

# Deep Learning HW 5

Baseline Score: 0.77511

The screenshot shows the Kaggle website interface for the Titanic competition. On the left, a sidebar lists several submissions, each with a green checkmark indicating it is complete. The submissions are: 'optimized\_rf\_submission.csv' (1h ago, opt rf), 'optimized\_gb\_submission.csv' (2h ago), 'xgb\_submission.csv' (2h ago, xgb), 'ml\_submission\_.csv' (2h ago), 'ml\_submission\_nn.csv' (2h ago, neural network), 'ml\_submission.csv' (2h ago, 3 models), and 'baseline\_submission.csv' (3h ago, probability approach). The main panel displays the details for the 'baseline\_submission.csv' submission, showing a score of 0.77511 and a completion time of 3h ago. It also shows the uploaded file 'baseline\_submission.csv' (2 KIB) and a description box containing the text 'probability approach'. At the bottom right, there are 'Cancel' and 'Save' buttons.

Search

## Titanic - Machine Learning from D

Overview Data Code Models Discussion

- ✓ **optimized\_rf\_submission.csv**  
Complete · 1h ago · opt rf
- ✓ **optimized\_gb\_submission.csv**  
Complete · 2h ago
- ✓ **xgb\_submission.csv**  
Complete · 2h ago · xgb
- ✓ **ml\_submission\_.csv**  
Complete · 2h ago
- ✓ **ml\_submission\_nn.csv**  
Complete · 2h ago · neural network
- ✓ **ml\_submission.csv**  
Complete · 2h ago · 3 models
- ✓ **baseline\_submission.csv**  
Complete · 3h ago · probability approach

### Submission Details

✓ **baseline\_submission.csv** **Score: 0.77511**  
Complete · 3h ago

UPLOADED FILES

baseline\_submission.csv (2 KIB)

DESCRIPTION

probability approach

20 / 500

Cancel Save

Baseline performance evaluation results:-

BASELINE PERFORMANCE ON TRAINING DATA

Accuracy: 0.7868

Confusion Matrix:

```
[[540   9]
 [181 161]]
```

F1 Score: 0.6289

Prediction breakdown:

Predicted

0 721

1 170

Name: count, dtype: int64

Actual survival breakdown:

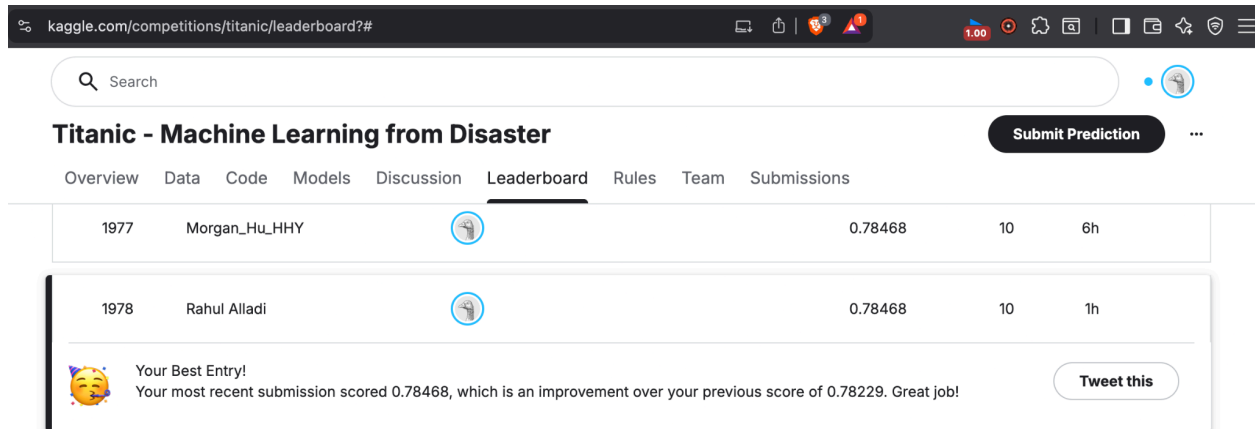
Survived

0 549

1 342

Name: count, dtype: int64

## Machine Learning Approach Score: 0.78468



The screenshot shows the Kaggle Titanic competition leaderboard. The top navigation bar includes a search bar, a 'Submit Prediction' button, and a menu icon. The main navigation tabs are Overview, Data, Code, Models, Discussion, Leaderboard (selected), Rules, Team, and Submissions. The leaderboard table lists two entries: entry 1977 by Morgan\_Hu\_HHY and entry 1978 by Rahul Alladi, both with a score of 0.78468. Entry 1978 is highlighted as the 'Your Best Entry!' with a message: 'Your most recent submission scored 0.78468, which is an improvement over your previous score of 0.78229. Great job!' and a 'Tweet this' button.

Rank	Participant	Score	Attempts	Time
1977	Morgan_Hu_HHY	0.78468	10	6h
1978	Rahul Alladi	0.78468	10	1h

Your Best Entry!  
Your most recent submission scored 0.78468, which is an improvement over your previous score of 0.78229. Great job!

Tweet this

### 1. What data preprocessing methods did you apply, such as feature scaling with a min-max method or one-hot encoding for a feature?

For data preprocessing, I applied median imputation to fill missing values in the Age and Fare columns, and mode imputation for missing Embarked values. I created engineered features including FamilySize (sum of SibSp and Parch) and IsAlone (binary indicator for solo passengers). For feature transformation, I used StandardScaler to normalize numerical features (Age, Fare, FamilySize, SibSp, Parch) and OneHotEncoder to convert categorical features (Pclass, Sex, Embarked, IsAlone) into binary dummy variables. Irrelevant features such as PassengerId, Name, Ticket, and Cabin were dropped from the dataset.

### 2. What machine learning model(s) did you apply, such as SVM or random forest?

I applied a Support Vector Machine (SVM) classifier with GridSearchCV for hyperparameter optimization. The grid search explored different kernel types (rbf and poly), regularization parameter C values (0.5, 1, 2, 5), and gamma values (0.05, 0.1, 0.2, 0.5). The model was trained using 5-fold stratified cross-validation to ensure robust performance evaluation and prevent overfitting, with the entire preprocessing and classification process integrated into a scikit-learn Pipeline.

### 3. What are performance evaluation results on training dataset, including accuracy, confusion matrix, and F1 score, on the training dataset?

On the training dataset, the best model achieved an accuracy of 0.8328 (83.28%), meaning it correctly classified 83.28% of all passengers. The F1 score was 0.7704, indicating a good balance between precision and recall for predicting survivors. The confusion matrix showed  $\begin{bmatrix} 492 & 57 \\ 92 & 250 \end{bmatrix}$ , which

means the model correctly predicted 492 non-survivors (true negatives), 250 survivors (true positives), while incorrectly predicting 57 non-survivors as survivors (false positives) and 92 survivors as non-survivors (false negatives).

#### **4.What the best score and ranking at Leaderboard at Kaggle did you receive**

On the Kaggle competition leaderboard, my submission received a score of 0.78468 (78.468% accuracy on the test dataset) and ranked 1978th among all participants. This score is lower than the training accuracy, suggesting some degree of overfitting to the training data.