In general, we did not perform any feature engineering on the modified Titanic dataset, which may have improved the following results in the questions below. In **Appendix 1**, we noticed that the distribution of column Survived (the label that we want to predict) is imbalanced. We will deal with this in the questions below.

There are some notable correlations from **Appendix 2**, such as:
- Fare and Pclass with −0.55
- Survived and Sex_male (the encoded Sex column) with −0.54
- Pclass and Age with −0.41
- Parch and SibSp with 0.41

We could interpret them as such:
- Higher-class passengers generally paid more, which makes sense historically.
- Females were more likely to survive, which is consistent with the "women and children first" evacuation practice.
- Younger passengers were somewhat more likely to be in higher classes.
- Passengers with parents/children onboard also tended to have siblings/spouses, reflecting family groups traveling together.

Report and discuss the performance of the Logistic Regression model on both the train and test sets (separately). You can use scikit-learn's classification report tool.

- See **Appendix 3** for the results with the non-resampled dataset and class_weight="balanced".
- See **Appendix 4** for the results with the resampled dataset.

The Logistic Regression classifier was developed on a resampled version of the Titanic dataset to address class imbalance and improve recall for the minority class (survivors). Hyperparameters were optimized using both grid search and random search, and performance was evaluated on training and test sets separately.

The grid search–tuned model achieved an accuracy of 78% on the training set and 80% on the test set, indicating strong generalization. Performance across both classes was different, with precision and recall hovering around 0.74-0.85. We noticed that class 0 (non-survived) usually has the higher precision and recall values. On the test set, the model performed slightly better at identifying non-survivors (precision 0.83, recall 0.85) than survivors (precision 0.77, recall 0.74). The training confusion matrix showed 70 false positives (predicted survivors who did not survive) and 82 false negatives (missed survivors). The test confusion matrix showed 16 false positives and 19 false negatives.

The random search-tuned model produced identical results to the grid search model on both training and test sets, with test accuracy at 80%. The precision–recall trade-off mirrored that of the grid search model, favoring non-survivors slightly while still achieving balanced performance compared to models trained on the original imbalanced dataset. The confusion matrices were identical as well, confirming no practical difference in tuning strategy for this classifier.

Both hyperparameter search strategies produced the same logistic regression model, suggesting the parameter space was relatively stable and easy to optimize. Compared with models trained on the original imbalanced dataset, resampling improved recall for survivors, reducing bias toward the majority class. However, the model still misclassifies some survivors (74% recall on test data), reflecting the inherent difficulty of the task.

Report and discuss the performance of the Perceptron model on both the train and test sets (separately). You can use scikit-learn's classification report tool.

- See **Appendix 5** for the results with the non-resampled dataset and class_weight="balanced".
- See **Appendix 6** for the results with the resampled dataset.

The Perceptron classifier was trained and evaluated on the Titanic dataset using both grid search and random search hyperparameter tuning. Performance was assessed on the training and test sets separately, using classification reports and confusion matrices.

In training the Perceptron classifier, the parameter class_weight="balanced" was applied. This setting automatically adjusts the importance of each class inversely proportional to its frequency in the training data. Since the Titanic dataset is imbalanced (fewer survivors than non-survivors), this weighting penalizes mistakes on survivors more heavily than mistakes on non-survivors.

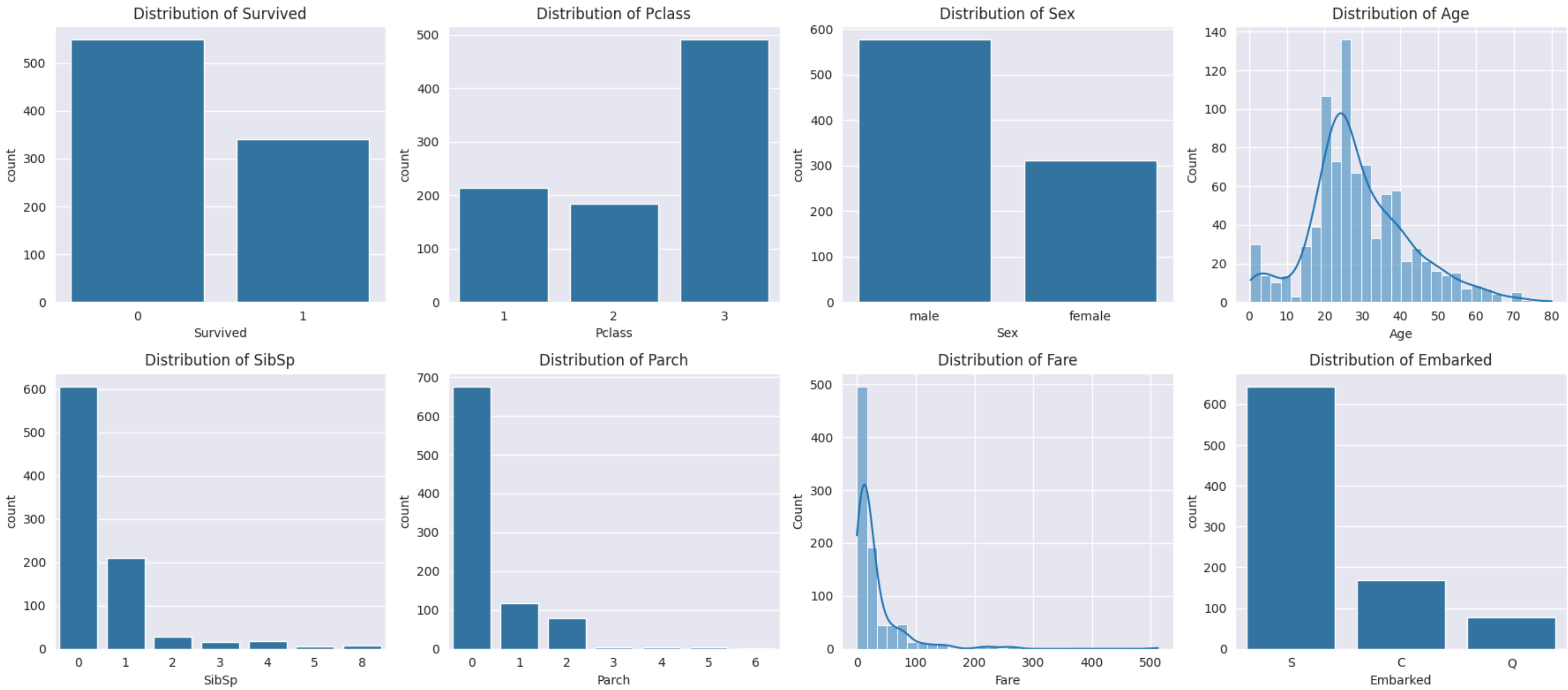
The grid search–tuned model achieved an accuracy of 76% on the training set and 80% on the test set, showing no evidence of overfitting. For the majority class (non-survivors), the model demonstrated high recall values (0.78-0.85), indicating it was effective at correctly identifying passengers who did not survive. For the minority class (survivors), precision (0.68-0.76) varied more than recall (0.72-0.73). This indicates that while the model consistently identified a similar fraction of true survivors, the reliability of its "survived" predictions fluctuated, with some non-survivors being misclassified as survivors. The training confusion matrix has 76 false negatives and 60 false positives, and the test confusion matrix has 16 false negatives and 20 false positives.

The random search–tuned model achieved 77% accuracy on the training set and 83% on the test set, which shows good generalization. Compared to the grid search model, it produced slightly higher recall for survivors (0.76), while still having that unstable precision (0.68-0.81). The confusion matrices for this model are a bit better than the grid-search tuned model, except the fact that there is 1 more false negative on the training set.

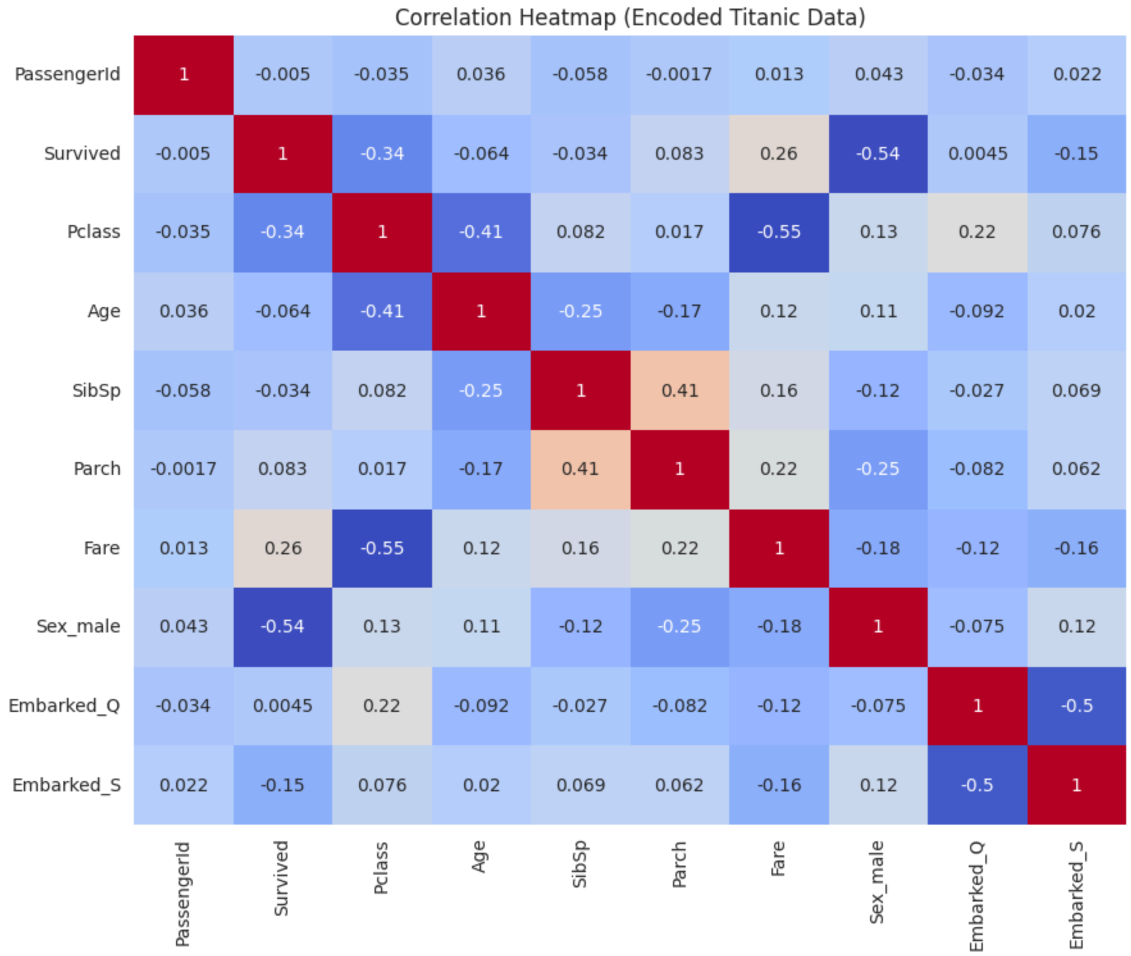
However, when the dataset was oversampled using SMOTE, the Perceptron classifier performed noticeably worse. The Perceptron is a simple linear classifier that relies on finding a single separating hyperplane, and it struggles when the data contains overlapping or artificially smoothed regions. As a result, the model overfit to the synthetic patterns in the oversampled training set, but failed to generalize to the real distribution in the test set. This explains the inconsistent/poor accuracy, precision, and recall observed in **Appendix 6** compared to the non-resampled dataset.

Given the context of the Titanic survival prediction, prioritizing recall is more important. It is preferable to flag more potential survivors, even if this includes some false positives, rather than miss actual survivors. Therefore, the random search model is more suitable to maximize survivor identification.

Appendix 1: Distribution of features in the Titanic dataset



Appendix 2: Correlation Heatmap



Appendix 3: Logistic Regression results on non-resampled dataset, with class_weight="balanced"

Keep the recall values in mind.

```
=== GRID - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.80      0.88      0.84       347
           1       0.78      0.66      0.71       221

    accuracy                           0.79       568
   macro avg       0.79      0.77      0.77       568
weighted avg       0.79      0.79      0.79       568

=== GRID - Training Confusion Matrix ===
[[305  42]
 [ 76 145]]
=== GRID - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.79      0.91      0.85       106
           1       0.82      0.65      0.73        72

    accuracy                           0.80       178
   macro avg       0.81      0.78      0.79       178
weighted avg       0.81      0.80      0.80       178

=== GRID - Test Confusion Matrix ===
[[96 10]
 [25 47]]
```

```
=== RANDOM - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.80      0.88      0.84       347
           1       0.77      0.66      0.71       221

    accuracy                           0.79       568
   macro avg       0.79      0.77      0.77       568
weighted avg       0.79      0.79      0.79       568

=== RANDOM - Training Confusion Matrix ===
[[304  43]
 [ 76 145]]
=== RANDOM - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.79      0.89      0.84       106
           1       0.80      0.65      0.72        72

    accuracy                           0.79       178
   macro avg       0.79      0.77      0.78       178
weighted avg       0.79      0.79      0.79       178

=== RANDOM - Test Confusion Matrix ===
[[94 12]
 [25 47]]
```

Appendix 4: Logistic Regression results on SMOTE resampled dataset

This one has noticeably higher recall values than Appendix 3.

```
=== GRID - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.77      0.80      0.78       347
           1       0.79      0.76      0.78       347

    accuracy                           0.78       694
   macro avg       0.78      0.78      0.78       694
weighted avg       0.78      0.78      0.78       694

=== GRID - Training Confusion Matrix ===
[[277  70]
 [ 82 265]]
=== GRID - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.83      0.85      0.84       106
           1       0.77      0.74      0.75        72

    accuracy                           0.80       178
   macro avg       0.80      0.79      0.79       178
weighted avg       0.80      0.80      0.80       178

=== GRID - Test Confusion Matrix ===
[[90 16]
 [19 53]]
```

```
=== RANDOM - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.77      0.80      0.78       347
           1       0.79      0.76      0.78       347

    accuracy                           0.78       694
   macro avg       0.78      0.78      0.78       694
weighted avg       0.78      0.78      0.78       694

=== RANDOM - Training Confusion Matrix ===
[[277  70]
 [ 82 265]]
=== RANDOM - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.83      0.85      0.84       106
           1       0.77      0.74      0.75        72

    accuracy                           0.80       178
   macro avg       0.80      0.79      0.79       178
weighted avg       0.80      0.80      0.80       178

=== RANDOM - Test Confusion Matrix ===
[[90 16]
 [19 53]]
```

Appendix 5: Perceptron results on non-resampled dataset, with class_weight="balanced"

```
=== GRID - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.82      0.78      0.80       347
           1       0.68      0.73      0.70       221

    accuracy                           0.76       568
   macro avg       0.75      0.75      0.75       568
weighted avg       0.76      0.76      0.76       568

=== GRID - Training Confusion Matrix ===
[[271  76]
 [ 60 161]]
=== GRID - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.82      0.85      0.83       106
           1       0.76      0.72      0.74        72

    accuracy                           0.80       178
   macro avg       0.79      0.79      0.79       178
weighted avg       0.80      0.80      0.80       178

=== GRID - Test Confusion Matrix ===
[[90 16]
 [20 52]]
```

```
=== RANDOM - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.83      0.78      0.80       347
           1       0.68      0.76      0.72       221

    accuracy                           0.77       568
   macro avg       0.76      0.77      0.76       568
weighted avg       0.78      0.77      0.77       568

=== RANDOM - Training Confusion Matrix ===
[[270  77]
 [ 54 167]]
=== RANDOM - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.85      0.88      0.86       106
           1       0.81      0.76      0.79        72

    accuracy                           0.83       178
   macro avg       0.83      0.82      0.82       178
weighted avg       0.83      0.83      0.83       178

=== RANDOM - Test Confusion Matrix ===
[[93 13]
 [17 55]]
```

Appendix 6: Perceptron results on SMOTE resampled dataset

Notice the inconsistency between grid/random results. This may be an indication that overfitting is happening.

```
=== GRID - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.71      0.90      0.80       347
           1       0.86      0.64      0.74       347

    accuracy                           0.77       694
   macro avg       0.79      0.77      0.77       694
weighted avg       0.79      0.77      0.77       694

=== GRID - Training Confusion Matrix ===
[[311  36]
 [124 223]]
=== GRID - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.73      0.92      0.82       106
           1       0.82      0.50      0.62        72

    accuracy                           0.75       178
   macro avg       0.77      0.71      0.72       178
weighted avg       0.77      0.75      0.74       178

=== GRID - Test Confusion Matrix ===
[[98  8]
 [36 36]]
```

```
=== RANDOM - Training Set Performance ===
              precision    recall  f1-score   support

           0       0.60      0.44      0.51       347
           1       0.56      0.71      0.62       347

    accuracy                           0.57       694
   macro avg       0.58      0.57      0.56       694
weighted avg       0.58      0.57      0.56       694

=== RANDOM - Training Confusion Matrix ===
[[152 195]
 [102 245]]
=== RANDOM - Test Set Performance ===
              precision    recall  f1-score   support

           0       0.74      0.37      0.49       106
           1       0.46      0.81      0.59        72

    accuracy                           0.54       178
   macro avg       0.60      0.59      0.54       178
weighted avg       0.63      0.54      0.53       178

=== RANDOM - Test Confusion Matrix ===
[[39 67]
 [14 58]]
```