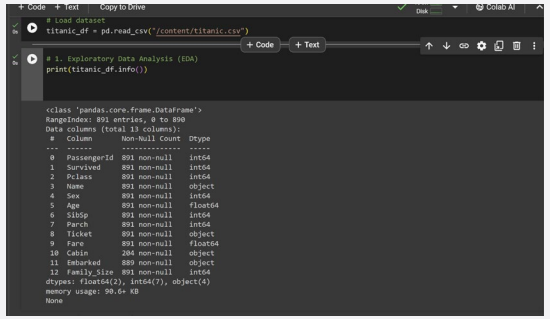
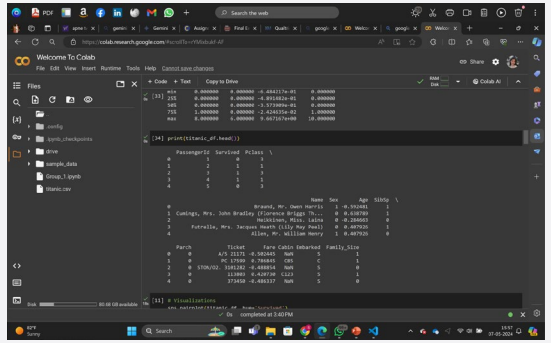
GROUP-1 TITANIC DATASET

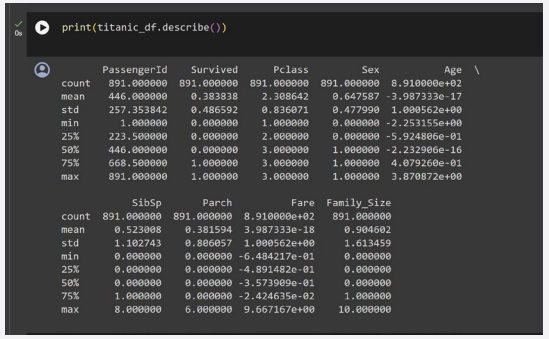
NAME:
KHUSHBOO
NAVYA
HARSHA

PROBLEM STATEMENT:

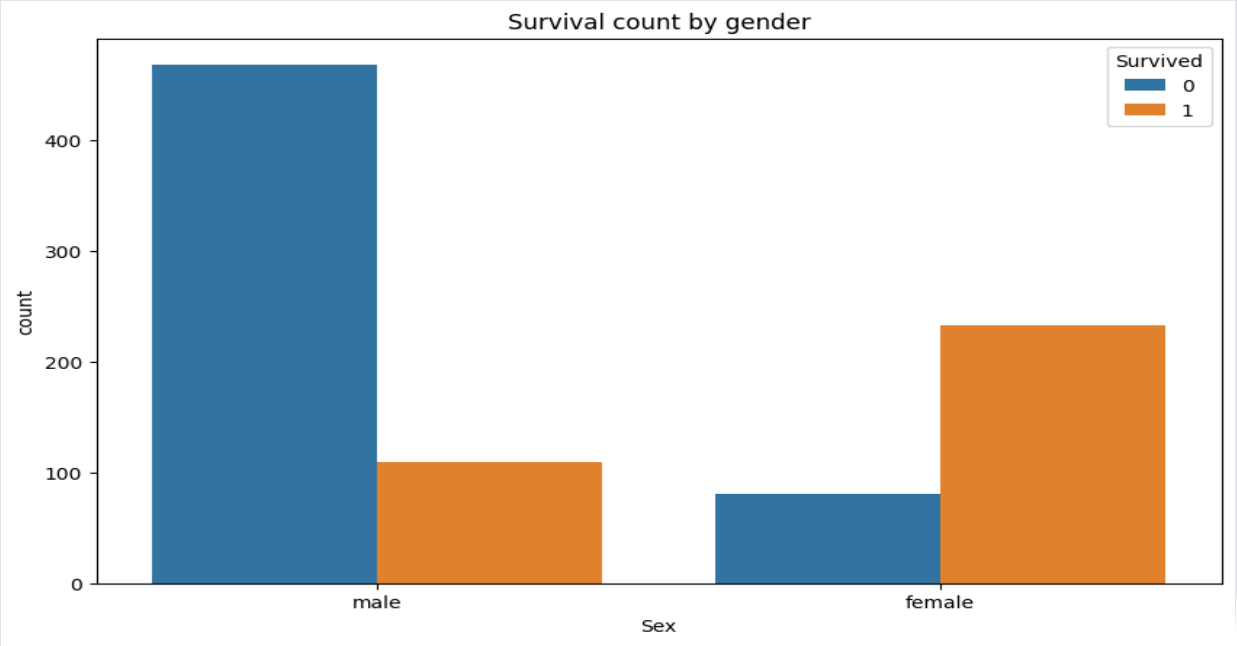
TO FIND SURVIVAL STATUS OF PASSENGER ON THE TITANIC.

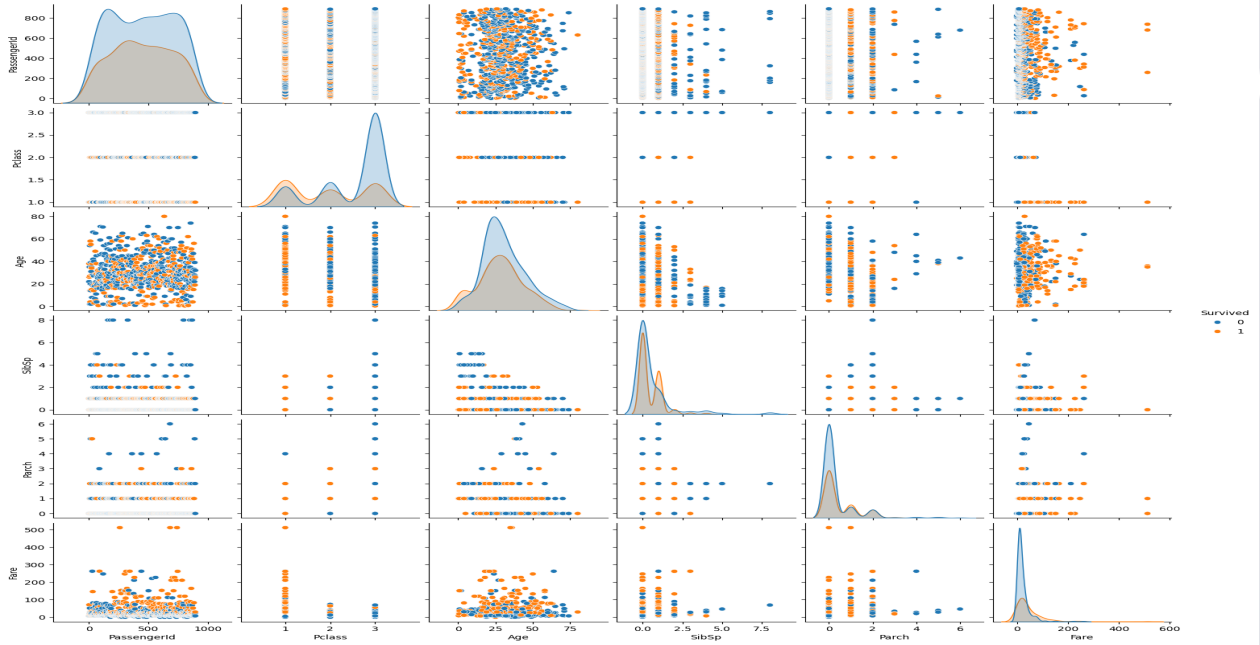






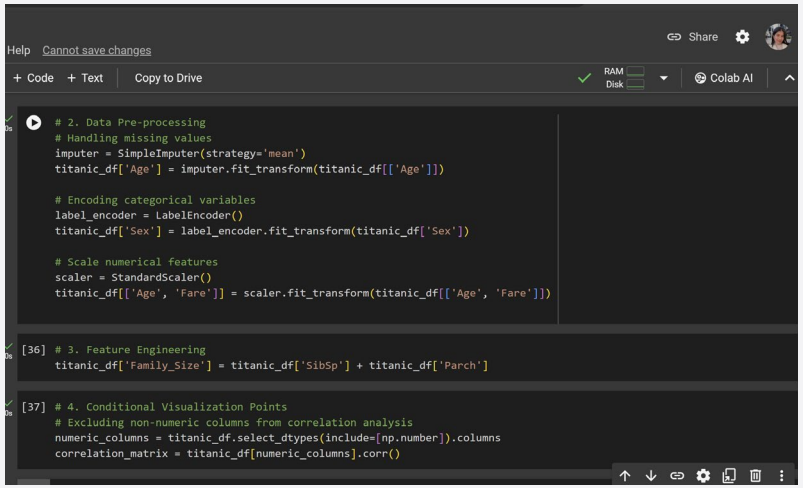
PRINTING HEAD, DESCRIPTION AND INFORMATION OF TITANIC DATASET





PAIRPLOT AND COUNTPLOT :-

- HANDLING MISSING VALUES, ENCODING CATEGORICAL VARIABLES AND SCALE NUMERICAL FEATURES:-
- FEATURE ENGINEERING:
- CONDITIONAL VISUALIZATION POINTS:-



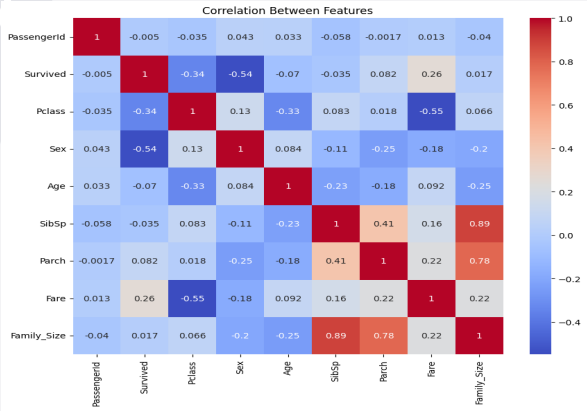
HEATMAP ,BOXPLOT AND COUNTPLOTS:

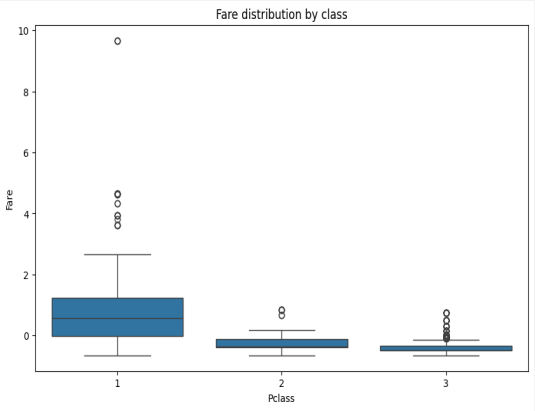
PLOTTING THE HEATMAP WITH CORRELATION BETWEEN FEATURES.

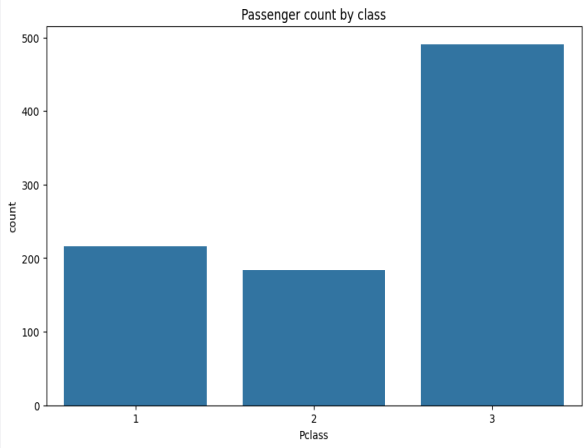
PLOTTING BOXPLOT FOR FARE DISTRIBUTION BY CLASS.

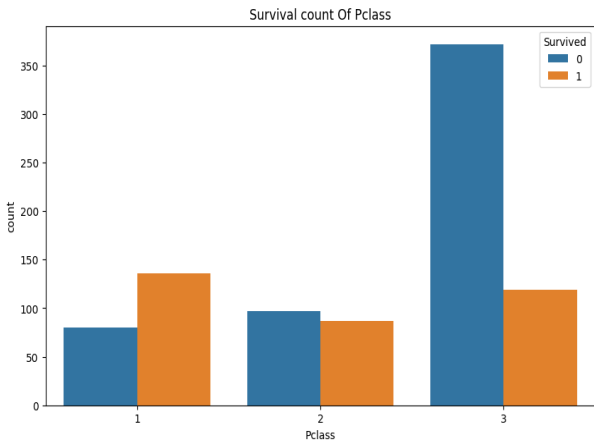
PLOTTING COUNTPLOT FOR PASSENGER COUNT BY CLASS.

PLOTTING COUNTPLOT OF SURVIVAL COUNT OF PCLASS.









```
[25] # Count of Non-Surviving Passengers Between Ages 20-30
     young_non_survivors = titanic_df[(titanic_df['Age'] >= 20) & (titanic_df['Age'] <= 30) & (titanic_df['Survived'] == 0
     print("Count of non-surviving passengers between ages 20-30:", young_non_survivors)

     Count of non-surviving passengers between ages 20-30: 0
```

Help Cannot save changes

Share

+ Code + Text Copy to Drive

RAM Disk

Colab AI

```
[39] # 5. Models for Analysis
     X = titanic_df.drop(['Survived', 'Name', 'Ticket', 'Cabin', 'Embarked'], axis=1)
     y = titanic_df['Survived']

[40] # Spliting data into train and test sets
     X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

[41] # Initializing models
     models = {
         "Logistic Regression": LogisticRegression(),
         "K Nearest Neighbors": KNeighborsClassifier(),
         "Support Vector Classifier": SVC(),
         "Decision Tree": DecisionTreeClassifier(),
         "Random Forest": RandomForestClassifier()
     }
```

COUNTING, ANALYZING, SPLITTING AND INITIALIZING.

INITIALIZING, SCALING, TRAINING AND EVALUATING MODELS

```
[31] # Initializing logistic regression model with increased max_iter
     logistic_model = LogisticRegression(max_iter=1000)

     # Scaling the numerical features
     scaler = StandardScaler()
     X_train_scaled = scaler.fit_transform(X_train)
     X_test_scaled = scaler.transform(X_test)

     # Training the model on scaled data
     logistic_model.fit(X_train_scaled, y_train)

     # Evaluating the model
     y_pred = logistic_model.predict(X_test_scaled)
     accuracy = accuracy_score(y_test, y_pred)
     print(f"Accuracy of Logistic Regression: {accuracy}")

     Accuracy of Logistic Regression: 0.8100558659217877
```

Help Cannot save changes

Share

+ Code + Text Copy to Drive

RAM Disk Colab AI

```
[41] }

# Training & evaluating models
for name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    print(f"Accuracy of {name}: {accuracy}")

/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWarning: lbfgs failed to con
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
Accuracy of Logistic Regression: 0.7988826815642458
Accuracy of K Nearest Neighbors: 0.5642458100558659
Accuracy of Support Vector Classifier: 0.5865921787709497
Accuracy of Decision Tree: 0.7486033519553073
Accuracy of Random Forest: 0.8212290502793296

[31] # Initializing logistic regression model with increased max_iter
```

CONCLUSION:-

Here by survival by Gender we have anlayse that Females survival Count is more than Male.

Non-Survival count for Class 3 is greater than class 1 and class 2.

Overall, a box plot analysis on fare by passenger class in the Titanic dataset offers a visual summary of fare distributions within each class and facilitates comparisons between classes, enabling deeper insights into the socioeconomic dynamics among Titanic passengers.

The count of passengers are way higher in 3rd class followed by 1st class and 2nd class.

Count of non-surviving passengers between ages 20-30: 0Random Forest has good accuracy compared to other algorithms in this project.

THANK YOU