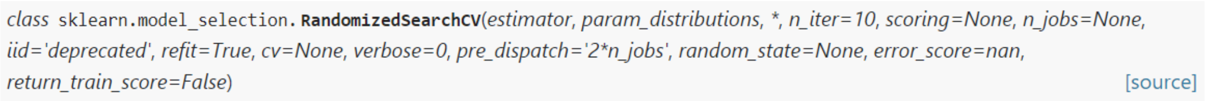


Figure 18 SMOTE balanced classes

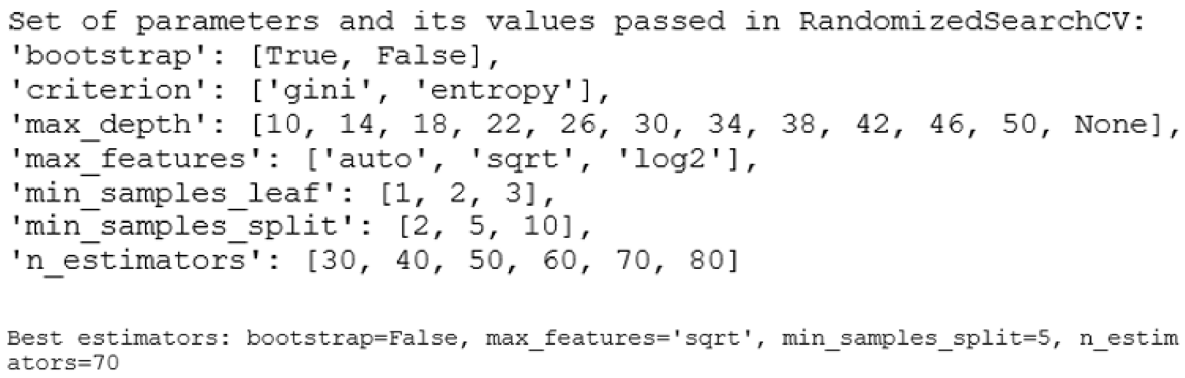
4.6 Model Creation

4.6.1 Random Forest



```
class sklearn.model_selection.RandomizedSearchCV(estimator, param_distributions, *, n_iter=10, scoring=None, n_jobs=None,
iid='deprecated', refit=True, cv=None, verbose=0, pre_dispatch='2*n_jobs', random_state=None, error_score=nan,
return_train_score=False)                                                                                        [source]
```

RandomForest has various Parameters and Attributes which helps to achieve and tune the model according to the requirement. Tuning these parameters is also called as hyper-parameters tuning which is performed using RamdomizedSearchCV. It helps to search the best combination of parameters for models that will best fit the data.



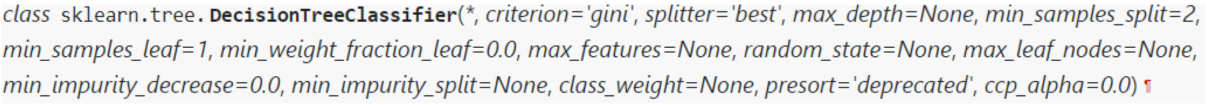
```
Set of parameters and its values passed in RandomizedSearchCV:
'bootstrap': [True, False],
'criterion': ['gini', 'entropy'],
'max_depth': [10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, None],
'max_features': ['auto', 'sqrt', 'log2'],
'min_samples_leaf': [1, 2, 3],
'min_samples_split': [2, 5, 10],
'n_estimators': [30, 40, 50, 60, 70, 80]
```

```
Best estimators: bootstrap=False, max_features='sqrt', min_samples_split=5, n_estim
ators=70
```

Figure 19 Random Forest best parameters selection

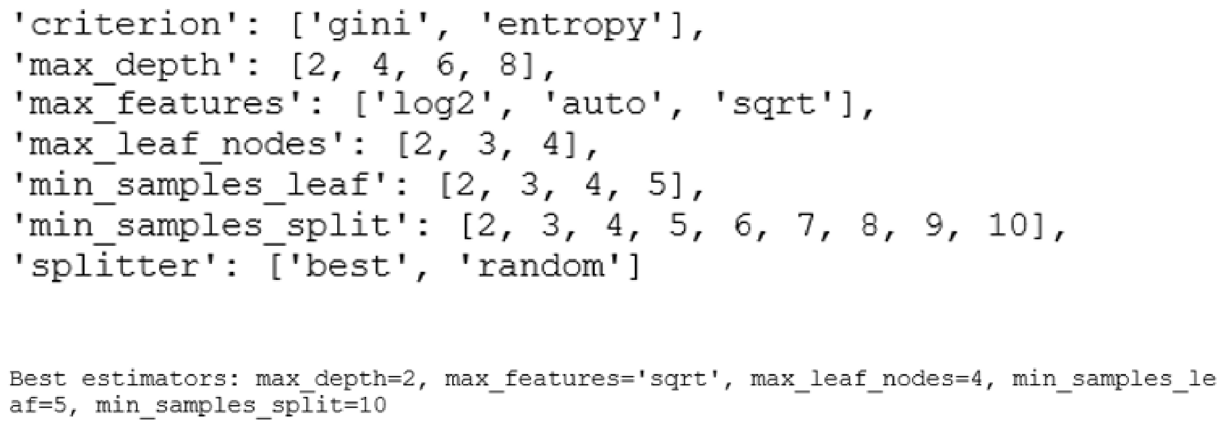
Obtained best estimators are applied to Random Forest classifier with best score=0.9926542010684799 and it took 20.6 minutes to finish the 3-fold Cross-Validation of 50 candidates totalling to 150 fits of combination to compute.

4.6.2 Decision Tree



```
class sklearn.tree.DecisionTreeClassifier(*, criterion='gini', splitter='best', max_depth=None, min_samples_split=2,
min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, random_state=None, max_leaf_nodes=None,
min_impurity_decrease=0.0, min_impurity_split=None, class_weight=None, presort='deprecated', ccp_alpha=0.0) ¶
```

Above are the few parameters and attributes of decision tree classifier, the best fit of parameters is checked using RandomizedSearchCV as shown below are the set of combination as input for RandomizedSearchCV



```
'criterion': ['gini', 'entropy'],
'max_depth': [2, 4, 6, 8],
'max_features': ['log2', 'auto', 'sqrt'],
'max_leaf_nodes': [2, 3, 4],
'min_samples_leaf': [2, 3, 4, 5],
'min_samples_split': [2, 3, 4, 5, 6, 7, 8, 9, 10],
'splitter': ['best', 'random']
```

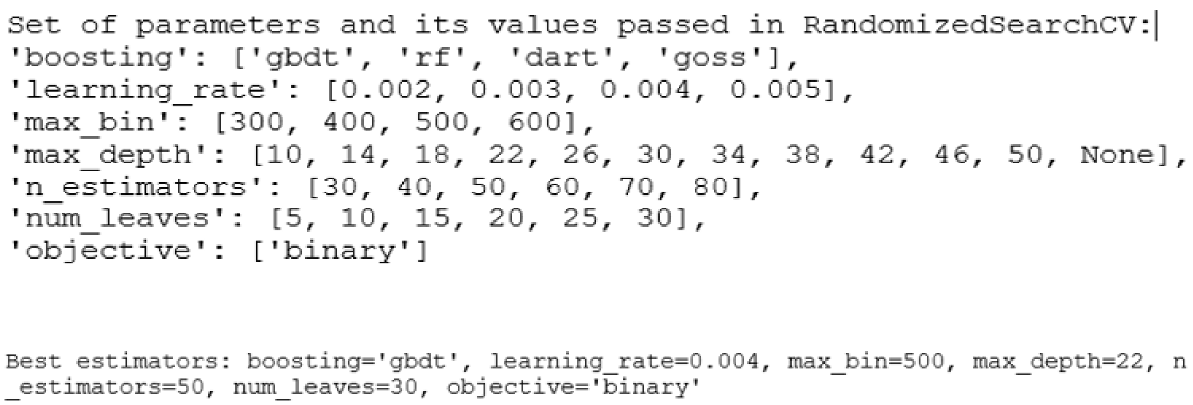
```
Best estimators: max_depth=2, max_features='sqrt', max_leaf_nodes=4, min_samples_le
af=5, min_samples_split=10
```

Figure 20 Decision Tree best parameters selection

Obtained best estimators are applied to Decision Tree classifier with best score=0.8638538125303546 and it took 20.5 seconds to finish the 3-fold Cross-Validation of 50 candidates totalling to 150 fits of combination to compute.

4.6.3 Light GBM

LightGBM can be tuned in three ways as shown explained (Parameters Tuning — LightGBM 3.0.0 documentation, 2020) these three themes are 1. For Faster Speed 2. For Better Accuracy 3. Deal with Over-fitting. The set of parameters used is a combination of achieving higher accuracy and with optimal speed using RandomizedSearchCV.



```
Set of parameters and its values passed in RandomizedSearchCV:
'boosting': ['gbdt', 'rf', 'dart', 'goss'],
'learning_rate': [0.002, 0.003, 0.004, 0.005],
'max_bin': [300, 400, 500, 600],
'max_depth': [10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, None],
'n_estimators': [30, 40, 50, 60, 70, 80],
'num_leaves': [5, 10, 15, 20, 25, 30],
'objective': ['binary']
```

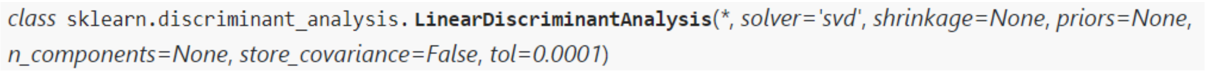
```
Best estimators: boosting='gbdt', learning_rate=0.004, max_bin=500, max_depth=22, n
_estimators=50, num_leaves=30, objective='binary'
```

Figure 21 Light GBM best parameters selection

Obtained best estimators have applied to Light GBM classifier with best score=0.948866763801198 and it took 1.5 minutes to finish the 3-fold Cross-Validation of 50 candidates totalling to 150 fits of combination to compute.

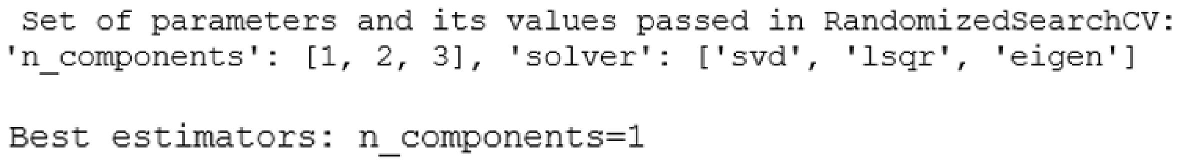
4.6.4 LDA

Parameters available for LDA



class sklearn.discriminant_analysis.**LinearDiscriminantAnalysis**(**, solver='svd', shrinkage=None, priors=None, n_components=None, store_covariance=False, tol=0.0001*)

This were tuned using RandomizedSearcCV



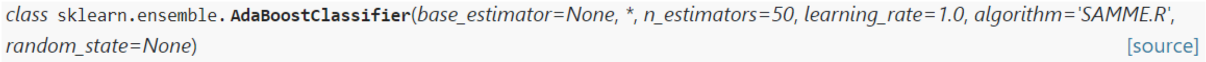
```
 Set of parameters and its values passed in RandomizedSearchCV:
'n_components': [1, 2, 3], 'solver': ['svd', 'lsqr', 'eigen']

 Best estimators: n_components=1
```

Figure 22 LDA parameters selection

Obtained best estimators are applied to Linear Discriminant Analysis classifier with best score=0.9214505423344667 and it took 1.7 seconds to finish the 3-fold Cross-Validation of 9 candidates totalling to 27 fits of combination to compute.

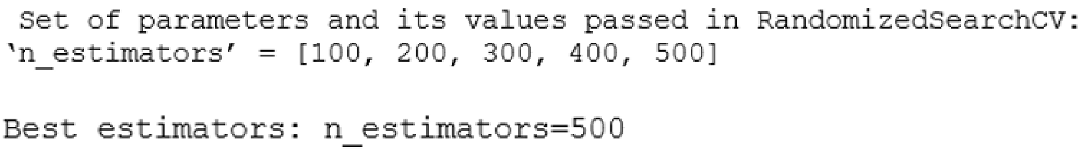
4.6.5 AdaBoost:

It is also an ensemble technique using a sequential boosting approach for classification. Parameters for classifiers are:



class sklearn.ensemble.**AdaBoostClassifier**(*base_estimator=None, *, n_estimators=50, learning_rate=1.0, algorithm='SAMME.R', random_state=None*) [source]

Above parameters are tuned using RandomizedSearcCV and collection of values passed for n_estimators is shown below:



```
 Set of parameters and its values passed in RandomizedSearchCV:
'n_estimators' = [100, 200, 300, 400, 500]

Best estimators: n_estimators=500
```

Figure 23 AdaBoost parameters selection

Obtained best estimators have applied to AdaBoost classifier with best score=0.9605755220981059 and it took 10.3 minutes to finish the 3-fold Cross-Validation of 5 candidates totalling to 15 fits of combination to compute.

4.6.6 Regression Model:

All the 5 models used for regression which are LASSO, Ridge, XGBoost, Gradient Boosted Tree and Linear Regression was built using default parameters. However, the dataset which was used for regression modelling was different form classification which had more features included in it. Due to more dimensionality and shortage of computation resource needed in implementing process of hyper-parameter tuning. It would be time-consuming and might not support the system being used for this experiment. This was one of the biggest limitations for solving this problem. However, when the data inputted for regression model, it was pre-processed and standardize which made the processing faster and the obtained result was unadulterated from noisy data.

5. Analysis and Findings

5.1 Result Discussion

The design and methodology which was used for the implementation of the experiment under consideration of multiple issues like assimilating with cloud, handling data in best industrial practice implementing different models and techniques to evaluate it has provided the result. This outcome will profoundly affect the decision-making process and management of Supply Chain Industry, this section interprets the result and discusses some major insights which will benefit the business in the long term.

Model	Accuracy%	Recall%	F1%	ROC_AUC%
Random Forest	**98.3325947**	**65.96958174**	**60.5848974**	77.6681912
LDA	84.4338577	12.81414830	22.7172717	**92.0346958**
LightGBM	91.8771696	20.51138484	33.3181749	90.3260558
AdaBoost	93.2306669	21.90321833	34.0410219	84.9888818
Decision Tree	88.8396484	16.20481080	27.5994250	90.8604781

Table 2 Metrics for Tuned Models

Performance results obtained for tuned models shows that Random Forest has the best overall score as it has the highest value for Accuracy=98%,recall=65% and F1=60% despite having the least value for AUC score. Similarly, LDA has the lowest Accuracy, recall and f1 score which is 84%, 12% and 22% respectively however the model with the highest AUC score i.e. 92%. Light GBM, AdaBoost and Decision Tree are mid-ranged models of which Decision Tree has low Accuracy, Recall and F1 score than other 2 models.

Model	Training Time(s)	Prediction Time(s)	Tuning time(s)
Random Forest	65.774	0.296	1302
LDA	0.615	**0.0**	**5.4**
Light GBM	1.448	0.06	99.5
AdaBoost	**180.171**	**3.022**	**2352**
Decision Tree	**0.308**	0.016	17.4

Table 3 Computation Time for Tuned Models

When comparing the computational time required for the model to train, deploy and tune we can see from the numbers that AdaBoost is the most time-consuming algorithm in total consuming 2535.193s. Decision Tree is the model with the least training time requirement and LDA is model which consumes the least time for a prediction.

Model	1st Important Feature	2nd Important Feature	3rd Important Feature
LDA	Type	Days for shipping (real)	Customer Id
Light GBM	Days for shipping (real)	Customer Id	Order City
AdaBoost	Days for shipping (real)	Sales	Customer Segment
Decision Tree	Type	Order Item Discount	-
Random Forest	Type	Late_delivery_risk	Days for shipping (real)

Table 4 Top 3 Important Features of Tuned Models

Feature importance for all the model showed ‘Type’ is the most important feature for 3 models: LDA, Decision Tree and Random Forest. For Light GBM and AdaBoost ‘Days for shipping (real)’ is the most important feature. ‘Days for shipping (real)’ feature in common for all the models. The unique thing here is the Decision Tree only has 2 prominent features. We can also see that descriptive features ‘Order City’ and ‘Customer Id’ is one of the important features for Light GBM and LDA respectively where Light GBM has 2 categorical feature for top 3 important feature list.

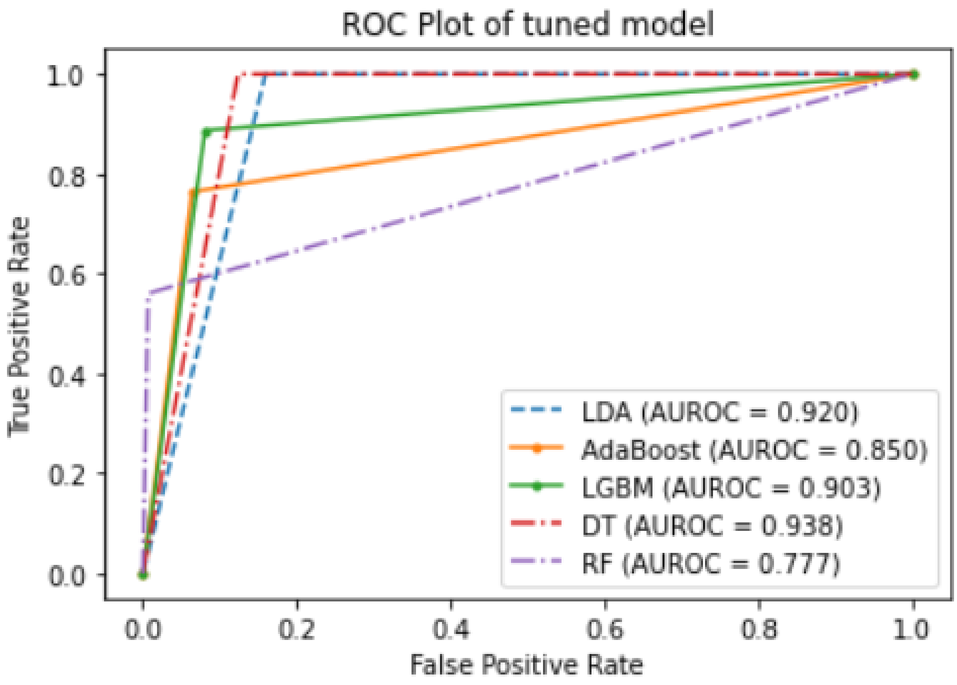


Figure 24 ROC curve for Tuned Models

The Area under the ROC curve is useful for evaluating classifier output quality. In this X-axis has False Positive rate and Y-axis has True Positive rate. This means the top left corner is the ideal point where the rate for a True Positive is 1 and for False Positive it is 0. A model with more steepness is considered as best because it is ideal for maximizing True Positive and minimizing false Positive rate. It also indicates higher the AUC score better the performance of the model in distinguishing between Negative classes and positive classes (Aniruddha Bhandari, 2020). After analysing the ROC curve for tuned models, it can be seen that in terms of least False Positive rate Random Forest is the best model but with least True Positive Rate. On the other side, Decision Tree and LDA model has the True Positive rate of 1 with higher False Positive rate compared to other models. LGBM and AdaBoost are two models which have decent values for True positive and False Positive rate.

Model	Accuracy%	Recall%	F1%	ROC_AUC%
Random Forest	**98.1294778**	**59.1129032**	**59.136748**	**79.1012545**
LightGBM	95.9764384	32.7205882	44.960848	84.1869383
AdaBoost	**89.6631952**	**17.4750037**	**29.496221**	**92.0306875**
Decision Tree	97.5496713	47.2704714	53.454928	79.9474607

Table 5 Metrics of Default Models

Performance results obtained for tuned models shows that Random Forest has the best overall score as it has the highest value for Accuracy=98%,recall=59% and F1=59% despite having the least value for AUC score. Similarly, AdaBoost has the lowest Accuracy, Recall and F1 score which is 89%, 17% and 29% respectively however the model with highest AUC score i.e. 92%. Light GBM and Decision Tree are mid-ranged models of which Light GBM has low Accuracy, Recall and F1 score than Decision Tree. This also makes Decision Tree the most suitable model for prediction after Random Forest.

Model	Training Time(s)	Prediction Time(s)
Random Forest	**53.328**	**0.406**
Light GBM	**2.009**	0.108
AdaBoost	18.422	0.372
Decision Tree	3.065	**0.005**

Table 6 Computation Time of Default Models

When comparing the computational time required for the model to train and predict we can see from the numbers that Random Forest is the most time-consuming algorithm in total consuming 53.734 s. Least Training time is for Light GBM and least Prediction Time is for Decision Tree. After Random Forest, AdaBoost is the model which consumed maximum time for Training and Prediction.

Model	1st Important Feature	2nd Important Feature	3rd Important Feature
Light GBM	Order Item Discount	Days for shipping (real)	Customer City
AdaBoost	Days for shipping (real)	Customer Segment	Customer City
Decision Tree	Type	Late_delivery_risk	Days for shipping (real)
Random Forest	Type	Late_delivery_risk	Days for shipping (real)

Table 7 Top 3 important features of Default Models

Feature Importance analysis of models with default parameters shows that Decision tree and Random Forest model have the same top 3 features which are 'Type', 'Late_delivery_risk' and 'Days for shipping (real)'. 'Days for shipping (real)' is in the top 3 places for all the 4 models whilst it is the most significant feature for AdaBoost model. 'Order Item Discount' and 'Customer Segment' are the only features which are not repeated in the top 3 places for any other model apart from Light GBM and AdaBoost respectively.

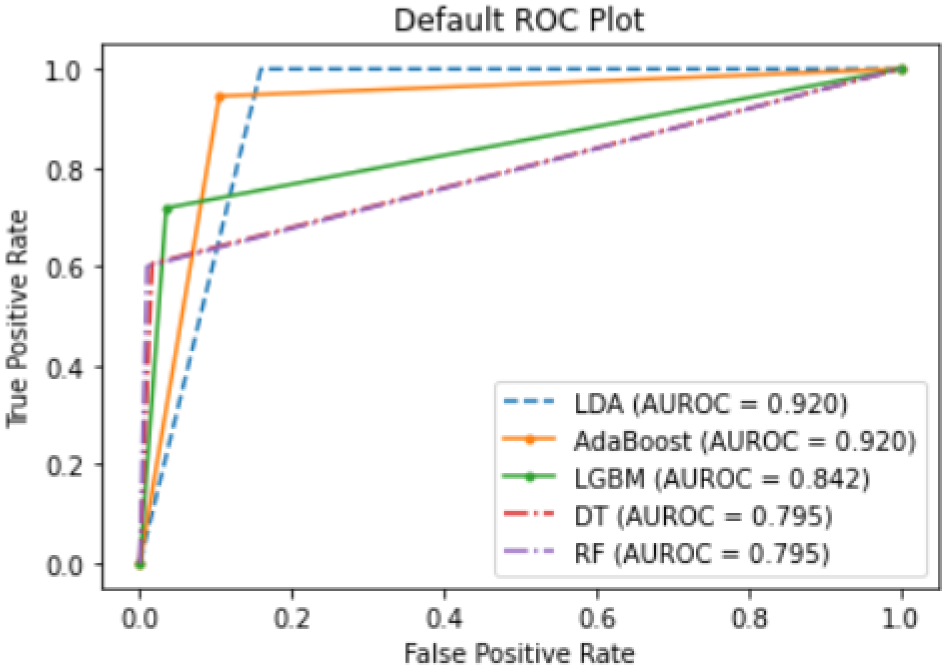


Table 8 ROC curve for Default Models

Analysing the ROC curve for models with default parameters shows that in terms of least False Positive rate Random Forest and Decision Tree are the best model but with least True Positive Rate. Whereas the LDA model has the True Positive rate of 1 with higher False Positive rate compared to other models. AdaBoost is the second-best model according to the ROC curve plot and Light GBM model is a mid-ranged making it better than Random Forest and Decision tree but dull against LDA and AdaBoost.

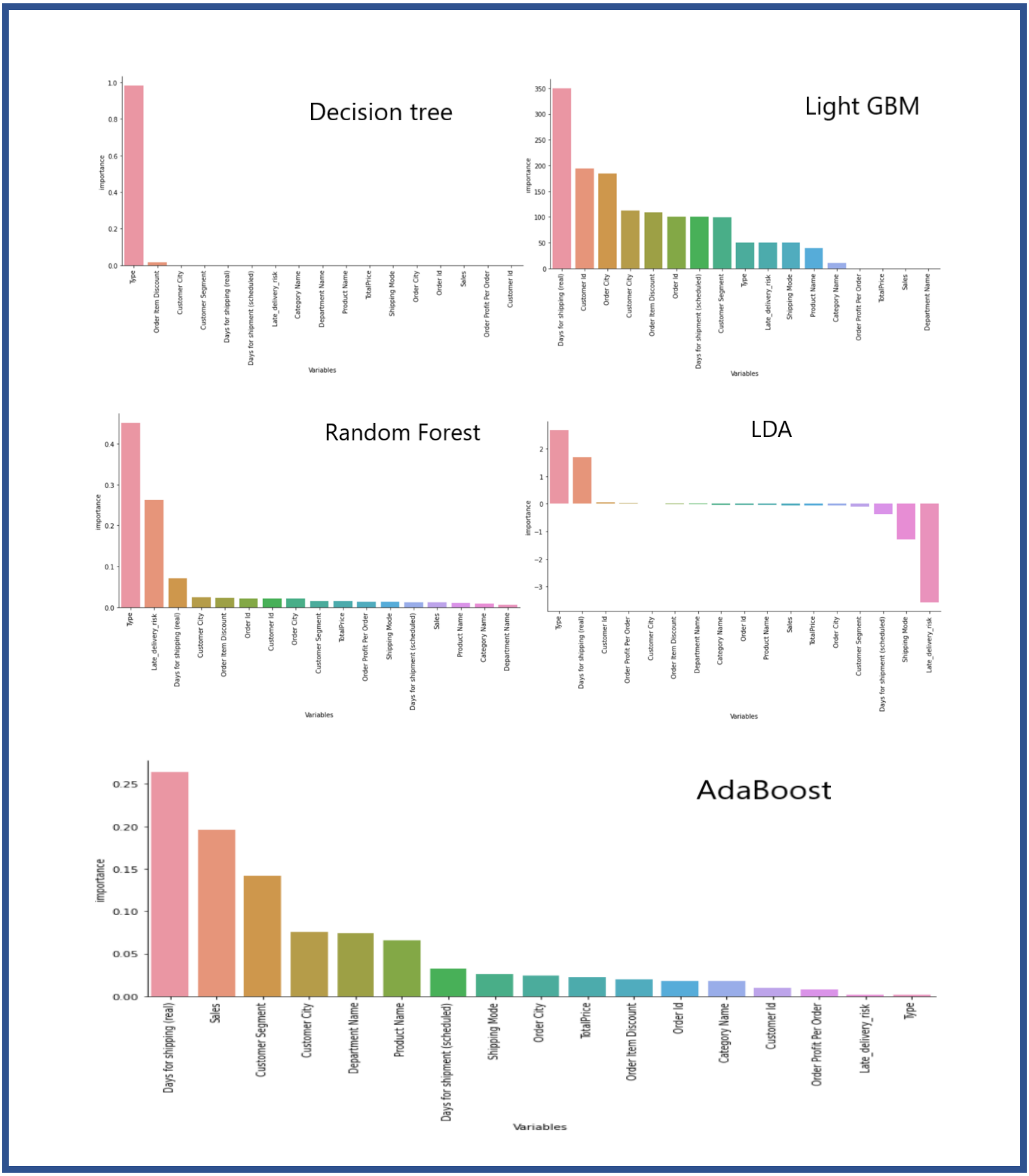


Figure 25 Important Features of Tuned Models

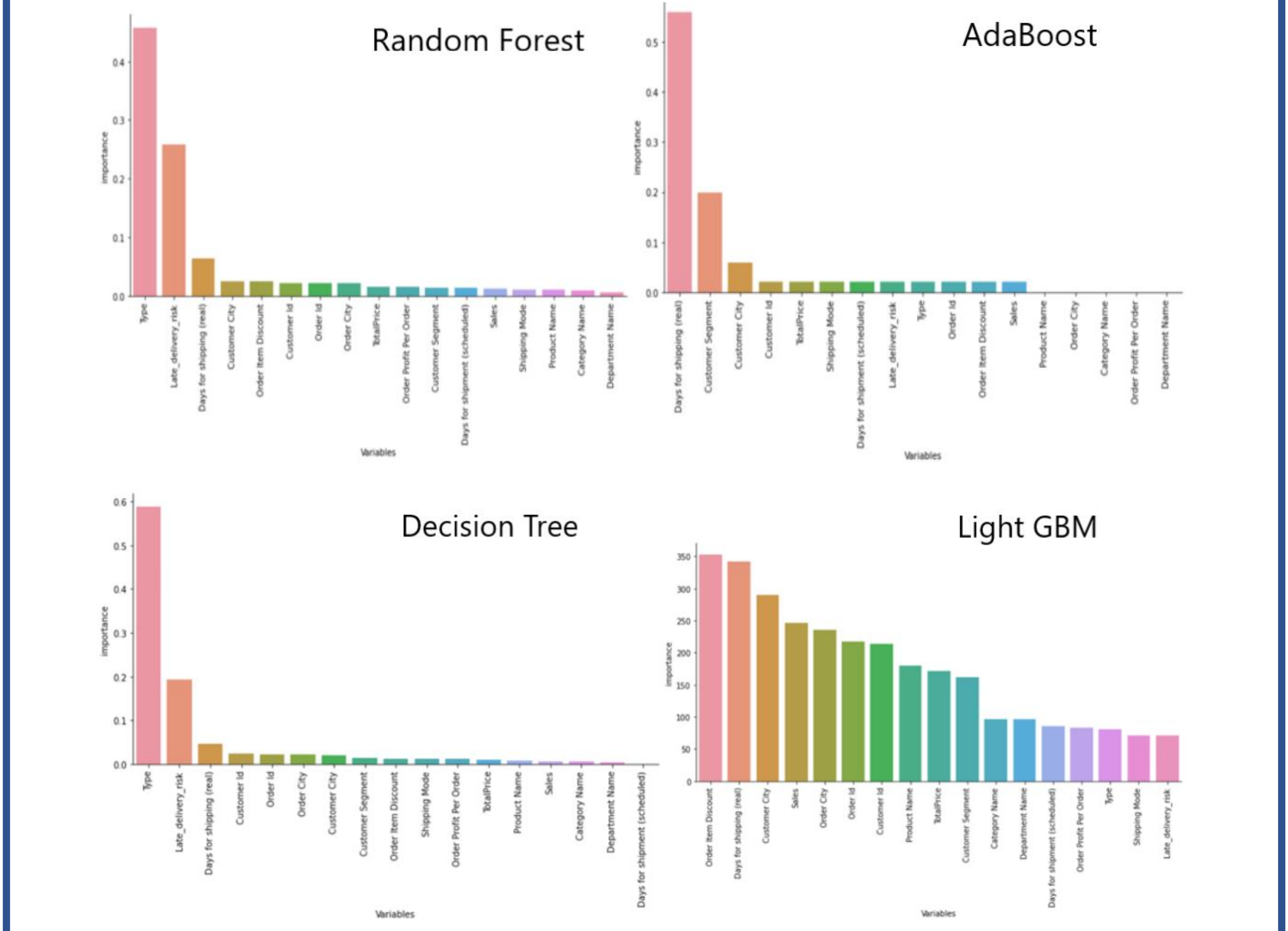


Figure 26 Important features of Default Models

Regression models are analysed based on MAE and RMSE score which are explained in the section 3.4.2. Regression models are evaluated based on the error rate which means a model with least error rate will eventually be the model with a higher accuracy rate.

Model	MAE	RMSE
LASSO	0.08339548200648535	0.11536865510851727
Ridge	0.0010036470043190342	0.001882263872915812
XGBoost	1.1313262058563323	**4.793739223668036**
Linear Regression	**0.0005448947680783427**	**0.0014938985645114012**
Gradient Boosted Tree	**1.8431233686568962**	3.392379767811959

Table 9 MAE & RMSE rate

Based on the error rates obtained for the models Linear Regression has least MAE and RMSE error rate making it the most preferable model for the application of sales prediction followed by Ridge regression model. Tree-based model XGBoost and Gradient Boosted Tree are the only models having MAE and RMSE value above 1 of which Gradient Boosted Tree is the least suitable model as it has a maximum error rate. LASSO model has also provided good results which are close to the result of Ridge regressor.

Model	Training Time(s)	Prediction Time(s)
LASSO	0.04	**0.0**
Ridge	**0.032**	0.008
XGBoost	19.919	**0.775**
Linear Regression	0.08	**0.0**
Gradient Boosted Tree	**39.152**	0.144

Table 10 Computation time for regression models

Computation time for regression models shows that Ridge model takes the least time to train and Gradient Boosted Tree is the model with maximum train time requirement. LASSO and

Linear Regression show they take 0 seconds for the prediction process whilst XGBoost consuming more time for prediction which is also the model with maximum training time requirement after Gradient Boosted Tree. It also showed that despite Gradient Boosted Tree consuming the maximum time for training it does not top the list when it comes to the time required for prediction.

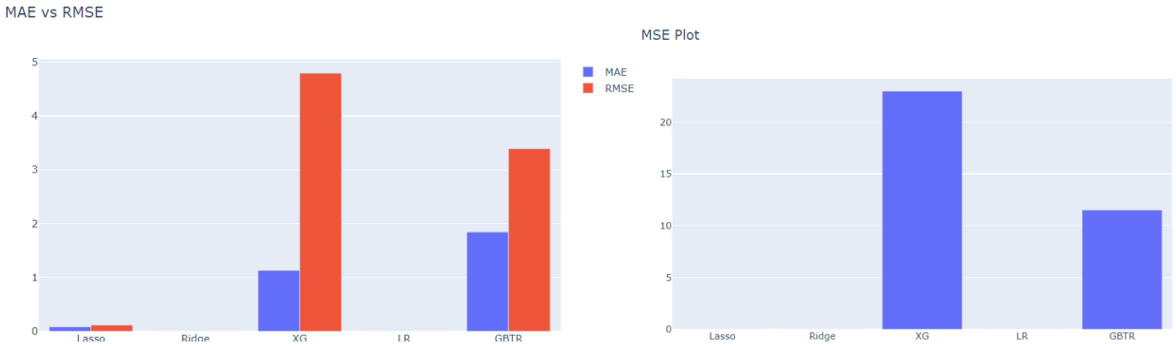


Figure 27 RMSE, MAE & MAE plot

Above graphs clearly highlights the difference in RMSE, MAE and MSE error rate for all the models.

5 .2 Strengths and Weakness:

After applying various algorithms for classification and regression problem and evaluating it against appropriate metrics, the knowledge obtained is totally unique which would have not been possible to extract from the raw data. With the help of machine learning algorithm, we understood which feature is having a major impact on fraud prediction and feature which decided the sales for that instance. As the data set was highly imbalanced with the help of SMOTE data was balanced in the best practical way without making data bias for any particular class of observation. Before applying SMOTE there were **123540** class with the genuine transaction and **2823** with the fraudulent transaction which was later balanced equally for value of **123540** observation for each class. Once the issue of unbalanced data was solved models were tuned using RandomizedSearchCV which showed that overall modelling time was increased.

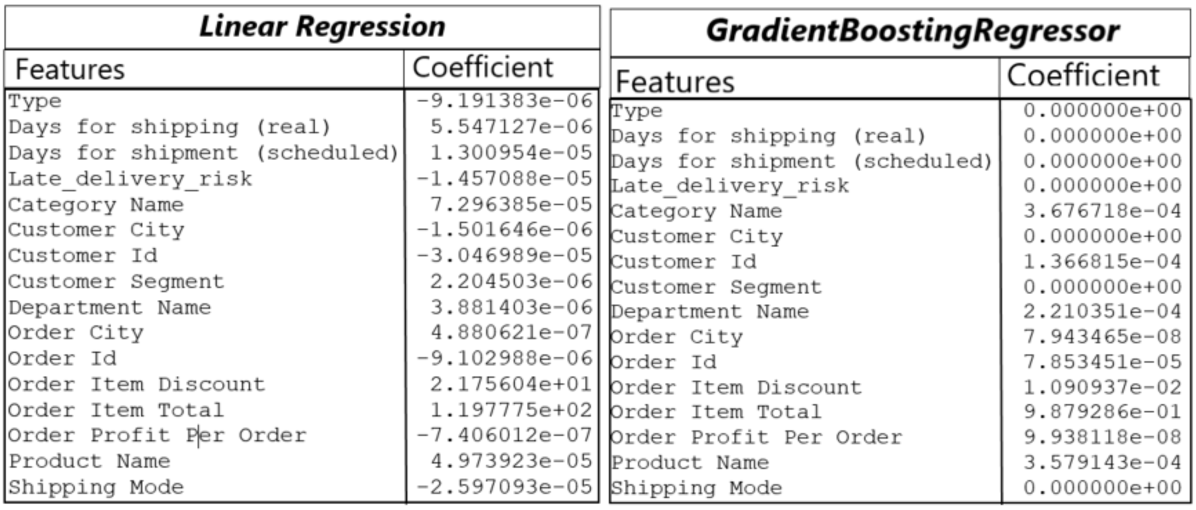
It was found that tuned models despite consuming more time for tuning they needed less time to predict once trained. Random Forest is the model which has the best Accuracy, Recall and F1 score for tuned and default parameters. It consumes **0.296 s** for prediction when parameters are tuned and **0.406 s** when parameters are set for default making the most time-consuming model for prediction when all the other models are set at default. The difference in time makes the tuned RF model to consume **0.11 s** less than the default model. When Decision tree is tuned the AUC score obtained is 90% and for default, it is 79% increasing it by **11%** for the tuned model which resulted in the loss of performance for Accuracy, Recall and F1 scores by **9%**, **31%**, and **26%** respectively for Tuned Decision Tree model.

AdaBoost which is least performing model at default setup obtained better result for Accuracy, Recall and F1 score increasing it by 4%, 4% and 5% respectively when tuned parameters were used. AUC score dropped by 8% reaching to 84% when tuned. The noticeable thing for AdaBoost model was that it consumed maximum time for training and prediction when it is was tuned but for the default set up, it only takes **10422** seconds to train hence reducing the time by **161.749** s. However, it also the model which takes maximum time for tuning i.e. **2352 s**.

Decision Tree and Light GBM are the only models in which improvement of AUC score is noticed when the parameters are tuned but on the cost of lower Accuracy, Recall and F1 performance. Training and prediction time for both the model is also improved when they are tuned making Decision Tree the model to get trained fastest (**0.308 s**) when tuned and Light GBM to predict fastest (**0.06 s**) after LDA model. The results showed that Random Forest is the best fit model in both the scenario (tuned/default) and LDA being the least suitable model for fraud prediction.

In terms of feature importance which plays a vital role in modelling as well as for decision making. Random Forest, which is the best fit model, has ‘Type’ as the most important feature which is also common for Decision tree and LDA model however it was known that ‘Type’ feature was not most important for Light GBM and AdaBoost. ‘Days for shipping (real)’ is the most important feature for all the models indicating how essential it is to predict the class of the target variable.

In Sales prediction Linear regression is the model with least error rate based on RMSE and MAE values followed by Ridge Regression. The worst score is for both the tree-based regressor model where GBT model has MAE=**1.8** highest among all the models and XGBoost having RMSE=**4.7** making it the least preferable model for the sales prediction task. Ridge being providing good result also happens to be the fastest in the training process whereas Linear Regression and LASSO model to be the fastest model for prediction process. Gradient Boosted Tree is the model which requires maximum time to train making it **39.112** seconds slower than linear regression the best performing model with MAE=**0.00054** and RMSE=**0.00149**. Below list shows coefficient values for Linear Regression (best model) and Gradient Boosting Regressor (poor model).



Linear Regression	
Features	Coefficient
Type	-9.191383e-06
Days for shipping (real)	5.547127e-06
Days for shipment (scheduled)	1.300954e-05
Late_delivery_risk	-1.457088e-05
Category Name	7.296385e-05
Customer City	-1.501646e-06
Customer Id	-3.046989e-05
Customer Segment	2.204503e-06
Department Name	3.881403e-06
Order City	4.880621e-07
Order Id	-9.102988e-06
Order Item Discount	2.175604e+01
Order Item Total	1.197775e+02
Order Profit Per Order	-7.406012e-07
Product Name	4.973923e-05
Shipping Mode	-2.597093e-05

GradientBoostingRegressor	
Features	Coefficient
Type	0.000000e+00
Days for shipping (real)	0.000000e+00
Days for shipment (scheduled)	0.000000e+00
Late_delivery_risk	0.000000e+00
Category Name	3.676718e-04
Customer City	0.000000e+00
Customer Id	1.366815e-04
Customer Segment	0.000000e+00
Department Name	2.210351e-04
Order City	7.943465e-08
Order Id	7.853451e-05
Order Item Discount	1.090937e-02
Order Item Total	9.879286e-01
Order Profit Per Order	9.938118e-08
Product Name	3.579143e-04
Shipping Mode	0.000000e+00

Table 11 Coefficient of Linear Regression Model & GBTR

Below Figure shows the sample of predicted values and actual value for Linear Regression a nd Gradient Boosted Tree Regressor. Showing the difference between actual values and pre dicted values.

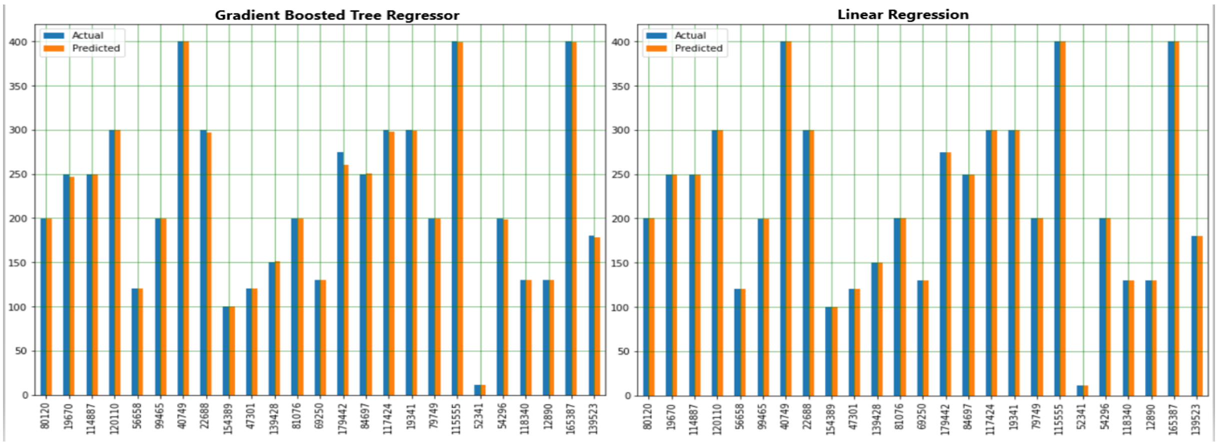


Figure 28 Comparison of predicted & actual observation

6. Conclusion

6.1 Research Overview

In this research, we have carried out a comprehensive experiment to evaluate the performance of Machine Learning models for Fraud prediction and Sales prediction. With the help of EDA, it was understood that Western Europe and Central America are the regions with the highest number of the sale however accounting for maximum revenue loss. Total loss of revenue is **-3883547.345768667**. It also indicated that this loss was due to the large frequency of fraudulent transaction. Customers mostly preferred to pay using a debit card and wired transaction rather than cash payment which indicates the need to increase the cybersecurity of payment service.

Dataset available for classification was highly unbalanced, this was solved using SMOTE technique where both the classes of target variable were made equal to **123540** observations in each class. RandomizedSearchCV is implemented for hyper-parameter tuning which proved to be better process when we need to minimise the prediction time however consuming excessive computational time.

Random Forest is the best fit model for fraud prediction having Accuracy=**98.332%** Recall=**65.969%** F1=**60.584%** and AUC=**77.668%** when tuned and Training time =65.774s, Prediction Time =0.296s and tuning time=.1302s. LDA came out as the least suitable model for fraud prediction amongst all the 5 classification models where tuned LDA model has Accuracy=**84.433%** Recall=**12.814%** F1 =**22.717%** and AUC =**92.034%**. Time consumed by LDA model to train is 0.615s, for prediction **0.0s** and tuning **5.4s**. 'Days for shipping (real)' is the most common feature and essential for all the tuned models except for Decision Tree and most important feature for Light GBM and AdaBoost model. 'Type' is also the most important feature for tuned LDA, Decision Tree and Random Forest model. Based on the ROC curve for tuned models, Random Forest has the lowest False Positive Rate whereas LDA and Decision Tree with highest True Positive Rate. ROC curve for default models showed that Random Forest and Decision Tree has shared co-ordinates while indicated lowest False Positive rate and LDA model having maximum True Positive Rate followed by AdaBoost.

For Regression task Linear regression and Ridge Regression is the best fit model for sales prediction based on RMSE and MAE error rate vales which for Linear Regression is

0.0014938985645114012 and 0.0005448947680783427. For ridge regression RMSE=0.001882263872915812 and MAE =0.0010036470043190342. Based on MAE score Gradient Boosted Tree has the highest error rate of 1.8431233686568962 and for RMSE XGBoost has the maximum error rate of 4.793739223668036.

Based on the computation time of models Ridge is fastest for training (**0.032 s**), Linear Regression and LASSO model fastest for prediction taking only **0.0 s**. Gradient Boosted Tree consuming **39.152** s for training making it the slowest model, XGBoost is the most time-consuming model when it comes for prediction as it took **0.775 s**. Setting this process on AWS helped by saving the time of managing and saving the progress on code. It also helped in storing the data on S3 bucket which helped in increasing the data fetching speed which will be a crucial factor when deployed in a real-world environment. Even though this research was at a small scale still cloud technologies helped in smooth completion of the tasks.

6.2 Limitation:

This study has some limitations which leave room for future study and researches as the research experiment was set up on a cloud infrastructure quantity of data could be more so to check the computation process to its full capacity. Also, the cost of cloud is higher if the processing is not in huge quantity, so it was like racing Ferrari against Truck. Some of the features of the dataset provided more descriptive information which could provide poor result quality and therefore increasing time for pre-processing. As these features do not have a major impact on the target variable. E.g. First Name, Last Name column for customer when Customer Unique Id is already present performing same task.

Very few research paper which has implemented Deep Learning, Artificial Neural Networks, Recurrent Neural Network and several such advance ML model for Supply Chain Industry, highlighting that more exploration needs to be done in this sector. Another factor which could save computation time, reduce the cost and most important improve the quality of the result is to develop a technique which helps in identifying whether the classified class was fraudulent or genuine and once identified as genuine then only it will be added in data which is used for predicting sales. Because in the present method even the observation which is fraudulent is considered into data used for Sales Prediction which increases the computation time and hence increases resources cost.

6.3 Future Work:

This research has just used classic machine learning models which uses tree-based and linear modelling so there is still scope of evaluating performance by using models like k-NN, Naïve Bayes, Support Vector Machines and the most fascinating in recent period Deep Learning and Neural Networks model. Also, based on the available data, Clustering algorithm can be applied to find some useful visual insights. As this data is customer-based so Customer Segmentation can be performed which will help to understand needs and to target customers in a way which will generate more profit. More rigorous feature selection and hyper-parameter tuning can be performed. Also in this research for converting categorical values into numeric Label Encoder is used and for balancing the data in classification problem SMOTE is used so the study could be conducted to compare the effect of using some other techniques for solving imbalanced dataset problem and for converting data into numeric values, to check the difference in the output.

7. Bibliography

1. Ahn, J. J. et al. (2012) ‘Using ridge regression with genetic algorithm to enhance real estate appraisal forecasting’, Expert Systems with Applications, 39(9), pp. 8369–8379. doi: 10.1016/j.eswa.2012.01.183.

2. Alfaro, E. et al. (2008) ‘Bankruptcy forecasting: An empirical comparison of AdaBoost and neural networks’, Decision Support Systems, 45(1), pp. 110–122. doi: 10.1016/j.dss.2007.12.002.

3. Alicke, K., Rachor, J. and Seyfert, A. (2016) ‘Supply Chain 4.0 – the next-generation digital supply chain’, McKinsey & Company. Available at: https://www.mckinsey.com/~/media/McKinsey/Business%20Functions/Operations/Our%20Insights/Supply%20Chain%2040%20%20the%20next%20generation%20digital%20supply%20chain/08b1ba29ff4595ebea03e9987344dcbc.pdf.

4. Amazon EC2 (2020) Amazon Web Services, Inc. Available at: https://aws.amazon.com/ec2/ (Accessed: 19 July 2020).

5. ‘Amazon S3’ (2020) Wikipedia. Available at: https://en.wikipedia.org/w/index.php?title=Amazon_S3&oldid=962707691 (Accessed: 14 July 2020).

6. Amazon S3 Features - Amazon Web Services (2020) Amazon Web Services, Inc. Available at: https://aws.amazon.com/s3/features/ (Accessed: 14 July 2020).

7. Amazon SageMaker (2017) Amazon Web Services, Inc. Available at: https://aws.amazon.com/sagemaker/ (Accessed: 16 August 2020).

8. Aniruddha Bhandari (2020) ‘AUC-ROC Curve in Machine Learning Clearly Explained’. Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/.

9. Balachandran, B. M. and Prasad, S. (2017) ‘Challenges and Benefits of Deploying Big Data Analytics in the Cloud for Business Intelligence’, Procedia Computer Science, 112, pp. 1112–1122. doi: 10.1016/j.procs.2017.08.138.

10. Barr, J., Narin, A. and Varia, J. (2011) ‘Building Fault-Tolerant Applications on AWS’, p. 15.

11. Biau, G. (2012) ‘Analysis of a Random Forests Model’, p. 33.

12. Blakeborough, L. and Correia, S. G. (2017) The scale and nature of fraud: a review of the evidence, p. 30.

13. boto3: The AWS SDK for Python (2014). Available at: https://github.com/boto/boto3 (Accessed: 1 August 2020).

14. Case Studies (2020) Amazon Web Services, Inc. Available at: https://aws.amazon.com/solutions/case-studies/ (Accessed: 19 July 2020).

15. Chen, T. and Guestrin, C. (2016) 'XGBoost: A Scalable Tree Boosting System', Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. doi: 10.1145/2939672.2939785.

16. Chengsheng, T., Huacheng, L. and Bing, X. (2017) 'AdaBoost typical Algorithm and its application research', MATEC Web of Conferences. Edited by B. Xu and Y. Chen, 139, p. 00222. doi: 10.1051/matecconf/201713900222.

17. Cheriyan, S. et al. (2018) 'Intelligent Sales Prediction Using Machine Learning Techniques', in 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE). 2018 International Conference on Computing, Electronics & Communications Engineering (iCCECE), Southend, United Kingdom: IEEE, pp. 53–58. doi: 10.1109/iCCECOME.2018.8659115.

18. Chong, A. Y. L. et al. (2017) 'Predicting consumer product demands via Big Data: the roles of online promotional marketing and online reviews', International Journal of Production Research, 55(17), pp. 5142–5156. doi: 10.1080/00207543.2015.1066519.

19. Cloud Object Storage | Store & Retrieve Data Anywhere | Amazon Simple Storage Service (S3) (2020) Amazon Web Services, Inc. Available at: https://aws.amazon.com/s3/ (Accessed: 14 July 2020).

20. Constante, F., Silva, F. and Pereira, A. (2019) 'DataCo SMART SUPPLY CHAIN FOR BIG DATA ANALYSIS'. Mendeley, 5. doi: 10.17632/8gx2fvg2k6.5.

21. Constante-Nicolalde, F.-V., Guerra-Terán, P. and Pérez-Medina, J.-L. (2019) 'Fraud Prediction in Smart Supply Chains Using Machine Learning Techniques', in International Conference on Applied Technologies. Springer, pp. 145–159.

22. De Cock, D. (2011) 'Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project', Journal of Statistics Education. Taylor & Francis, 19(3), p. null-null. doi: 10.1080/10691898.2011.11889627.

23. Elsevier | An Information Analytics Business | Empowering Knowledge (2008). Available at: https://www.elsevier.com/ (Accessed: 19 July 2020).

24. Escalante-B, A. N. and Wiskott, L. (2013) 'How to Solve Classification and Regression Problems on High-Dimensional Data with a Supervised Extension of Slow Feature Analysis', p. 37.

25. Fatourechi, M. et al. (2008) 'Comparison of Evaluation Metrics in Classification Applications with Imbalanced Datasets', in 2008 Seventh International Conference on Machine Learning and Applications. 2008 Seventh International Conference on Machine Learning and Applications, San Diego, CA, USA: IEEE, pp. 777–782. doi: 10.1109/ICMLA.2008.34.

26. Fernandez, A. et al. (2018) 'SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary', Journal of Artificial Intelligence Research, 61, pp. 863–905. doi: 10.1613/jair.1.11192.

27. Gartner Reprint (2019). Available at: https://www.gartner.com/doc/reprints?id=1-1CMAPXNO&ct=190709&st=sb (Accessed: 19 July 2020).

28. Ghosh, J. and Shuvo, S. B. (2019) 'Improving Classification Model's Performance Using Linear Discriminant Analysis on Linear Data', in 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT). 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India: IEEE, pp. 1–5. doi: 10.1109/ICCCNT45670.2019.8944632.

29. Global Infrastructure (2020) Amazon Web Services, Inc. Available at: https://aws.amazon.com/about-aws/global-infrastructure/ (Accessed: 19 July 2020).

30. Hu, X., Chen, H. and Zhang, R. (2019) 'Credit Card Fraud Detection using LightGBM with Asymmetric Error Control', in 2019 Second International Conference on Artificial Intelligence for Industries (AI4I). 2019 Second International Conference on Artificial Intelligence for Industries (AI4I), Laguna Hills, CA, USA: IEEE, pp. 91–94. doi: 10.1109/AI4I46381.2019.00030.

31. Huang, J.-C. et al. (2020) 'Predictive modeling of blood pressure during hemodialysis: a comparison of linear model, random forest, support vector regression, XGBoost, LASSO regression and ensemble method', Computer Methods and Programs in Biomedicine, 195, p. 105536. doi: 10.1016/j.cmpb.2020.105536.

32. Jain, A., Menon, M. N. and Chandra, S. (2015) 'Sales Forecasting for Retail Chains', p. 6.

33. Karnin Zohar (2018) 'Algorithms in Amazon SageMaker', Amazon Web Services, Inc.

34. Ke, G. et al. (2017) ‘LightGBM: A Highly Efficient Gradient Boosting Decision Tree’, in Guyon, I. et al. (eds) Advances in Neural Information Processing Systems 30. Curran Associates, Inc., pp. 3146–3154. Available at: http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf (Accessed: 27 July 2020).

35. Khandelwal, P. (2017) ‘Which algorithm takes the crown: Light GBM vs XGBOOST?’ Analytics Vidhya.

36. Korolev, M. and Ruegg, K. (2015) ‘Gradient Boosted Trees to Predict Store Sales’, Stanford University, p. 6.

37. Lanteri, S. (1992) ‘Full validation procedures for feature selection in classification and regression problems’, Chemometrics and Intelligent Laboratory Systems, 15(2–3), pp. 159–169. doi: 10.1016/0169-7439(92)85006-O.

38. Li, B. et al. (2016) ‘Predicting online e-marketplace sales performances: A big data approach’, Computers & Industrial Engineering, 101, pp. 565–571. doi: 10.1016/j.cie.2016.08.009.

39. M, H. and M.N, S. (2015) ‘A Review on Evaluation Metrics for Data Classification Evaluations’, International Journal of Data Mining & Knowledge Management Process, 5(2), pp. 01–11. doi: 10.5121/ijdkp.2015.5201.

40. Marston, S. et al. (2011) ‘Cloud computing — The business perspective’, Decision Support Systems, 51(1), pp. 176–189. doi: 10.1016/j.dss.2010.12.006.

41. McKay, S. (2012) ‘How to communicate risks using a heat map’, in. American Institute of Certified Public Accountants, Inc.

42. Meixell, M. J. and Gargeya, V. B. (2005) ‘Global supply chain design: A literature review and critique’, Transportation Research Part E: Logistics and Transportation Review, 41(6), pp. 531–550. doi: 10.1016/j.tre.2005.06.003.

43. Mell, P. and Grance, T. (2011) ‘The NIST Definition of Cloud Computing’, National Institute of Standards and Technology Special Publication U.S. Department of Commerce, p. 7.

44. Mojtahed, V. (2019) ‘Big Data for Fraud Detection’, in Cecconi, F. and Campennì, M. (eds) Information and Communication Technologies (ICT) in Economic Modeling. Cham: Springer International Publishing (Computational Social Sciences), pp. 177–192. doi: 10.1007/978-3-030-22605-3_11.

45. Ogutu, J. O., Schulz-Streeck, T. and Piepho, H.-P. (2012) ‘Genomic selection using regularized linear regression models: ridge regression, LASSO, elastic net and their extensions’, BMC Proceedings, 6(S2), p. S10. doi: 10.1186/1753-6561-6-S2-S10.

46. Parameters Tuning — LightGBM 3.0.0 documentation (2020). Available at: https://lightgbm.readthedocs.io/en/latest/Parameters-Tuning.html (Accessed: 16 August 2020).

47. Pedregosa, F. et al. (2011) ‘Scikit-learn: Machine Learning in Python’, Journal of Machine Learning Research, 12(85), pp. 2825–2830.

48. Persico, V., Montieri, A. and Pescapè, A. (2016) ‘On the Network Performance of Amazon S3 Cloud-Storage Service’, in 2016 5th IEEE International Conference on Cloud Networking (Cloudnet). 2016 5th IEEE International Conference on Cloud Networking (Cloudnet), pp. 113–118. doi: 10.1109/CloudNet.2016.16.

49. Raschka, S., Patterson, J. and Nolet, C. (2020) ‘Machine Learning in Python: Main developments and technology trends in data science, machine learning, and artificial intelligence’, arXiv:2002.04803 [cs, stat]. Available at: http://arxiv.org/abs/2002.04803 (Accessed: 15 August 2020).

50. Rehurek, R. (2015) smart-open: Utils for streaming large files (S3, HDFS, GCS, Azure Blob Storage, gzip, bz2...). Available at: https://github.com/piskvorky/smart_open (Accessed: 1 August 2020).

51. ‘Ridge Regression’ (2020), p. 21.

52. Rokach, L. and Maimon, O. (2005) ‘Decision Trees’, in The Data Mining and Knowledge Discovery Handbook, pp. 165–192. doi: 10.1007/0-387-25465-X_9.

53. Rushin, G. et al. (2017) ‘Horse race analysis in credit card fraud—deep learning, logistic regression, and Gradient Boosted Tree’, in 2017 Systems and Information Engineering Design Symposium (SIEDS). 2017 Systems and Information Engineering Design Symposium (SIEDS), Charlottesville, VA, USA: IEEE, pp. 117–121. doi: 10.1109/SIEDS.2017.7937700.

54. Safavian, S. R. and Landgrebe, D. (1991) ‘A survey of decision tree classifier methodology’, IEEE Transactions on Systems, Man, and Cybernetics. IEEE Transactions on Systems, Man, and Cybernetics, 21(3), pp. 660–674. doi: 10.1109/21.97458.

55. Sammut, C. and Webb, G. I. (eds) (2010) Encyclopedia of Machine Learning. Boston, MA: Springer US. doi: 10.1007/978-0-387-30164-8.

56. Schoenherr, T. and Speier-Pero, C. (2015) ‘Data Science, Predictive Analytics, and Big Data in Supply Chain Management: Current State and Future Potential’, Journal of Business Logistics, 36(1), pp. 120–132. doi: 10.1111/jbl.12082.

57. Share & Manage Research Datasets - Mendeley (2019). Available at: https://www.mendeley.com/datasets (Accessed: 19 July 2020).

58. Singh, A. and Jain, A. (2019) ‘Adaptive Credit Card Fraud Detection Techniques Based on Feature Selection Method’, in Bhatia, S. K. et al. (eds) Advances in Computer Communication and Computational Sciences. Singapore: Springer Singapore (Advances in Intelligent Systems and Computing), pp. 167–178. doi: 10.1007/978-981-13-6861-5_15.

59. SMOTE explained for noobs - Synthetic Minority Over-sampling TEchnique line by line · Rich Data (2020). Available at: https://rikunert.com/SMOTE_explained (Accessed: 7 August 2020).

60. Srivastava, T. (2019) Evaluation Metrics Machine Learning. Available at: https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/# (Accessed: 22 July 2020).

61. Sun, Y., Wong, A. K. C. and Kamel, M. S. (2009) ‘CLASSIFICATION OF IMBALANCED DATA: A REVIEW’, International Journal of Pattern Recognition and Artificial Intelligence, 23(04), pp. 687–719. doi: 10.1142/S0218001409007326.

62. Taylor, D. A. H. (1996) ‘Global Cases in Logistics and Supply Chain Management’, in. International Thompson Business Press.

63. Tharwat, A. et al. (2017) ‘Linear discriminant analysis: A detailed tutorial’, Ai Communications, 30, pp. 169-190,. doi: 10.3233/AIC-170729.

64. Tibshirani, R. (1994) ‘Regression Shrinkage And Selection Via The LASSO’, Department of Statistics Stanford University. Available at: https://statistics.stanford.edu/sites/g/files/sbiybj6031/f/EFS%20NSF%20465.pdf.

65. Velte, T., Velte, A. and Elsenpeter, R. (2009) Cloud Computing, A Practical Approach. McGraw-Hill.

66. Viktorovich, P. A. et al. (2018) ‘Predicting Sales Prices of the Houses Using Regression Methods of Machine Learning’, in 2018 3rd Russian-Pacific Conference on Computer Technology and Applications (RPC). 2018 3rd Russian-Pacific Conference on Computer

Technology and Applications (RPC), Vladivostok: IEEE, pp. 1–5. doi: 10.1109/RPC.2018.8482191.

67. Wei, S. et al. (2019) 'A Novel Noise-Adapted Two-Layer Ensemble Model for Credit Scoring Based on Backflow Learning', IEEE Access, 7, pp. 99217–99230. doi: 10.1109/ACCESS.2019.2930332.

68. What is AWS (2020) Amazon Web Services, Inc. Available at: https://aws.amazon.com/what-is-aws/ (Accessed: 19 July 2020).

69. Zage, D., Glass, K. and Colbaugh, R. (2013) 'Improving supply chain security using big data', in 2013 IEEE International Conference on Intelligence and Security Informatics. 2013 IEEE International Conference on Intelligence and Security Informatics (ISI), Seattle, WA, USA: IEEE, pp. 254–259. doi: 10.1109/ISI.2013.6578830.

70. Zaidi, N. (2015) Feature Engineering in Machine Learning. doi: 10.13140/RG.2.1.3564.3367.