# 1. Background: general information about the ADS

## a. What is the purpose of this ADS? What are its stated goals?

The purpose of Akshay Pawar’s ADS is to acquire stable results on predicting Titanic disaster outcome (which passengers would survive). Pawar’s goal is to tune classification models based on ROC and AUC scores to achieve the best stability/accuracy and choose the most consistent predictions under a Voting Classifier. In other words, this ADS values accuracy and consistency by choosing the best attributes to predict survival and choosing the most consistent prediction among all the most accurate classification models.

## b. If the ADS has multiple goals, explain any trade-offs that these goals may introduce.

As Pawar’s ADS values accuracy and consistency with its predictions, there is a trade-off that may be observed with predictive fairness. Pawar’s ADS should be evaluated for disparate impact and/or disparate treatment of individuals based on the protected characteristics: sex, age, ticket class, etc.. We should observe if/how individuals are treated differently within such groups. We may also want to observe trade-offs regarding the privacy of passenger data because the ADS does not have any function to mitigate sensitive information that can be used to identify passengers.

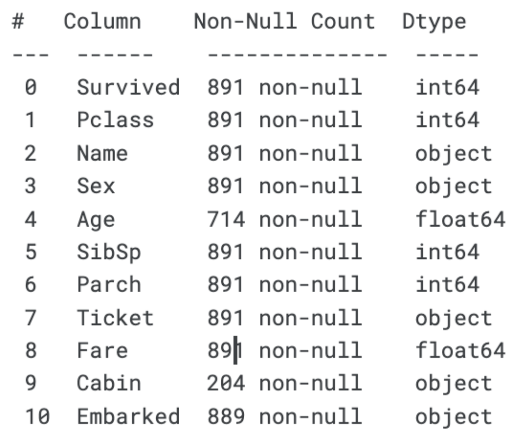
# 2. Input and Output

## a. Describe the data used by this ADS. How was this data collected or selected?

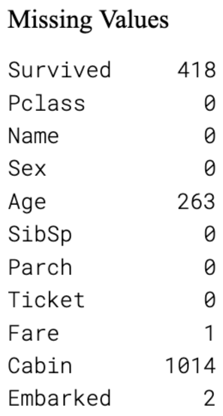
This ADS uses both categorical and numerical data — categorical features include ‘Survived’ (Outcome attribute), ‘Pclass’, ‘Sex’, etc. and numerical features include ‘Age’, and ‘Fare’. The data used in this ADS is provided by the Titanic Kaggle competition. ***The ‘train’ dataset contains exact details of a subset of the passengers to be used as the ground truth, while the ‘test’ dataset does not include the ground truth values.*** As you will see later, as we recreate Pawar's ADS, he imputes the values of "Survived" for the "test" dataframe and uses those imputed values to test his ADS.

For our analysis purposes, we decided to split the provided `train.csv` into "train" and "test" data frames so we have "true values" of surviving passengers. We will still use Pawar's imputed values for the "Age" column however as that is an attribute we wish to analyze further.

b. For each input feature, describe its datatype, give information on missing values and on the value distribution. Show pairwise correlations between features if appropriate. Run any other reasonable profiling of the input that you find interesting and appropriate.



```
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Survived  891 non-null    int64
 1   Pclass    891 non-null    int64
 2   Name      891 non-null    object
 3   Sex       891 non-null    object
 4   Age       714 non-null    float64
 5   SibSp     891 non-null    int64
 6   Parch     891 non-null    int64
 7   Ticket    891 non-null    object
 8   Fare      891 non-null    float64
 9   Cabin     204 non-null    object
 10  Embarked  889 non-null    object
```



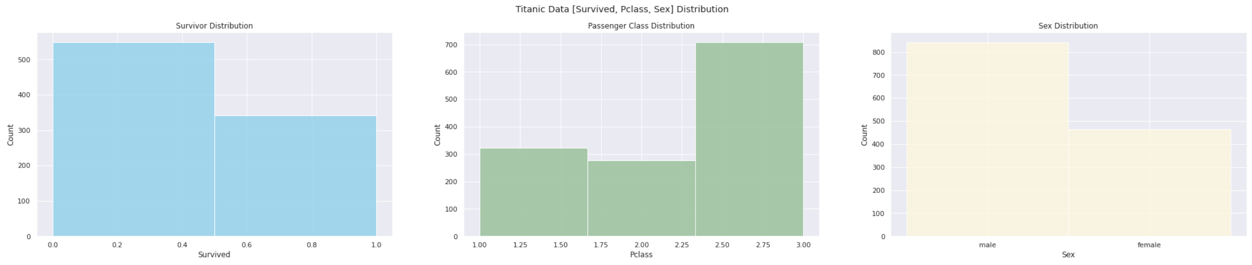
```
Missing Values

Survived     418
Pclass         0
Name           0
Sex            0
Age          263
SibSp          0
Parch          0
Ticket         0
Fare           1
Cabin       1014
Embarked       2
```

From this information, we may need to recode values for fairness analysis, namely "Sex", "Age", "Fare" and "PClass"

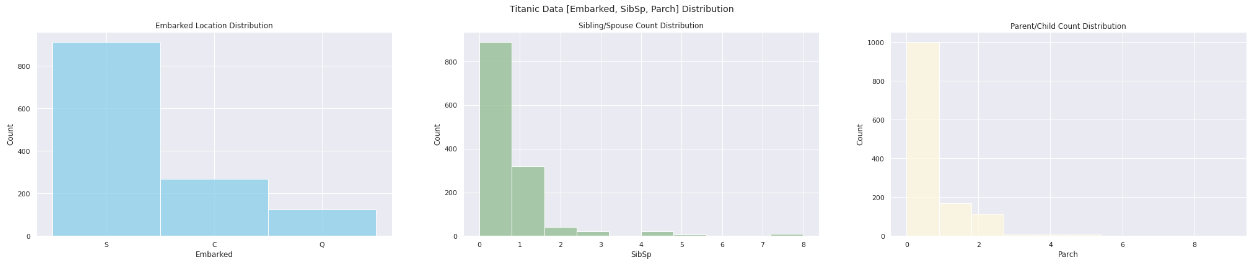
We can also see here that the test.csv does not provide true values for "Survived" given the 418 missing values.

It is unfortunate that "Age" has so many missing values since it is one attribute of interest for our analysis. Pawar imputes the missing values for this attribute, so we will use the same imputations for our analysis.

Frequency Plots for categorical data:

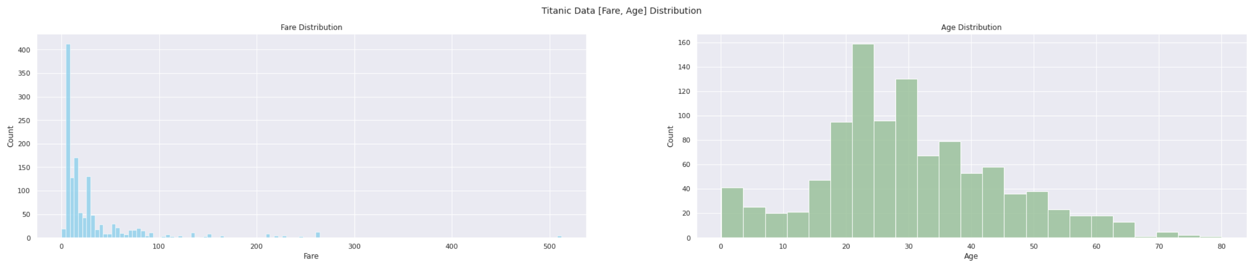


It appears here that there were almost twice as many people who did not survive, who were in the 3rd Passenger class and who were male. These may tell us more information about what we choose to be our protected attributes, privileged groups and unprivileged groups



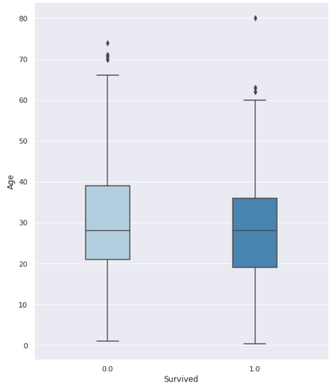
These categorical values are skewed far to the right. We're not sure if we want to use these for our fairness analysis given the number of null values and lack of relevance in contributing to survival.

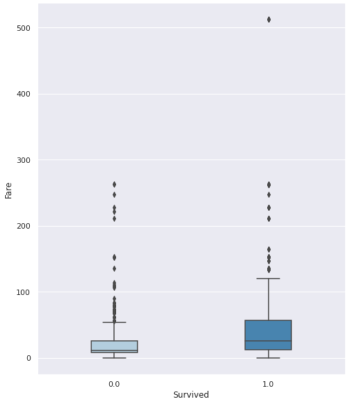
Frequency Plots for Age and Fare:



These distributions tell us that the majority of passengers were aged between 20 and 30 years of age. We may want to look at the survival rate of different age groups later.

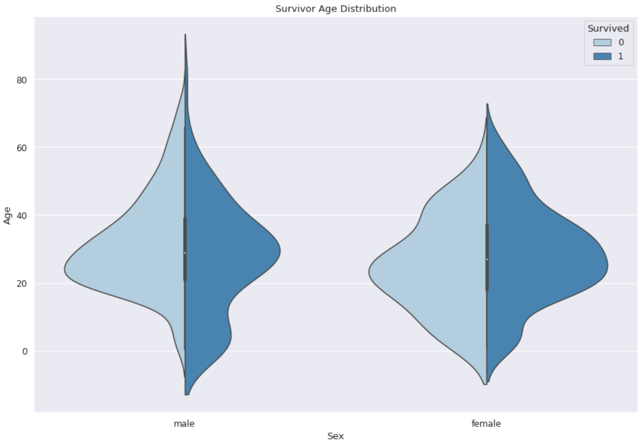
The fare distribution is extremely skewed to the right as well. There may be some data errors given that some passengers paid nearly 250 dollars for a ticket! That's almost 7.5k dollars today!



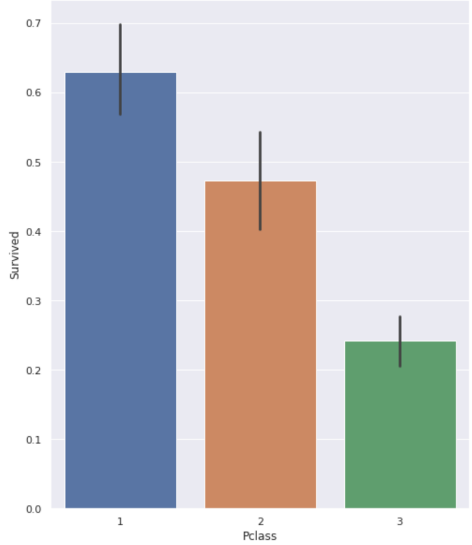


These box plots tell us a lot more about the outliers and errors we observed in the fare distribution. Majority of the ticket prices were below 50 dollars and anything above 100 dollars is considered an outlier.

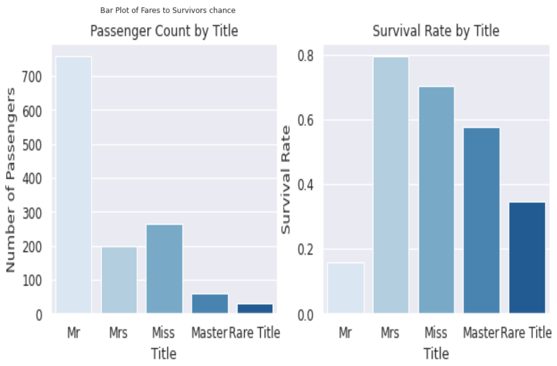
For the age distribution, it seems the distribution of those who survived are slightly younger than those who did not. However, those who did not survive still have a comparable distribution to those who did.



This plot is the most interesting and tells us the most about our privileged and unprivileged groups. It seems that young adult males did not survive as much as young adult females. Younger passengers (male or female) were given the priority of survival. As Pawar notes as well, these distributions fall into the line of "saving the women and children first"



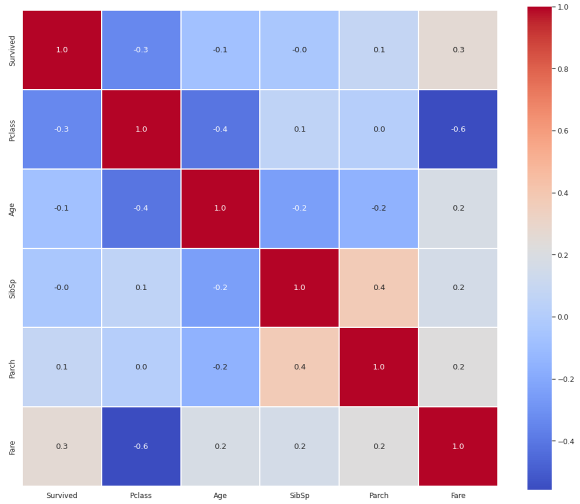
This plot shows us the rate of survival for different passenger ticket classes. 1 being the best and 3 being the worst. As it is clear here, those who were in 1st class or 2nd class had a higher rate of survival than those with 3rd class tickets.



This visualization was created by Pawar and shows the distribution of survivors given their title. We thought it was interesting to include as a visualization, but we decided not to use titles in our fairness analysis given the number of uncommon/rare titles. However, this does tell us that men (yet again) had the lowest survival rate, even though the greatest number of passengers are men.

This visualization shows us the distribution of survival given the fare price. Pawar bins these values, and we decided to do the same given the large variation of fare prices and outliers. Here, we decided that when comparing fare bins, those with a low or median fare range were underprivileged while those with high or very high fare ranges were privileged.

From this correlational matrix, we see that survival has the strongest correlations with Fare and Passenger Class, which we will choose to analyze the fairness of later. Ticket class and passenger fare are strongly negatively correlated, which tells us that the 1st class tickets are more expensive than 3rd class.



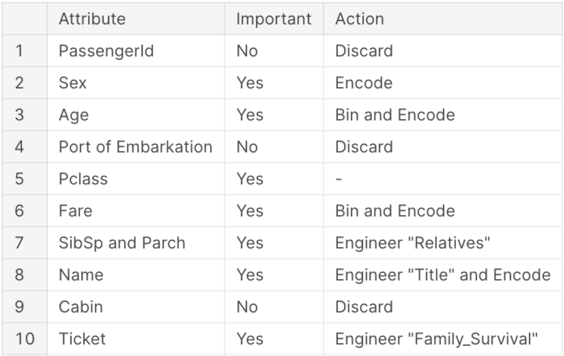
## c. What is the output of the system (e.g., is it a class label, a score, a probability, or some other type of output), and how do we interpret it?

The output of the system is a class label, which predicts whether or not a passenger survives based on the features provided in the data, as well as the features engineered by Pawar (i.e.: Family Survival).

## 3. Implementation and validation: present your understanding of the code that implements the ADS. This code was implemented by others (e.g., as part of the Kaggle competition), not by you as part of this assignment. Your goal here is to demonstrate that you understand the implementation at a high level.

## a. Describe data cleaning and any other pre-processing

Pawar decides to drop and encode the data in a few ways before training his ADS. To re-create his ADS, we decided to follow his same methods.



| | Attribute | Important | Action |
|---|---|---|---|
| 1 | PassengerId | No | Discard |
| 2 | Sex | Yes | Encode |
| 3 | Age | Yes | Bin and Encode |
| 4 | Port of Embarkation | No | Discard |
| 5 | Pclass | Yes | - |
| 6 | Fare | Yes | Bin and Encode |
| 7 | SibSp and Parch | Yes | Engineer "Relatives" |
| 8 | Name | Yes | Engineer "Title" and Encode |
| 9 | Cabin | No | Discard |
| 10 | Ticket | Yes | Engineer "Family_Survival" |

Pawar then uses an IterativeImputer to impute null values in the entire dataset (train and test). We decided to allow imputations for "Age" in our own analysis, but use only true values from the train dataset (not use imputed values for Survived).

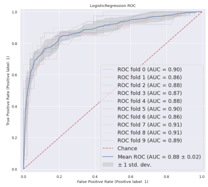
## b. Give high-level information about the implementation of the system

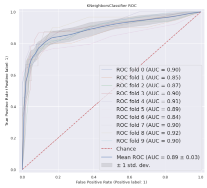
Pawar encodes the existing data into attributes that have the greatest correlation with survival before creating his ADS model. So for original attributes like “Age”, “SibSp”, “Parch”, “Fare”, and “Ticket”, he creates new attributes with the existing data. To create his ADS, Pawar first observes the base predictions of generic classifiers (without modification) (XGBClassifier, LGBMClassifier, RandomForestClassifier, KNeighborsClassifier, DecisionTreeClassifier, AdaBoostClassifier, LogisticRegression, GaussianNB). Then, Pawar creates a Voting Classifier that combines the predictions of all these base classifiers and observes predictive performance compared to each constituent individual classifier (known as Ensemble Learning). Pawar tunes the base classifiers using `GridSearchCV()` to find the best parameters that improve classifier accuracy and chooses the best estimator models. Pawar recreates the Voting Classifier with the tuned classifiers – creating an ADS that values accuracy and consistency among all the different base classifiers.

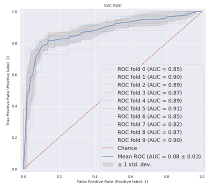
## c. How was the ADS validated? How do we know that it meets its stated goal(s)?

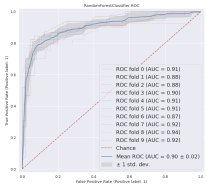
Pawar validates his ADS in multiple ways:

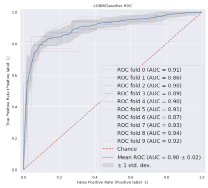
- He observes ROC of individual classifiers (Images provided from Pawar’s code implementation)

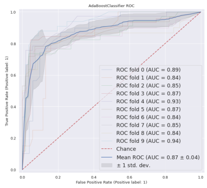




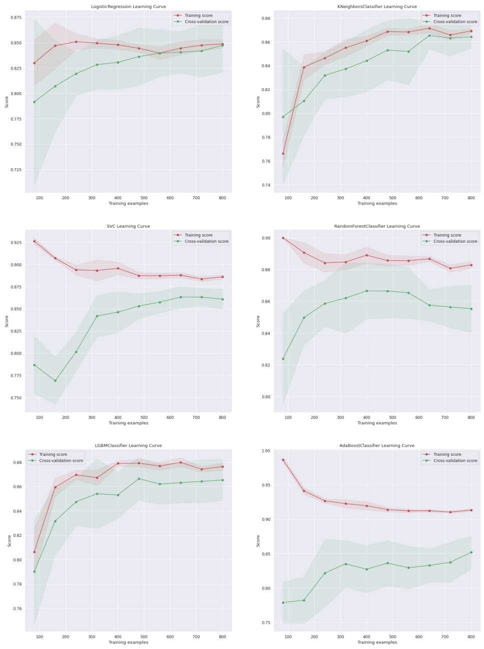




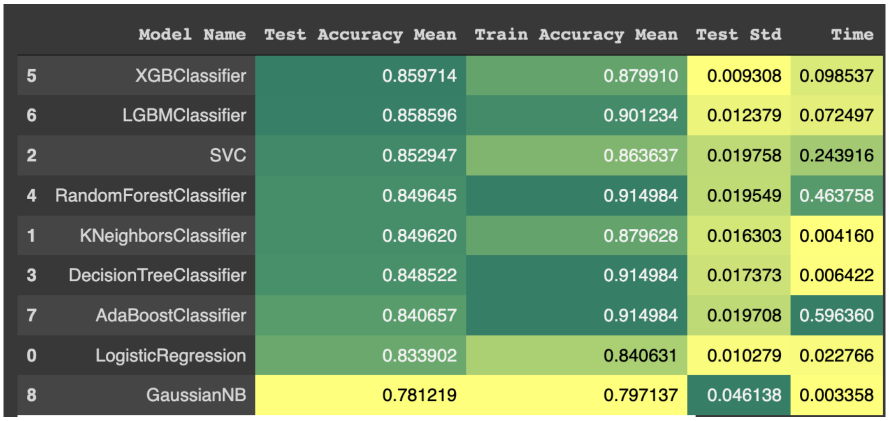




- Observe Learning Curve of individual classifiers (if the model is overfitting or underfitting) (Images provided from Pawar’s code implementation)

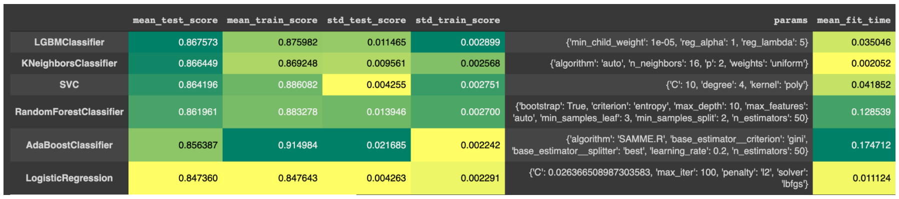


- Observe Neighborhood Components Analysis (NCA), which tries to find a feature space such that a stochastic nearest neighbor algorithm will give the best accuracy (Images provided from Pawar's code implementation)



| | Model Name | Test Accuracy Mean | Train Accuracy Mean | Test Std | Time |
|---|---|---|---|---|---|
| 5 | XGBClassifier | 0.859714 | 0.879910 | 0.009308 | 0.098537 |
| 6 | LGBMClassifier | 0.858596 | 0.901234 | 0.012379 | 0.072497 |
| 2 | SVC | 0.852947 | 0.863637 | 0.019758 | 0.243916 |
| 4 | RandomForestClassifier | 0.849645 | 0.914984 | 0.019549 | 0.463758 |
| 1 | KNeighborsClassifier | 0.849620 | 0.879628 | 0.016303 | 0.004160 |
| 3 | DecisionTreeClassifier | 0.848522 | 0.914984 | 0.017373 | 0.006422 |
| 7 | AdaBoostClassifier | 0.840657 | 0.914984 | 0.019708 | 0.596360 |
| 0 | LogisticRegression | 0.833902 | 0.840631 | 0.010279 | 0.022766 |
| 8 | GaussianNB | 0.781219 | 0.797137 | 0.046138 | 0.003358 |

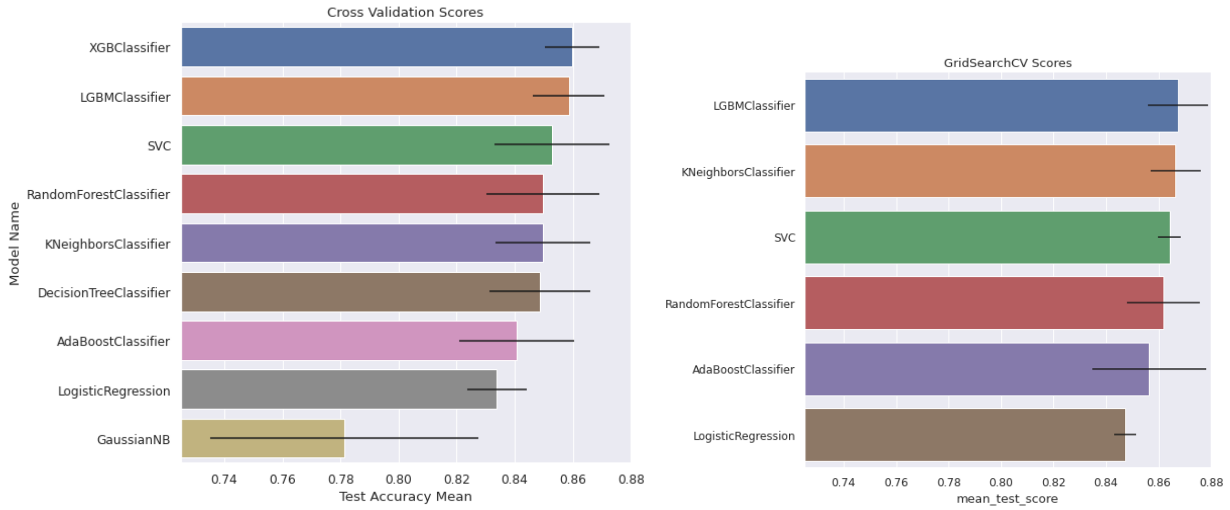
*For base classifiers (untuned) ^



| | mean_test_score | mean_train_score | std_test_score | std_train_score | params | mean_fit_time |
|---|---|---|---|---|---|---|
| LGBMClassifier | 0.867573 | 0.875982 | 0.011465 | 0.002899 | {'min_child_weight': 1e-05, 'reg_alpha': 1, 'reg_lambda': 5} | 0.035046 |
| KNeighborsClassifier | 0.866449 | 0.869248 | 0.009561 | 0.002568 | {'algorithm': 'auto', 'n_neighbors': 16, 'p': 2, 'weights': 'uniform'} | 0.002052 |
| SVC | 0.864196 | 0.886082 | 0.004255 | 0.002751 | {'C': 10, 'degree': 4, 'kernel': 'poly'} | 0.041852 |
| RandomForestClassifier | 0.861961 | 0.883278 | 0.013946 | 0.002700 | {'bootstrap': True, 'criterion': 'entropy', 'max_depth': 10, 'max_features': 'auto', 'min_samples_leaf': 3, 'min_samples_split': 2, 'n_estimators': 50} | 0.128539 |
| AdaBoostClassifier | 0.856387 | 0.914984 | 0.021685 | 0.002242 | {'algorithm': 'SAMME.R', 'base_estimator__criterion': 'gini', 'base_estimator__splitter': 'best', 'learning_rate': 0.2, 'n_estimators': 50} | 0.174712 |
| LogisticRegression | 0.847360 | 0.847643 | 0.004263 | 0.002291 | {'C': 0.026366508987303583, 'max_iter': 100, 'penalty': 'l2', 'solver': 'lbfgs'} | 0.011124 |

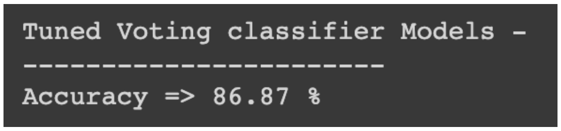
*For tuned classifiers (tuned) ^

- Observe Accuracy Score of each classifier (Images provided from Pawar's code implementation)



*Untuned on the left, tuned on the right

- Observe Accuracy of Mean Score of final Voting Classifier (Images provided from Pawar's code implementation)



```
Tuned Voting classifier Models -
-----------------------
Accuracy => 86.87 %
```

## 4. Outcomes

### a. Analyze the effectiveness (accuracy) of the ADS by comparing its performance across different subpopulations.

In order to analyze the effectiveness (accuracy) of our chosen ADS, we used `aif360` to compare performance across different subpopulations:

- [Female (privileged), Male (unprivileged)]
- [40 years old or below (privileged), Above 40 years old (unprivileged)]
- [High/very high fare (privileged), lower fares (unprivileged)]
- [First and second class (privileged), lower classes (unprivileged)]

For data preparation, we converted all data types to be compatible with `aif360`, and we converted the variables 'Age', 'Fare_Bin', and 'Pclass' to be binary based on thresholds we set. We then split the imputed `train.csv` data into training and test sets for each protected attributes (sex, age, fare, class), and standardized the data using a `StandardScaler()`. Next, we fitted Pawar's ADS with each training set, gathered the model's predicted y-values, converted both the prediction set and the test set to `AIF StandardDatasets`, and calculated metrics comparing privileged and unprivileged groups (which we set according to the plots produced earlier).

For the variable ‘Sex’, we set Female as privileged and Male as unpriviliged. The accuracy for Females is lower than for Males (0.72 v. 0.78), and the overall accuracy is 0.76. The AUC score is 0.73. Precision is 0.71, and statistical parity difference is -0.208.

For the variable ‘Age’, we set 40 years or below as privileged and above 40 as unprivileged. The accuracy for <=40 is lower than for >40 (0.82 v. 0.83), and the overall accuracy is 0.82. The AUC score is 0.79. Precision is 0.84, and statistical parity difference is -0.04.

For the variable ‘Fare’, we set high/very high fares as privileged and lower fares as unprivileged. The accuracy for high/very high fares is lower than for lower fares (0.8 v. 0.85), and the overall accuracy is 0.82. The AUC score is 0.79. Precision is 0.83, and statistical parity difference is -0.24.

For the variable ‘Class’, we set 1st and 2nd class as privileged and 3rd class as unprivileged. The accuracy for first and second class is lower than for lower fares (0.77 v. 0.85), and the overall accuracy is 0.81. The AUC score is 0.77. Precision is 0.82, and statistical parity difference is -0.21.

Overall, we see a consistent tradeoff between accuracy and disparate impact (discussed in next section) where the privileged groups have lower accuracy because there is bias favoring the privileged groups. If we look at the AUC scores, they generally tell us that Pawar’s model’s aggregate performance across all possible classification thresholds is fair, given the > 70% AUC scores. The AUC scores tell us that the model has a higher chance of detecting true positives and true negatives. Precision is around 70-80% for all of the protected attributes, lowest precision being 0.71 for ‘Sex’. The statistical parity difference is around -0.2 for all of the protected attributes except for ‘Age’, which is valued at -0.04. This indicates that the probability of surviving is lower for unprivileged classes in all cases, but there is less of a difference for the different age groups.

## b. Select one or several fairness or diversity measures, justify your choice of these measures for the ADS in question, and quantify the fairness or diversity of this ADS.

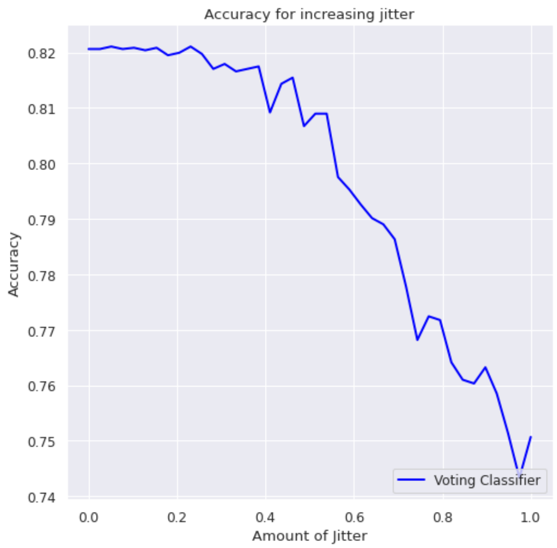
Looking at disparate impact for “Sex” (0.56), the ADS seems to favor Females over Males. The FPR difference of 0.15 indicates that Males have a higher false positive rate than Females. The results are consistent with our original thoughts and hypothesis that the model would be biased towards giving females the more positive outcome (Survival). Based on the disparate impact score, the model is biased against males, giving them a more unfavorable outcome. The tradeoff between accuracy and fairness is also depicted by these results – where a more accurate model is generally more unfair.

Looking at disparate impact for “Age” (0.86), the ADS seems to slightly favor <=40 over >40. The FPR difference of -0.05 indicates that those <=40 have a slightly higher false positive rate than >40. This is consistent with the barplot for age distribution observed earlier – the distributions of age between survivors and non-survivors are similar, but generally younger passengers were prioritized for survival.

Looking at disparate impact for “Fare” (0.4), the ADS seems to heavily favor high/very high fares over lower fares. The FPR difference of 0.007 indicates that lower fares have a slightly higher false positive rate than high/very high fares. The results for this group were interesting – especially for the FPR difference, which indicated the unprivileged group having a more likely chance of receiving a false positive result.

Looking at disparate impact for “Class” (0.48), the ADS seems to favor first and second class over lower classes. The FPR difference of 0.03 indicates that lower classes have a slightly higher false positive rate than first and second class.

### c. Develop additional methods for analyzing ADS performance: think about stability, robustness, performance on difficult or otherwise important examples (in the style of LIME), or any other property that you believe is important to check for this ADS.



At first, we attempted to analyze feature importances in Pawar's VotingClassifier, but the ensemble model made it difficult to extract feature importances given how many ML models were used in the final VotingClassifier. In other words, each classifier in the final VotingClassifier may have used different features and it is difficult to extract feature scores from the whole VotingClassifier. +

Instead, we decided to test the robustness of Pawar's ADS by adding some noise to the test data. We decided to alter the magnitude of the noise to infer how well the model will perform with new data and different sources of noise. In our analysis example we add some random, normally-distributed noise. This test is what we call a `jitter_test` that runs a prediction on the new jitter data over several different jitter scales (standard deviations). To make the resulting curves a little smoother, we're performing the experiment several times and taking the average. This code was sourced and interpreted from here.

Overall, we can see that the ADS loses its robustness after adding more than 0.6 standard deviations of noise.

## 5. Summary

### a. Do you believe that the data was appropriate for this ADS?

Yes, this ADS seems to appropriately use the data provided by the Kaggle competition. The original dataset has many missing values that need to be imputed however. The test dataset also was missing true values.

### b. Do you believe the implementation is robust, accurate, and fair? Discuss your choice of accuracy and fairness measures, and explain which stakeholders may find these measures appropriate.

**Accuracy**: The implementation seems to be relatively accurate, as the ADS focuses on accuracy and results in an overall score of about 80% accuracy. Across privileged and unprivileged groups, accuracy does not fall below 70% either. Therefore, we can conclude that Pawar's model does prioritize accuracy and is successful in doing so. Precision would be an appropriate measure of accuracy, as the goal of this ADS and our evaluation of the ADS is to be sure of the predictions being made — *all passengers/future* *passengers of potentially dangerous transportation could benefit from this measure*. Life insurance companies may find precision appropriate to determine/recommend insurance policy, rates and contracts given to a certain applicant in an area where a disaster may occur.

**Robustness**: From the Jitter Test, we conclude that the ADS is robust to a certain level by adding noise to the dataset. It may have been more effective to observe how null values may have affected the robustness of the model, but generally, the ADS is robust when some noise is added to the dataset.

**Fairness**: As discussed earlier, this ADS is biased towards privileged groups (female, younger, higher class, higher ticket price) – which is expected given the tradeoff of accuracy. The fairness measurements only support earlier observations from visualizations. Furthermore, predictive parity would be an appropriate measure of fairness, as it checks whether the precision rates are equivalent amongst different subgroups — as we observe, predictive parity differs between privileged and unprivileged groups. We are unsure who would benefit from this fairness measure other than future disaster response groups, which seems quite unethical given a machine determines who should be saved or who should not be saved.

### c. Would you be comfortable deploying this ADS in the public sector, or in the industry? Why so or why not?

No, as this ADS predicts whether a hypothetical individual would have survived a historical event. It is difficult to apply this ADS to any public sector or industry given the original testing data. Perhaps this could be used by (life) insurance companies to recommend certain insurance policies/rates/contracts given a larger dataset with other disastrous events or determining the potential survival rate of terminally ill patients to accept more patients; however it feels morally wrong to depend on a computer system to predict whether or not certain individuals will survive disastrous events. Predictions could be incorrect, and then influence individuals to behave in certain ways.

### d. What improvements do you recommend to the data collection, processing, or analysis methodology?

**Data Collection:** It would be helpful to expand or collect a larger sample of boat-related or disaster events in general. The current dataset is based on a single limited event with a lot of missing data most likely due to the fact that the information is historical.

**Processing:** We have not tried any to incorporate any additional pre- , in-, or post processing methods other than the ones that Pawar originally added, so it may have been interesting to observe how these methods may have impacted our fairness or robustness analysis, given that the data is biased towards certain groups, leading to a biased model (impact of pre-existing bias). In fact, we may have tried to not impute any values (or attempted another method to impute values)  as it may have skewed the bias of the data.

**Analysis Methodology:** We would have liked to analyze feature importances for this ADS, but given our limited ability to extract feature importances and their scores from each model in the final `VotingClassifer` ADS, we were unable to do so. Feature importance would allow us to observe which features were given the greatest weight in the ADS which would allow us to understand the relationship between the features and the target variable, since a correlation matrix is not very specific.