

**SCTR's Pune Institute of Computer Technology
Dhankawadi, Pune**

**A PROJECT REPORT ON
“Titanic Survival Prediction using Machine Learning”**

SUBMITTED BY

41141 Kalme Akash Namdev

41124 Dhawale Harsh Vijay

**Under the guidance of
Prof. S. W. Jadhav**



**DEPARTMENT OF COMPUTER ENGINEERING
Academic Year 2023-24**



DEPARTMENT OF COMPUTER ENGINEERING

**SCTR's Pune Institute of Computer Technology
Dhankawadi, Pune, Maharashtra 411043**

CERTIFICATE

This is to certify that the SPPU Curriculum-based Mini Project titled 'Titanic Survival Prediction using Machine Learning'

Submitted by

41141 Akash Kalme

41124 Harsh Dhawale

has satisfactorily completed the curriculum-based Mini Project under the guidance of Prof. S. W. Jadhav towards the partial fulfillment of the final year of Computer Engineering Semester VII,
Academic Year 2023-24 of Savitribai Phule Pune University.

Date:

Place: PUNE

Name & Sign of Project Guide:

Acknowledgment

It gives me great pleasure to present the mini project on - Build a machine learning model that predicts the type of people who survived the Titanic shipwreck using passenger data (i.e. name, age, gender, socio-economic class, etc.).

First of all, I would like to take this opportunity to thank my guide Prof. S. W. Jadhav for giving me all the help and guidance needed. I am grateful for his kind support and valuable suggestions that proved to be beneficial in the overall completion of this project.

I am thankful to our Head of the Computer Engineering Department, Dr. G. V. Kale, for her indispensable support and suggestions throughout the internship work. I would also genuinely like to express my gratitude to the CC, Prof. Samadhan Jadhav, for his constant guidance.

Finally, I would again like to thank my mentor, Prof. S. W. Jadhav for his constant help and support during the overall process.

| Sr. No | Title | Page No. |
|---------------|-------------------|-----------------|
| 1. | Title | 5 |
| 2. | Problem Statement | 5 |
| 3. | Objectives | 5 |
| 4. | Abstract | 5 |
| 5. | Theory | 5 |
| 6. | Results | 9 |
| 7. | Conclusion | 10 |

❖ Title

Titanic Survival Prediction using Machine Learning.

❖ Problem Statement

Build a machine learning model that predicts the type of people who survived the Titanic shipwreck using passenger data (i.e. name, age, gender, socio-economic class, etc.).

❖ Objective

- To build a model for classification
- To analyze its performance on Titanic Dataset.
- To use different ML and Feature Selection concepts to optimize the model's performance.

❖ Abstract

This project is based on the [Titanic dataset](#) found on Kaggle. The sinking of the Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, the widely considered “unsinkable” Titanic sank after colliding with an iceberg. Unfortunately, there weren't enough lifeboats for everyone on board, resulting in the **death**. In this project, we see how we can use machine-learning techniques to predict survivors of the Titanic. With a dataset of 891 individuals containing features like sex, age, and class, we attempt to predict the survivors of a small test group of 418. We are using various classification models for the same.

❖ Theory:

○ Data Set:

The data we used for our project was provided on the Kaggle website. We were given 891 passenger samples for our training set and their associated labels of whether the passenger survived. For each passenger, we were given his/her passenger class, name, sex, age, number of siblings/spouses aboard, number of parents/children aboard, ticket number, fare, cabin embarked, and port of embarkation.

For the test data, we had 418 samples in the same format. The dataset is not complete, meaning that for several samples, one or many of fields were not available and marked empty (especially in the latter fields – age, fare, cabin, and port). However, all sample points contained at least information about gender and passenger class.

To normalize the data, we replace missing values with the mean value of the remaining data set or other values.

So first we will understand our [titanic dataset](#). This is a dataset of Titanic ship passengers & here each row represents the data of 1 passenger columns represent the features. We have 10 features/ variables in this dataset.

1. **Survival:** This variable shows whether the person survived or not. This is our target variable & we have to predict its value. It's a binary variable. *0 means not survived and 1 means survived.*
2. **pclass:** The ticket class of passengers. 1st (upper class), 2nd (middle), or 3rd (lower).
3. **Sex:** Gender of passenger
4. **Age:** Age (in years) of a passenger
5. **sibsp:** The no. of siblings/spouses of a particular passenger who were there on the ship.
6. **parch:** The no. of parents/children of a particular passenger who were there on the ship.
7. **ticket:** Ticket Number
8. **fare:** Passenger fare (like 1st class ticket fare must be greater than 2nd pr 3rd class ticket right)
9. **cabin:** Cabin Number
10. **embarked:** Port of Embarkation; From where that passenger took the ship. (C = Cherbourg, Q = Queenstown, S = Southampton)

o Machine Learning Models

1. Support Vector Machines (SVM):

- SVM is a supervised learning algorithm used for classification and regression tasks.
- It finds a hyperplane that best separates data points in a high-dimensional space, maximizing the margin between classes.
- SVM can handle linear and non-linear data by using various kernel functions.

2. K-Nearest Neighbors (K-NN):

- K-NN is a simple and intuitive classification algorithm.
- It classifies data points based on the majority class among their k-nearest neighbors in the feature space.
- The value of k determines the number of neighbors to consider.

3. Logistic Regression:

- Despite its name, logistic regression is a classification algorithm.
- It models the probability of a binary outcome (0 or 1) as a function of one or more independent variables using the logistic function (sigmoid function).

4. Random Forest Classifier:

- Random Forest is an ensemble learning method.
- It builds multiple decision trees and combines their predictions to make more accurate and robust classifications.
- It reduces overfitting and increases accuracy by introducing randomness in the tree-building process.

5. Naive Bayes Classifier:

- Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem.
- It assumes that features are independent, which is often an oversimplification (hence "naive").
- It's fast, especially for text classification tasks, and works well when independence assumptions are met.

6. Perceptron:

- The perceptron is a basic neural network model.
- It's a linear binary classifier that can learn from data by adjusting its weights and bias.
- Perceptrons are building blocks for more complex neural network architectures.

7. Linear SVC (Support Vector Classifier):

- Linear SVC is a variant of the SVM algorithm focused on linear classification.
- It finds the best hyperplane to separate data points into classes, but unlike SVM, it doesn't handle non-linear data without feature engineering.

8. Decision Tree Model:

- A decision tree is a tree-like model that makes decisions by splitting data based on feature values.
- It's easy to understand and interpret, and can handle both classification and regression tasks.
- Prone to overfitting, but this can be mitigated with techniques like pruning.

9. Stochastic Gradient Descent (SGD) Model:

- SGD is an optimization algorithm used to train a variety of machine learning models, including linear models and neural networks.
- It updates model parameters with each individual data point, making it computationally efficient for large datasets.

10. Gradient Boosting Classifier:

- Gradient Boosting is an ensemble learning technique that builds an additive model in a stage-wise manner.
- It combines multiple weak learners (typically decision trees) to create a strong predictive model.
- Popular implementations include XGBoost, LightGBM, and AdaBoost.

○ Evaluation Metrics

- **Accuracy**

Accuracy simply measures how often the classifier correctly predicts. We can define accuracy as the ratio of the number of correct predictions and the total number of predictions.

$$accuracy = \frac{true\ positives + true\ negatives}{true\ positives + true\ negatives + false\ positives + false\ negatives}$$

- **Confusion Matrix**


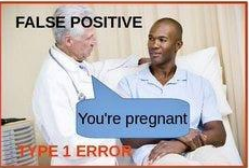


Confusion Matrix is a performance measurement for the machine learning classification problems where the output can be two or more classes. It is a table with combinations of predicted and actual values.

A confusion matrix is defined as the table that is often used to describe the performance of a classification model on a set of the test data for which the true values are known.

| | | Actual Values | |
|------------------|--------------|---------------|--------------|
| | | Positive (1) | Negative (0) |
| Predicted Values | Positive (1) | TP | FP |
| | Negative (0) | FN | TN |

It is extremely useful for measuring the Recall, Precision, Accuracy, and AUC-ROC curves.

Let's try to understand TP, FP, FN, TN with an example of pregnancy analogy.

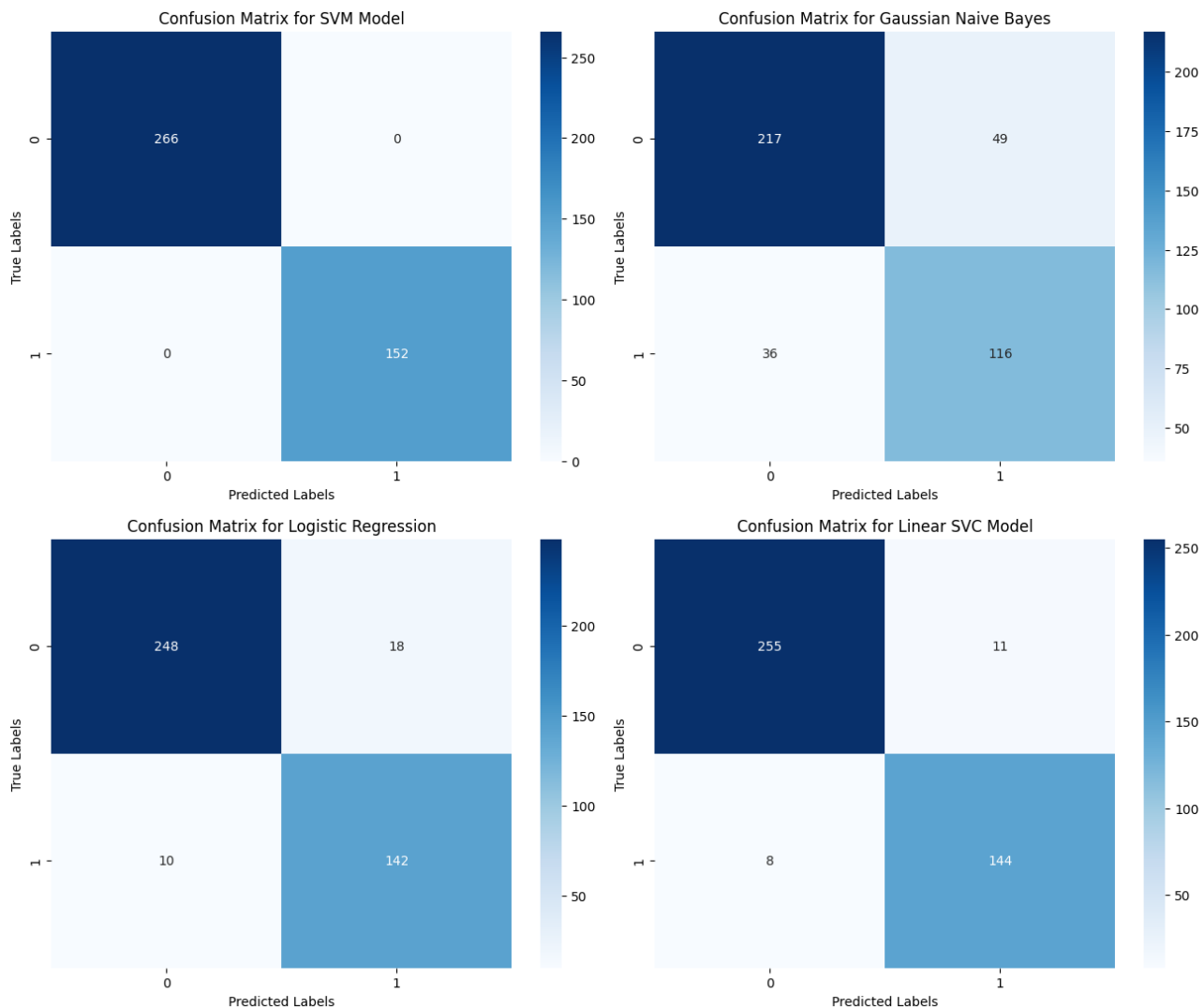
| | | $\hat{Y} = 0$ NEGATIVE | $\hat{Y} = 1$ POSITIVE |
|-------------------------|---------------------|---|--|
| $Y = 0$ NOT PREGNANT | TRUE NEGATIVE |  | FALSE POSITIVE TYPE 1 ERROR  |
| | $Y = 1$ PREGNANT | FALSE NEGATIVE TYPE 2 ERROR  | TRUE POSITIVE  |

1. True Positive: We predicted positive and it's true. In the image, we predicted

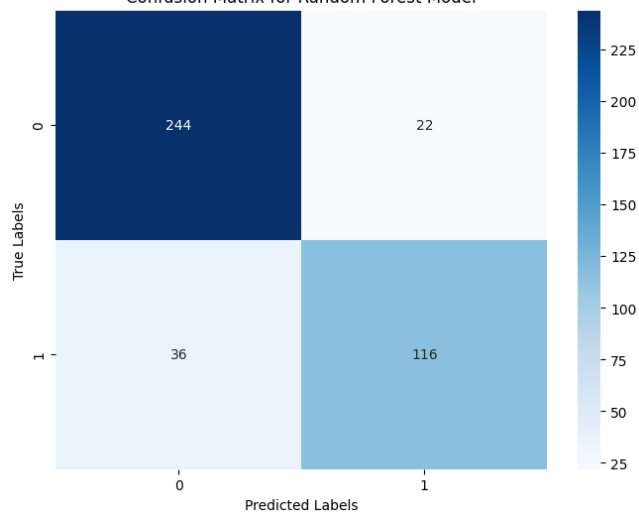
that a woman is pregnant, and she actually is.

2. **True Negative:** We predicted negative and it's true. In the image, we predicted that a man is not pregnant, and he actually is not.
3. **False Positive (Type 1 Error):** We predicted positive and it's false. In the image, we predicted that a man is pregnant, but he actually is not.
4. **False Negative (Type 2 Error):** We predicted negative and it's false. In the image, we predicted that a woman is not pregnant, but she actually is.

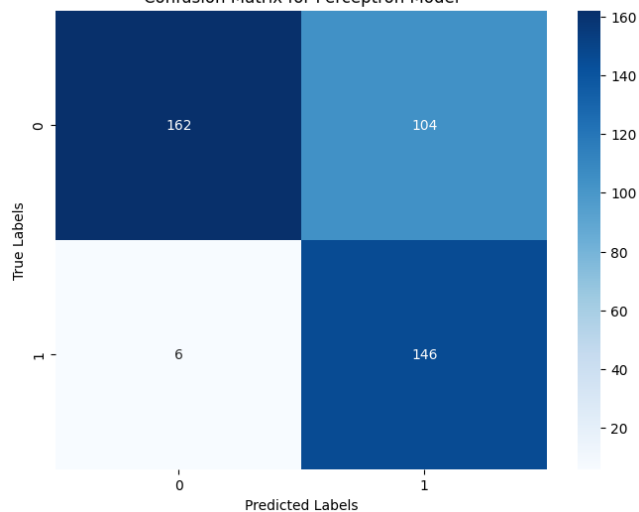
❖ Results



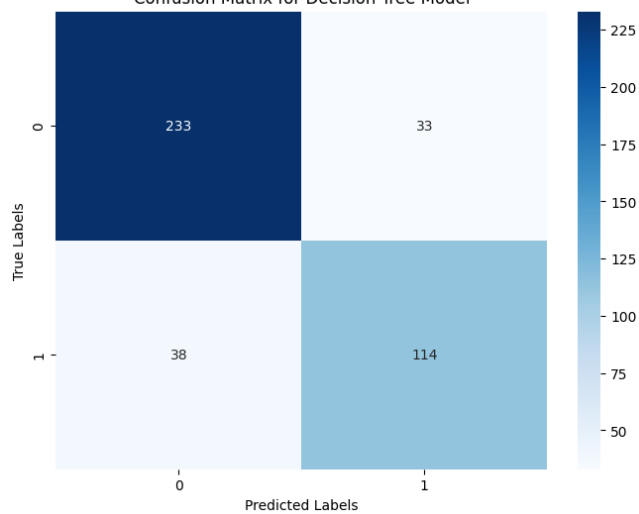
Confusion Matrix for Random Forest Model



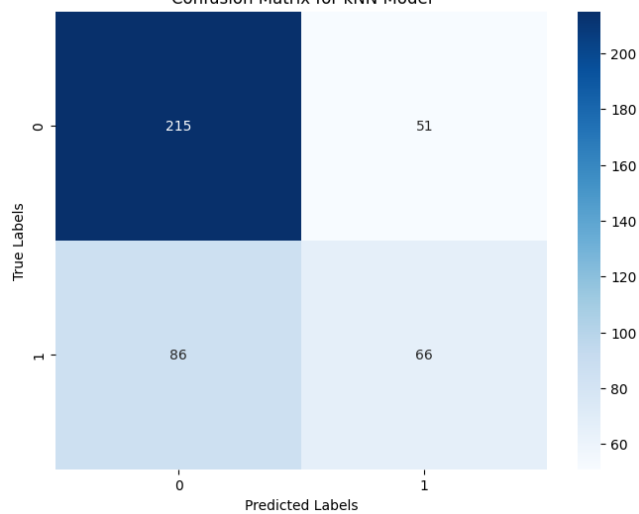
Confusion Matrix for Perceptron Model



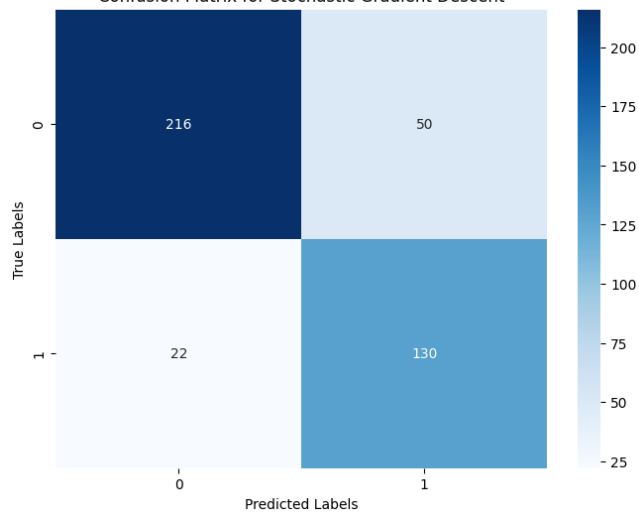
Confusion Matrix for Decision Tree Model



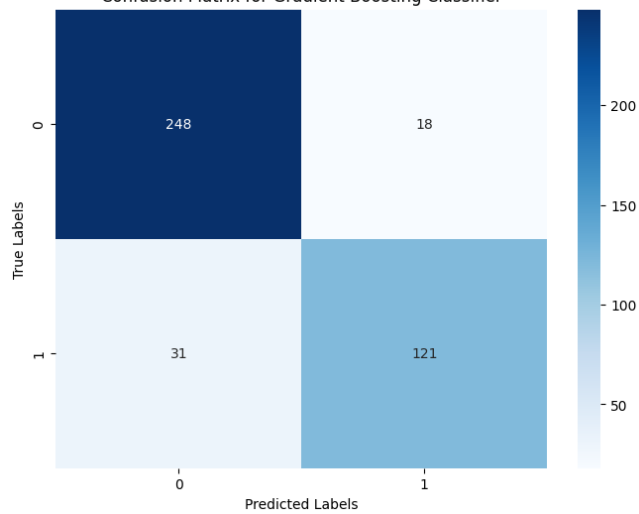
Confusion Matrix for kNN Model



Confusion Matrix for Stochastic Gradient Descent



Confusion Matrix for Gradient Boosting Classifier



| | Model | Score |
|---|-----------------------------------|----------|
| 0 | Support Vector Machines | 1.000000 |
| 6 | Linear SVC | 0.954545 |
| 2 | Logistic Regression | 0.933014 |
| 9 | Gradient Boosting Classifier | 0.882775 |
| 3 | Random Forest Classifier | 0.856459 |
| 7 | Decision Tree Model | 0.825359 |
| 8 | Stochastic Gradient Descent Model | 0.820574 |
| 4 | Naive Bayes Classifier | 0.796651 |
| 5 | Perceptron | 0.736842 |
| 1 | K-Nearest Neighbours | 0.672249 |

❖ Conclusion

The analysis revealed interesting patterns across individual-level features. Factors such as socioeconomic status, social norms and family composition appeared to have an impact on likelihood of survival. These conclusions, however, were derived from findings in the given data set.

It has been observed that female survival rates are very high (approx. 74%) while male survival rates are very low. To make predictions in classification problems, the technique of logistic regression is primarily used.

It would be interesting to play more with the dataset and introduce more attributes which might lead to better results. Various other machine learning techniques like Naive Bayes, K-NN classification can be used to solve the problem.