|  |  |  |
| --- | --- | --- |
| Understand data | Make first observation | 1 Passengerid is useless  2 Extract name info  3 Sex convert to numeric value  4 Missing age  5 Ticket is useful?  6 Carbin too many null, may drop this info  7 Embark has null  8 Embark convert to numeric value  9 Fare needs to be grouped  10 Age needs to be grouped |
|  | Use python to read basic data stats |  |
|  | Use tableau to visualize data from first observation | <https://public.tableau.com/profile/kenny2174#!/vizhome/Kaggle_titanic_15746524123480/agebin> |
| Feature engineering plan | List to-do | Train  7 Fill in missing Embark  Test  12 Fill in missing Fare  Both  1 Remove passengerid  2 Extract name info (Mr. Miss.)  3 Sex convert to numeric value (0 male, 1 female)  4 Fill in missing age  5 Remove tickets  6 Remove Carbin  8 Embark convert to numeric value (0S 1C 2Q)  9 Group fare (bin size )  10 Group age (bin size )  11 FamilySize/IsAlone  13 nameLen |
| Address issues in to-do list | Address issues in to-do list | 2 get Title first using Regex then convert it to ordinal value  title\_mapping = {"Mr": 1, "Miss": 2, "Mrs": 3, "Master": 4, "Rare": 5}  4 More accurate way of guessing missing values is to use other correlated features. In our case we note correlation among Age, Gender, and Pclass. Guess Age values using median values for Age across sets of Pclass and Gender feature combinations. So, median Age for Pclass=1 and Gender=0, Pclass=1 and Gender=1, and so on…  10 put age into 5 bins in the range of min/max, then convert age to ordinal value based on age bins  9 and 12 Used median then convert fare to ordinal value (hard-coded 7.91/14.454/31)  7 only 2 missing Embarked in train\_df, simply replace them with the most frequent one  11 FamilySize = SibSp + Parch; IsAlone is based on FamilySize, choose to keep SibSp/Parch |
| Implement feature engineering |  |  |
| Model | a classification and regression problem |  |
|  |  |  |

Starter: <https://www.kaggle.com/startupsci/titanic-data-science-solutions>

Stacking: <https://www.kaggle.com/arthurtok/introduction-to-ensembling-stacking-in-python>

Feature engineering: <https://www.kaggle.com/gunesevitan/advanced-feature-engineering-tutorial-with-titanic>

Generic Algo: <https://www.kaggle.com/guesejustin/91-genetic-algorithms-explained-using-geap>

Ultimate: <https://www.kaggle.com/ldfreeman3/a-data-science-framework-to-achieve-99-accuracy>