import pandas as pd

from sklearn.preprocessing import RobustScaler

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score

from sklearn.preprocessing import MinMaxScaler

import lightgbm as lgb

def remove\_outliers\_iqr\_all\_numerical(df, threshold=1.5):

numerical\_columns = df.select\_dtypes(include='number').columns

df\_filtered = df.copy()

for column\_name in numerical\_columns:

Q1 = df[column\_name].quantile(0.25)

Q3 = df[column\_name].quantile(0.75)

IQR = Q3 - Q1

lower\_bound = Q1 - threshold \* IQR

upper\_bound = Q3 + threshold \* IQR

df\_filtered = df\_filtered[(df\_filtered[column\_name] >= lower\_bound) & (df\_filtered[column\_name] <= upper\_bound)]

return df\_filtered

train\_data = pd.read\_csv('/content/train.csv')

train\_data.drop('id', axis=1, inplace=True)

test\_data = pd.read\_csv('/content/test.csv')

train\_data.dropna(inplace=True)

m = test\_data['id']

test\_data.drop('id', axis=1, inplace=True)

X = train\_data.drop("Response", axis=1)

y = train\_data['Response']

lgb\_params2 = {

'n\_estimators': 100,

'max\_depth': 6,

'learning\_rate': 0.05,

'subsample': 0.20,

'colsample\_bytree': 0.56,

'reg\_alpha': 0.25,

'reg\_lambda': 5e-08,

'objective': 'binary',

'boosting\_type': 'gbdt',

'device': "cpu",

'random\_state': 0,

'metric': 'roc\_auc',

'verbose': -1,

'n\_jobs': -1,

'metric': 'roc\_auc',

'seed': 43

}

model = lgb.LGBMRegressor(\*\*lgb\_params2)

model.fit(X, y)

y\_predict = model.predict(test\_data)

c = pd.DataFrame(y\_predict)

c.rename(columns={0: "Response"}, inplace=True)

c.insert(0, 'id', m)

c.to\_csv('predictied.csv', index=False)

|  |  |
| --- | --- |
| lightgbm | 0.89349 |
| Forrest Regression | **[0.](https://www.kaggle.com/competitions/cis-321-data-mining-2nd2024/submissions)**  **[0.78652](https://www.kaggle.com/competitions/cis-321-data-mining-2nd2024/submissions)** |
| Linear regression | **[0.84804](https://www.kaggle.com/competitions/cis-321-data-mining-2nd2024/submissions)** |
| Naïve baise | **[0.87681](https://www.kaggle.com/competitions/cis-321-data-mining-2nd2024/submissions)** |
|  |  |