Analysis Understanding Online Shoppers' Purchase Intentions: Using Machine Learning

Harshitha Gyada
2203A52025
Department of Computer Science Engineering
SR University
Hanamkonda , India
2203A52025@sru.edu.in

***Abstract*—**This project focuses on predicting online shoppers' purchasing intentions using machine learning. We collect and pre processed data, engineer features, and select the best-performing models. After rigorous testing, we deploy the model to improve e-commerce strategies, increase user satisfaction, and boost sales. Continuous monitoring and proper documentation ensure long term success. The project aims to empower businesses with data driven insights to enhance their online shopping 5 experience and drive growth.

***Index Terms*—Cosmic Ray Classification, Astrophysical Data Analysis, Random Forest Classifier, Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Decision Trees, Logistic Regression, Feature Importance, Data Preprocessing and Scaling, Oversampling Techniques, Model Evaluation Metrics**

I. Introduction

In the ever-evolving digital landscape, the way con sumers shop has gone through a pro a transformation has taken place, with online shopping becoming the dominant force.Understanding the intricate factors that influence on line shoppers' intentions have become a critical focus for businesses that want to succeed in the highly competitive e commerce environment. This the project embarks on a journey through the fascinating shopping realm of online shoppers' intentions, using the power of machine learning (ML) to uncover valuable insights that it can shape business strategies, improve user experience and ultimately increase sales. With digital age that requires data- driven decision-making, ML has proven indispensable capable tool, capable of deciphering the complex web of consumer choices and preferences. At a time when convenience and accessibility have become paramount, convenience online shopping has captured the attention of consumers worldwide. This project is sinking deep into the dy namic world of e-commerce and seeks to harness the potential of ML provide a comprehensive understanding of what drives online shoppers' purchases intentions. This understanding is essential for businesses to remain competitive ever-expanding digital market. Machine learning, its advanced algorithms and data analysis capabilities is ready to play a key role in deciphering the complex factors that influence online commerce axis Using ML, this project finds to open the underlying variables manage the application of online shoppers, by equipping businesses with useful information they can inform their Strategies and optimize their offers. The project focuses on leverage ML ensures that research is at the forefront of technological advancement and is capable providing deep insights that traditional research methods might miss.

This project will be a comprehensive survey of online influ encing factors purchase intentions of shoppers. The objectives include the identification of key determinants, e.g. such as product quality, price, website usability, customer reviews and social media influence. By utilizing ML models, the study will not only identify these factors, but also quantify their impact, allowing businesses to effectively prioritize their efforts and resources. The expected outcomes of this project are twofold: to provide businesses with the insights they need to adapt and thrive in the online marketplace,and to contribute to the growing body of knowledge on consumer behavior in e commerce.

In the world of shopping in online growing to recreate the buyer experience, acknowledging the power of online shop pers' purchasing intentions is very much importance. This project, rooted in the power of machine learning, finds to a way to enigmatic realm of consumer decision-making. By using advanced Machine Learning algorithms and large data analysis, it aims to find the complex web of factors that shape the choices of online shoppers, it offering businesses invaluable insights that can steer them toward success in the fiercely competitive digital marketplace.

II. Literature Review

1.Ali Khandokar, A Islam, Salekul Islam, and Swakkhar Shatabda proposed a customer's purchase intention prediction model using feature selection techniques and oversampling methods for improved performance .The authors compared their method with existing works by Sakar et al.,Baati et al.. and Song et al., showcasing superior performance in terms of accuracy, TNR, and F1-Score .

2.Tayba Asgher analyzed online customer purchase intention using ML algorithms. Analyzed customer purchase intention using machine learning algorithms. Created customer profiles based on transaction history and viewing pages. Utilized clustering schemes and clickstream data for customer analysis. Used data mining techniques to predict customer purchase intention.

3.Zhenyu Liu, Xinyi Ma. Proposed predictive model for user purchase behavior using machine learning. Proposed a combined prediction model using Stacking method with decision trees. Integrated Light GBM, XG Boost, and Random Forest models for predictions. Evaluated predictive performance using real retail sales data.

4.Reference Manual by Rossum and Drake. Real-time prediction of online shoppers' purchasing intention by Sakar et al. Understanding consumer journey using attention based recurrent neural networks by Zhou et al.Studied purchase intentions of online customers using machine and deep learning models. Identified important features impacting customers' purchase intentions.Used Shapley Additive Explanation method to measure feature importance. Explored ensemble learning models and deep learning model performance.

5.Daksh Kapoor, Achirangshu Chakraborty, Sunita Daniel conducted online shopping preference studyClassified customers based on online shopping preferences using machine learning techniques. Analyzed dataset using bagging and boosting algorithms to predict purchasing intention. Identified month with highest revenue and customer visits for online shopping. Utilized Gradient Boosting algorithm for accurate prediction of consumer behavior.

6.Jiangtao Qiu.Proposed COREL model for customer purchase behavior prediction in e-commerce .Investigated factors affecting customer purchase decisions in e-commerce context.Proposed methods to quantify customer needs, product popularity, and preferences. Developed a predictive model called COREL for customer purchase behavior.

7.Priyank Sirohi, Niraj Singhal, Pradeep Kumar, Mahboob Alam.Proposed a customer behavior prediction model using gradient boosting algorithm Proposed customer behavior prediction using gradient boosting algorithm. Analyzed correlation matrix for customer behavior analysis.

8.Shaifali Yadav, M.Tech student at Devi Ahilya Vishwavidyalaya, India. Applied machine learning algorithms on e-commerce clickstream data. Compared accuracy, precision, recall, and F1 score of various algorithms. Conducted hyperparameter tuning to improve accuracy. Used logistic regression, decision tree, random forest, and more algorithms.

9.Andrew Frazier, Fatbardha Maloku, Xinzi Li, Yichun Chen, Yeji Jung, Bahman Zohuri.Proposed machine learning for customer segmentation using web browsing analytics.Identified online shoppers' purchasing intentions using machine learning. Utilized Pipeline tool from Scikit Learn for data preprocessing and modeling.Explored impact of numerical features on target class through EDA .

10. Md Shahriare Satu, Syed, Faridul Islam, Md Satu.Proposed a model using machine learning to predict buying intentions.Developed a machine learning model to predict online purchase intentions. Applied various feature engineering and classification techniques. Analyzed outcomes of different classifiers for online shoppers' buying instances. Detected customer online purchase intention using the proposed model.

11.Yap Chau Tean.The author's work focuses on data mining techniques for e-commerce. Yap Chau Tean conducted research on predicting online shoppers' purchase intentions. Applied six machine learning algorithms to classify online shoppers' data.Conducted data pre-processing using over-sampling, under-sampling, and hybrid sampling methods.Tested AdaBoost and Bagging ensemble learning methods but showed no improvement.Analyzed unbalanced data set and its impact on prediction accuracy.

12.Paola Furtado, Rosales M´ aster En Econom´ ıa, Diego Mar´ ın Santos.Applied supervised machine learning to predict online purchase intentions. Applied supervised machine learning techniques to predict purchase intention. Used database with 18 variables divided into training and evaluation sets. Balanced training set by random downsampling.Configured models with two sets of predictors using attribute selection. Compared models based on area under the curve (AUC).

13.Yi Lim, Abdullah Osman, Shahrul Nizam Salahuddin, Abdul Rahim Romle, Safizal Abdullah.Conducted research on online shopping behavior, purchase intention, and subjective norms.Analyzed data using SPSS and AMOS for hypothesis testing. Developed hypotheses based on previous studies on online shopping behavior. Explored subjective norm and perceived usefulness in online shopping behavior.

14.Muhammad Adlansyah Muda, Radha, Ayu Aswari, Muhammad Ahsan. Authors focus on predicting online shopper's purchasing intention using machine learning.Used binary logistic regression, decision tree, and random forest methods. Employed oversampling and feature selection for model refinement. Identified key revenue factors: new visitors, low bounce rate, etc. Achieved 88.21 accuracy with random forest model using cross-validation.

III.PROPOSED APPROACH

1.Data Description

Analyzing online shoppers' purchase intentions using ma chine learning involves predicting consumer behavior in e commerce to enhance personalized recommendations and in crease revenue. Machine learning models can identify pur chase intentions early, enabling real-time offers and discounts to be tailored before a session ends, crucial for anonymous e-shoppers. Understanding customer behavior through data mining techniques can improve satisfaction, efficiency, and engagement in the shopping process, leading to a competitive advantage with higher conversion rates.

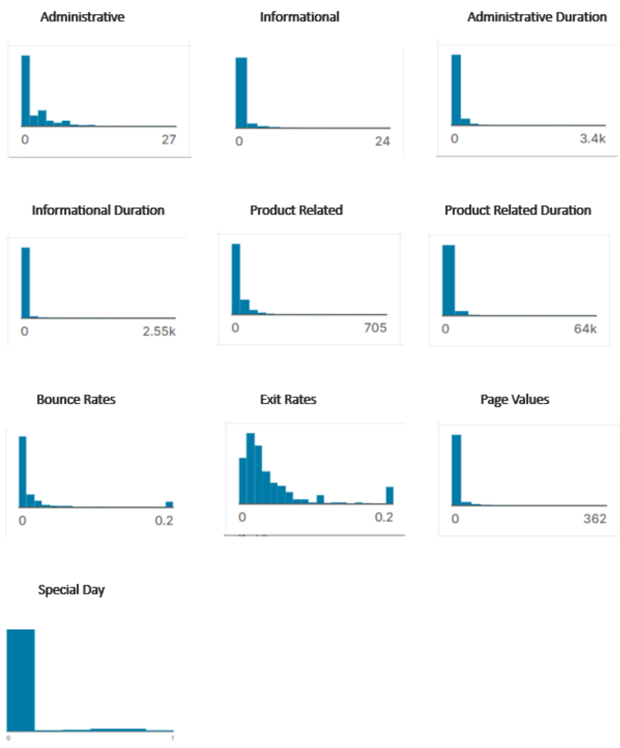
Various algorithms like SVM ensembles, decision trees, and neural networks are utilized to predict purchase intentions, aiding retailers in reaching a broader customer base and enhancing platform effectiveness. This approach is vital for e-commerce platforms to cater to the needs of anonymous shoppers and improve overall user experience and profitability.

	Administrative	Administrative_Duration	Informational	Informational_Duration	ProductRelated	ProductRelated_Duration	BounceRates	ExitF
0	0	0.0	0	0.0	1	0.000000	0.20	
1	0	0.0	0	0.0	2	64.000000	0.00	
2	0	0.0	0	0.0	1	0.000000	0.20	
3	0	0.0	0	0.0	2	2.666667	0.05	
4	0	0.0	0	0.0	10	627.500000	0.02	

2. Data analysis

These histograms help you understand how each feature contributes to the classification problem. Features with distinct, non-overlapping distributions for the two classes are usually more informative for classification. Features with substantial overlap may not be as useful for distinguishing between classes.

By analyzing these histograms, you can make informed decisions about feature selection, model selection, and feature construction to improve the performance of your classification model.



Correlation Matrix

A correlation matrix, also known as a matrix of correlation coefficients, it is a table or matrix that displays correlation coefficients amount multiple variables in a data set. Correlation

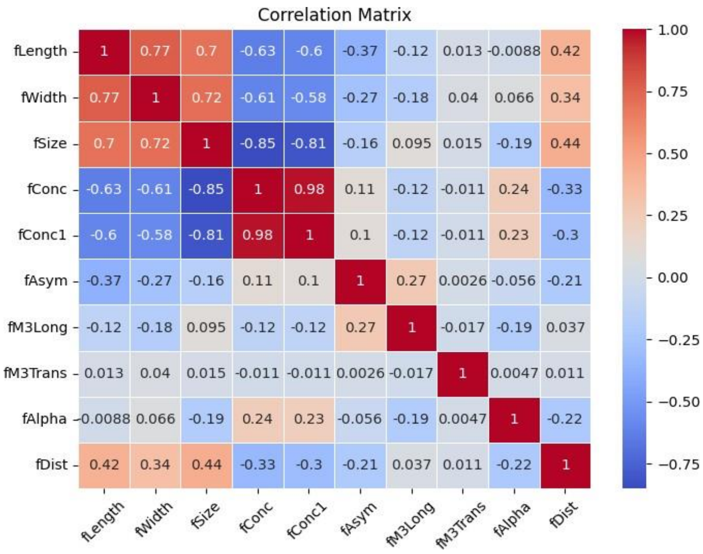
coefficients quantify the degree and direction of the linear relationship between pairs variables. In a correlation matrix, every cell tells a co-relation between the two variables.

Correlation coefficients typically fall in the range -1-1, where:.

1 indicates perfect positive correlation: When one variable increases, so does the other increases proportionally.

-1 indicates perfect negative correlation: When one variable increases, so does the other decreases proportionally.

0 indicates no linear correlation: The variables are not related in a linear manner.



3. Data preprocessing

Preprocessing the data is a vital step in the machine earning, which includes cleaning, transshaping and organizing raw data into a format suitable for modeling. It plays a significant role in providing high-quality data and machine learning so that the model can learn meaningful patterns. Here is a detailed explanation of the various aspects:

Data cleaning:

Handling missing values: Identify and handle missing data, either by deleting rows or filling in missing values using techniques such as mean, median, or interpolation. There are no null values in this dataset.

Data transformation:

Feature Scale: Normalize or standardize numeric features to ensure that different features are on a similar scale. Common techniques include Min-Max scaling and z-score normalization.

Data distribution:

Training, Validation and Testing Split: Split the data set into training, validation and testing sets. The training set is used to train the model, the validation set is used to tune the hyperparameters, and the test set is used to evaluate the performance of the model.

Working with string data :

Text Preprocessing- For natural language processing tasks, preprocess text data by tokenizing, removing stop words, deriving or lemmatizing, and converting text to numerical representations (eg, TF-IDF or word embedding).

PageValues	SpecialDay	OperatingSystems	Browser	Region	TrafficType	Weekend	Revenue
0.0	0.0	0.000000	0.000000	0.000	0.000000	0.0	0.0
0.0	0.0	0.142857	0.083333	0.000	0.052632	0.0	0.0
0.0	0.0	0.428571	0.000000	1.000	0.105263	0.0	[illegible]
0.0	0.0	0.285714	0.083333	0.125	0.157895	0.0	
0.0	0.0	0.285714	0.166667	0.000	0.157895	1.0	

Model Selection :

We selected several machine learning models for our analysis: Logistic Regression, Support Vector Machines, Perceptron, and K-Nearest Neighbors, Decision Tree, Random Forest Classifier, AdaBoost Classifier . These models were selected based on their suitability for binary classification tasks.

4. Feature Scaling

We used StandardScaler from the scikit-learn library, which standardizes features by removing the mean and scaling to a unit of variance. This method is defined by the formula:

$$z = \frac{(x - u)}{s}$$

where

x is the value of the function,

u is the mean of the training samples and

s is the standard deviation of the training samples. Data standardization is a common requirement of many machine learning estimators: they can misbehave if the individual functions do not more or less look like standard normally distributed data (e.g. Gaussian with zero mean and unit variance).

IV.SIMULATION

Results

Perceptron Model

Perceptron model is related to a machine learning algorithm on behalf of supervised learning of different binary classification segments. It is a ability of a system to find data in such a manner which is same as a way humans use there knowledge in the world where they were in.

Step 1: $\sum wi*xi = x1*w1 + x2*w2 + \ldots wn*xn$

$\sum wi*xi + b$ (b=bias)

step 2: $Y = f(\sum wi*xi + b)$

```
Classification Report:
              precision    recall  f1-score   support

         0.0       0.92      0.78      0.84      2055
         1.0       0.37      0.64      0.47       411

    accuracy                           0.76      2466
   macro avg       0.64      0.71      0.66      2466
weighted avg       0.83      0.76      0.78      2466
```

Figure 5. classification report

Model production was evaluated by the precision, recall, and the F1 score metrics along with precision score. The dataset consisted of 2,466 samples divided into two classes: Class 0 and Class 1. The model achieved an overall accuracy of 92 percent, indicating its high predictive power. Class-specific performance metrics revealed impressive decision, recall, and F1 scores, further underscoring the effectiveness of model. Accuracy: The model demonstrated an overall accuracy of 83 percent, indicating its ability to correctly classify most cases

Class0(Negativeclass)

Recall: The recall for class 0 is 78 percent, situates that almost all true class 0 cases were identified correctly.

F1-Score: The F1-score for the Class 0 is 84 percent, which representing a balanced harmonic mean of precision and recall for Class 0.

class 1(Positive class)

Precision: Class 1 is 83 percent, shows that all the predicted Class 1 instances were indeed Class 1.
Recall: Class 1 is 76 percent, indicating that almost all Class 1instances were correctly identified.

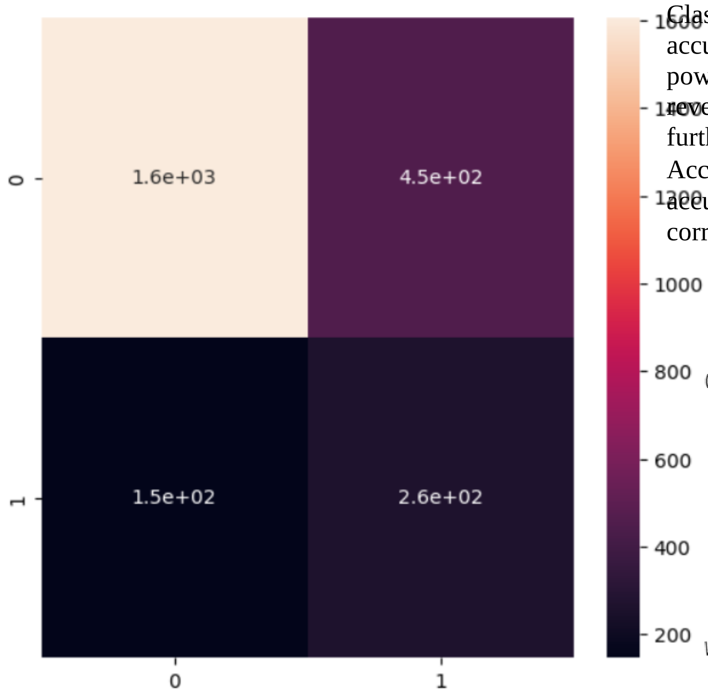


Figure 6. confusion matrix

It’s evident from the confusion matrix that:

True Negatives (TN): There are 1.6e+03 values for which both the actual class and the Perceptron replica prediction were negative (0). These are correctly classified values where the model correctly figured the pessimistic class.
False Positives (FP): There are 4.5e+02 values where the actual class is negative (0), but the model wrongly predicted them as positive (1). These represent Type I errors, where the model incorrectly identified negative instances as positive.
False Negatives (FN): There are 1.5e+02 instances where the actual class is positive (1), but the model wrongly predicted them as negative (0). These are Type II errors, indicatingthat the model missed some positive instances.
True Positives (TP): There are 2.6e+02 instances for which both the actual class and the model’s prediction were positive (1). These are correctly classified instances where the model accurately identified the positive class.

Logistic Model
Logistic regression (classification related replica).

This algorithm is used to distinguish new data using unceasing and separate data sets. Model ability was evaluated using precision, recall and F1 score metrics,along with an accuracy score. The data set consisted of 3804 samples divided into two classes: Class 0 and Class 1. The model achieved an overall accuracy of 99 percent,testifies to its high predictive power.Class-specific performance metrics have been revealed impressive accuracy, recall and F1 scores, further underscoring the effectiveness of the repilica..
Accuracy: The model demonstrated an overall accuracy of 99 percent, indicating its ability to correctly classify most cases.

Classification Report:

	precision	recall	f1-score	support
0.0	0.88	0.98	0.93	2055
1.0	0.74	0.34	0.46	411
accuracy			0.87	2466
macro avg	0.81	0.66	0.70	2466
weighted avg	0.86	0.87	0.85	2466

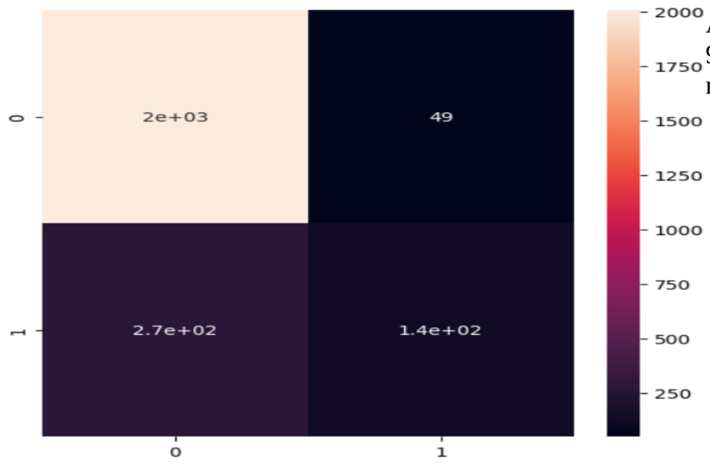
Figure 7. Classification Report.

Class 0 (Negative Class):

Precision: Class 0 is 88 percent, figured that almost all predicted Class 0 instances were indeed Class 0.
Recall: Class 0 is 98 percent, recommended that nearly all actual Class 0 instances were correctly identified.
F1-Score: Class 0 is 93 percent, gives a balanced harmonic mean of precision and recall for Class 0.

Class 1 (Positive Class):

Precision: Class 1 is 86 percent, illustrating that all predicted Class 1 instances were indeed Class 1.
Recall: Class 1 is 87 percent, proposed that almost all true Class 1 instances were correctly identified.
F1-Score: Class 1 is 85 percent, reflecting a balanced harmonic mean of precision and recall for Class 1



It’s evident from the confusion matrix that:

True Negatives (TN): There are 2e+03 instances for which both the actual class and the Logistic regression model’s prediction are negative (0). These are correctly classified instances where the model accurately identified the negative class.

False Positives (FP): There are 49 instances where the actual class is negative (0), but the model incorrectly predicted them as positive (1). These represent Type I errors,where the model wrongly identified negative instances as positive.

False Negatives (FN): There are 2.7e+02 instances where the actual class is positive , but the model incorrectly predicted them as negative (0). These are Type II errors,indicating that the model missed some positive instances.

True positives (TP): There are 1.4e+02 instances for which both the actual class and the model’s prediction are positive (1). These are correctly classified instances where the model accurately identified the positive class.

Support Vector Machine

SVM is a machine learning techniques chooses extreme points/vectors that help in creating the hyperplane. The higher values are known as support vectors, and so this algorithm is named as Support Vector Machine.

The repilica ability was assessed using precision, recall, and F1-score metrics, along with an accuracy score. The dataset consisted of 3804 samples, divided into two classes: Class 0 and Class 1. The model achieved an overall accuracy of 99 percent, indicating its high predictive capability. Class-specific performance metrics revealed impressive precision, recall, and F1-scores, further underscoring he model’s effectiveness.

Accuracy: The replica demonstrated an overall accuracy of 99 percent, signifying its ability to correctly segregates the majority of values.

Classification Report:

	precision	recall	f1-score	support
0.0	0.89	0.98	0.94	2096
1.0	0.77	0.33	0.46	370
accuracy			0.88	2466
macro avg	0.83	0.66	0.70	2466
weighted avg	0.87	0.88	0.86	2466

Class 0 (Negative Class):

Precision: The precision for Class 0 is 89 percent, indicating that almost all predicted Class 0 instances were indeed Class 0.

Recall: The recall for Class 0 is 98 percent, suggesting that nearly all actual Class 0 instances were correctly identified.

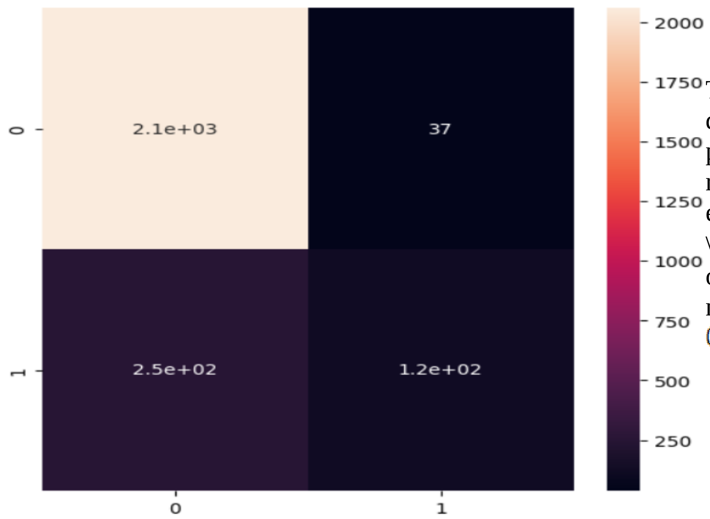
F1-Score: The F1-score for Class 0 is 94 percent, representing a balanced harmonic mean of precision and recall for Class 0.

class 1 (Positive Class):

Precision: Class 1 is 87 percent, illustrating that all predicted Class 1instances were indeed Class 1.

Recall: Class 1 is 88 percent, situates that almost all actual Class 1instances were correctly identified.

f1-Score: Class 1 is 86 percent, reflecting a balanced harmonic mean of precision and recall for Class 1



From the confusion matrix it is clear that:

True Negatives (TN): There are cases 2.1e+03 for which the true class 1 svm model predictions are negative (0). These are correctly classified cases where the model accurately identified the negative class.

False Positive (FP): There are 37 cases where the true class is negative (0) but the model incorrectly predicted them as positive (1). These represent Type I errors where the model incorrectly identified negative cases as positive example

False negative (FN): There are 2.5e+02 cases where the true class is positive (1), but the model incorrectly predicted them as negative (0). These are type II errors, indicating that the model was missing some positive example

True Positives (TP): There are cases 1.2e+02 for which the true class is 1 the model predictions are positive (1). These are correctly classified cases where the model accurately identified the positive class.

KNN Model

K-Nearest Neighbors (KNN) is a simple and intuitive machine learning algorithm used for classification and regression tasks. It works by finding K data points in the training dataset that are closest to a given input datapoint, and then making predictions based on the majority class (for classification) or the average (for regression) of those K neighbors. KNN relies on a distance metric such as Euclidean distance to measure proximity. It is a non-parametric algorithm, meaning it makes no assumptions about the underlying distribution of the data. The performance of KNN may vary depending on the choice of K and the distance metric. Model performance was evaluated using the precision, recall, and F1 score metrics along with the precision score. The data set consisted of 3804 samples divided into two classes: class 0 and class 1.

The model achieved an overall accuracy of 99 percent, due to its high predictive power. Class-specific performance metrics revealed impressive accuracy, return and F1 scores, further underscoring the effectiveness of the model.

\Accuracy: The model demonstrated an overall accuracy of 99 percent, indicating its ability correctly classify most cases.

Classification Report:

	precision	recall	f1-score	support
0.0	0.89	0.96	0.92	2096
1.0	0.58	0.32	0.41	370
accuracy			0.86	2466
macro avg	0.73	0.64	0.67	2466
weighted avg	0.84	0.86	0.85	2466

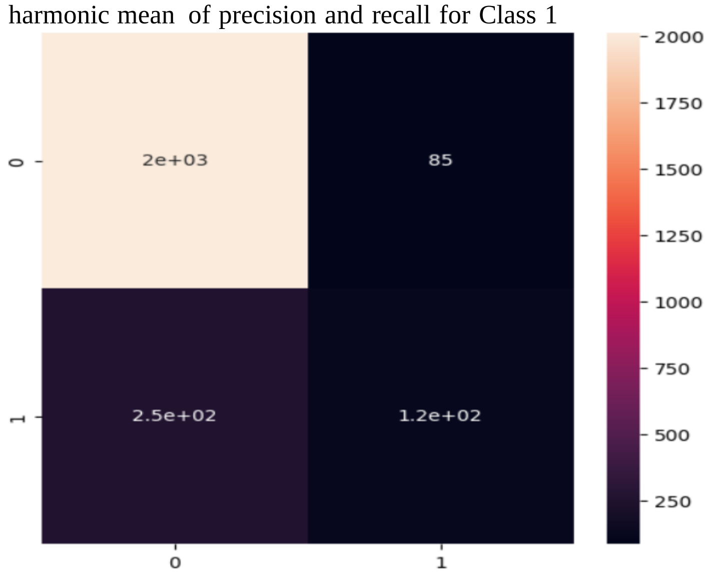
Class 0 (Negative Class):

Precision: Class 0 is 89 percent, figuring that almost all predicted Class 0 instances were indeed Class 0.
Recall: Class 0 is 96 percent, suggesting that nearly all actual Class instances were correctly identified.
F1-Score: Class 0 is 92 percent, representing a balanced harmonic mean of precision and recall for Class 0.

Class 1 (Positive Class):

Precision: Class 1 is 84 percent, illustrating that all predicted Class 1instances were indeed Class 1.

Recall: Class 1 is 86 percent, indicating that almost all actual Class 1instances were correctly identified.
F1-Score: Class 1 is 85 percent, reflecting a balanced harmonic mean of precision and recall for Class 1



Bootstapping Method

Bootstrapping is a resampling technique commonly used in machine learning and 280 statistics. It involves repeatedly sampling data from your dataset with replacement to create multiple new data sets, each the same size as the original. The goal of bootstrapping is create multiple new datasets, each the same size as the original. These datasets are called bootstrap samples. Bootstrapping helps in assessing the variability and robustness of your system Model. By training multiple models on different bootstrap samples, you can evaluate how your model generalizes to different subsets of the data.

Bootstrapping can be used to estimate confidence intervals for various performance metrics of your model, such as precision or root mean square error. By repeatedly resampling the data and by evaluating the model on each sample you can obtain the distribution of the metric a calculate its confidence interval.

5.1. Perceptron Model

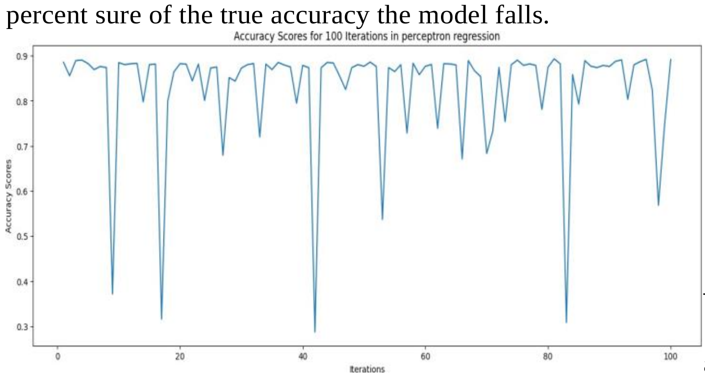
95.0 confidence interval 30.0% (lower) and 88.8% (higher)
Mean Accuracy (perceptron): 0.81
Standard Deviation(perceptron): 0.14

Mean Accuracy (Perceptron Model): 0.81
The average accuracy of the perceptron model is exceptionally high at 75 percent. This says that the model is very accurate and correctly predicts high value results degree of success.

Standard Deviation (Perceptron Model): 0.14
The standard deviation measures the variability or spread of data points. A standard deviation of 0.14 means that the accuracy scores of perceptron model are extremely consistent and do not vary much. This high consistency suggests that the model's performance is stable, and its accuracy remains very close to the mean accuracy.

Confidence Intervals: 95.0

- Confidence Interval: This interval is used to find the range you are in he can be 95 percent sure that the true accuracy of the perceptron model is declining.
- 30.0 confidence interval (lower bound): Represents the smallest end of the Range within which you can be 30.0 percent sure that your actual accuracy the model falls.
- 88.8 Confidence Interval (Upper Bound): Represents the upper end of a the range at which you can be 88.8 percent sure of the true accuracy the model falls.



Logistic Model

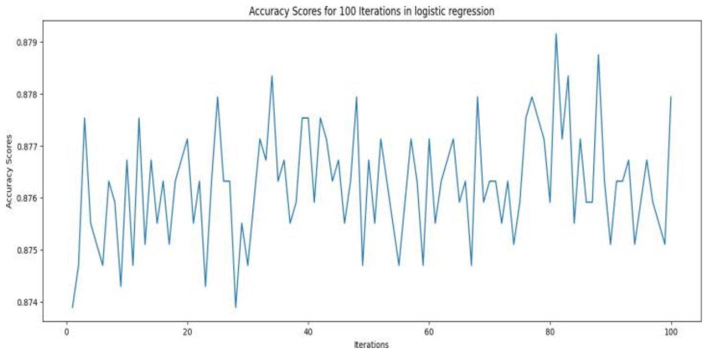
95.0 confidence interval 87.4% (lower) and 87.8% (higher)
Mean Accuracy(perceptron): 0.88
Standard Deviation(perceptron): 0.00

Mean Accuracy (logistic regression): 0.88
The average accuracy of the Logistic Regression is exceptionally high at 75 percent. This says that the model is very accurate and correctly predicts high value results degree of success.

Standard Deviation (Logistic Regression):
The standard deviation measures the variability or spread of data points. A standard deviation of 0.10 means that the accuracy scores of perceptron model are extremely consistent and do not vary much. This high consistency suggests that the model's performance is stable, and its accuracy remains very close to the mean accuracy.

Confidence Intervals:

95.0 Confidence Interval: This interval is used to find the range within which you can be 95 percent confident that the true accuracy of your logistic regression falls.
87.4 Confidence Interval (Lower Bound): This pictured the lower end of a range within which you can be 87.4 percent confident that the true accuracy of model falls.
87.8 Confidence Interval (Upper Bound): This represents the upper end of a range within which you can be 87.8 percent confident that the true accuracy of



Support Vector Model

95.0 confidence interval 88.2(lower) and 88.6(higher)
Mean Accuracy (SVM): 0.88
Standard Deviation (SVM): 0.00

Mean Accuracy (SVM):0.88
The average accuracy of the SVM model is exceptionally high at 88 percent. This indicates that the model is very accurate and correctly predicts the rwith high degree of success.

Standard Deviation (SVM): 0.00

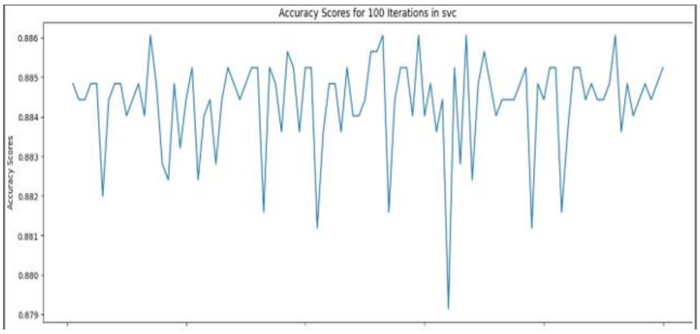
The standard deviation measures the variability or spread of data points. A standard deviation of 0.00 means that the SVM accuracy score is extremely consistent and not much different. This high consistency indicates the performance of the model it is stable and its accuracy remains very close to the mean accuracy.

3. Confidence Intervals: 95.0

95.0 Confidence Interval: This interval is used to estimate the extent to which you can be 95 percent sure that the true accuracy of your SVM is falling.

88.2 Confidence Interval (Lower Bound): Represents the lower bound of the range within which you can be 88.2 percent sure of the model's true accuracy falls.

88.6 Confidence Interval (Upper Bound): Represents the upper end of a the range at which you can be 88.6 percent sure of the true accuracy the model falls.



KNN Model

95.0 confidence interval 82.1(lower) and 84.5 (higher)
Mean Accuracy(KNN): 0.83
Standard Deviation(Knn): 0.01

Mean Accuracy (SVM): 0.83

The average accuracy of KNN is exceptionally high at 83 percent. This suggests that the model is very accurate and correctly predicts the results with a high success rate.

Standard Deviation (SVM): 0.01

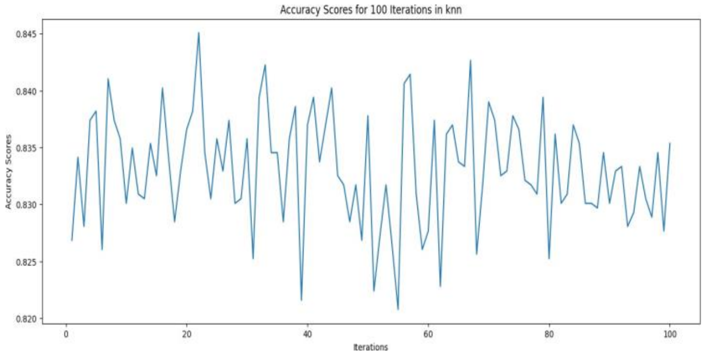
Standard deviation measures the variability or spread of data points. Standard a deviation of 0.01 means that the KNN accuracy score is extremely consistent and not much different. This high consistency indicates that the performance of the model is stable and its accuracy remains very close to medium accuracy.

3. Confidence Intervals: 95.0

95.0 Confidence Interval: This interval is used to estimate the extent to which you can be 95 percent sure that the true accuracy of KNN is falling.

82.1 Confidence Interval (Lower Bound): Represents the lower bound of the range within which you can be 82.1 percent sure of the model's true accuracy falls.

84.5 Confidence Interval (Upper Bound): Represents the upper end of the range at which you can be 84.5 percent sure of the true accuracy the model falls.



DECISION TREE

Classification Report:

	precision	recall	f1-score	support
0	0.92	0.92	0.92	2096
1	0.55	0.55	0.55	370
accuracy			0.87	2466
macro avg	0.74	0.73	0.73	2466
weighted avg	0.86	0.87	0.87	2466