df = {'Algorithm': ['KNN', 'Decistion Tree', 'SVM', 'LogisticRegression'], \

'Jaccard': jaccard\_score, 'F1-score': F1\_score, 'LogLoss': Logloss\_score}

df\_final = pd.DataFrame(data=df, columns=['Algorithm', 'Jaccard', 'F1-score', 'LogLoss'], index=**None**)

df\_final

*# Feature Engineering*

df = test\_df

df['due\_date'] = pd.to\_datetime(df['due\_date'])

df['effective\_date'] = pd.to\_datetime(df['effective\_date'])

df['dayofweek'] = df['effective\_date'].dt.dayofweek

df['weekend'] = df['dayofweek'].apply(**lambda** x: 1 **if** (x>3) **else** 0)

df.groupby(['Gender'])['loan\_status'].value\_counts(normalize=**True**)

df['Gender'].replace(to\_replace=['male','female'], value=[0,1],inplace=**True**)

df.groupby(['education'])['loan\_status'].value\_counts(normalize=**True**)

Feature = df[['Principal','terms','age','Gender','weekend']]

Feature = pd.concat([Feature,pd.get\_dummies(df['education'])], axis=1)

Feature.drop(['Master or Above'], axis = 1,inplace=**True**)

X\_test = Feature

y\_test = df['loan\_status'].values

X\_test = preprocessing.StandardScaler().fit(X\_test).transform(X\_test)

df.drop(['IncidntNum','Category','Descript','DayOfWeek','Date','Time','Resolution','Address','X','Y','Location','PdId'],1, inplace=True)

<https://github.com/aprajita11/github-first/blob/master/Segmenting%20and%20Clustering%20Assignment.ipynb>

df['loan\_status'].value\_counts()