

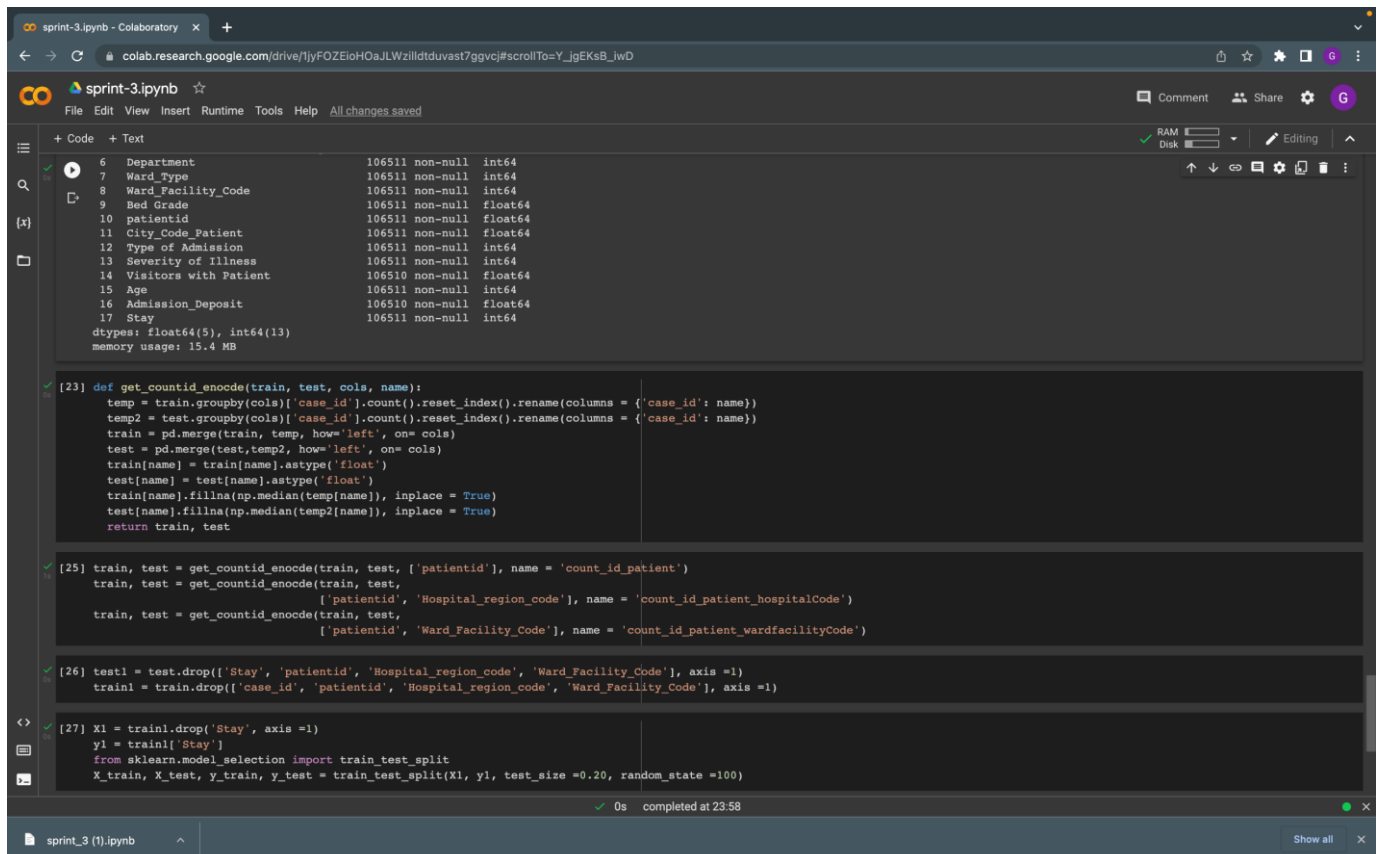
SPRINT 3

Date	13-11-2022
Team ID	PNT2022TMID04531
Project Title	Analytics for Hospitals' Health-Care Data
Team Members	Haris S, Gokul Kannan G, Abishek A, Hari vignesh K G

Feature Engineering

Once the data is cleaned and prepared, we grouped patientid and case_id to extract the new column “count_id_patient”. This variable contains the count of multiple admits of a patient under different case_id. Further two more columns “Hospital_region_code” and “ward_facility_code” were grouped to patientid and case_id. These two new variables “count_id_patient_hospitalCode” and “count_id_patient_wardfacilityCode” contain the count of multiple admissions in a hospital region and the count of multiple wards allocated to a patient.

Before getting into analysis, the train data must be split into two parts, the first part with all the feature variables and the second part with a target variable (“Stay”). Then preprocessed into train and validation sets. So, here we are portioning the train set with 80% and validation set with 20% of the data for Naïve Bayes and XGBoost models.



```
sprint-3.ipynb - Colaboratory
colab.research.google.com/drive/1jyFOZEioH0aJLWzilldtduvast7ggvcj#scrollTo=Y_jgEKsB_jwD

sprint-3.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share Settings

+ Code + Text
6 Department 106511 non-null int64
7 Ward_Type 106511 non-null int64
8 Ward_Facility_Code 106511 non-null int64
9 Bed_Grade 106511 non-null float64
10 patientid 106511 non-null float64
11 City_Code_Patient 106511 non-null float64
12 Type of Admission 106511 non-null int64
13 Severity of Illness 106511 non-null int64
14 Visitors With Patient 106510 non-null float64
15 Age 106511 non-null int64
16 Admission_Deposit 106510 non-null float64
17 Stay 106511 non-null int64
dtypes: float64(5), int64(13)
memory usage: 15.4 MB

[23] def get_countid_encode(train, test, cols, name):
    temp = train.groupby(cols)[case_id].count().reset_index().rename(columns = {'case_id': name})
    temp2 = test.groupby(cols)[case_id].count().reset_index().rename(columns = {'case_id': name})
    train = pd.merge(train, temp, how='left', on= cols)
    test = pd.merge(test, temp2, how='left', on= cols)
    train[name] = train[name].astype('float')
    test[name] = test[name].astype('float')
    train[name].fillna(np.median(train[name]), inplace = True)
    test[name].fillna(np.median(temp2[name]), inplace = True)
    return train, test

[25] train, test = get_countid_encode(train, test, ['patientid'], name = 'count_id_patient')
train, test = get_countid_encode(train, test,
    ['patientid', 'Hospital_region_code'], name = 'count_id_patient_hospitalCode')
train, test = get_countid_encode(train, test,
    ['patientid', 'Ward_Facility_Code'], name = 'count_id_patient_wardfacilityCode')

[26] test1 = test.drop(['Stay', 'patientid', 'Hospital_region_code', 'Ward_Facility_Code'], axis =1)
train1 = train.drop(['case_id', 'patientid', 'Hospital_region_code', 'Ward_Facility_Code'], axis =1)

[27] X1 = train1.drop('Stay', axis =1)
y1 = train1['Stay']
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X1, y1, test_size =0.20, random_state =100)

0s completed at 23:58
sprint_3 (1).ipynb Show all
```

Choosing ML models for analysis:

The goal is to predict Length of Stay i.e., “Stay” column (Target Variable) and it is classified into 11 levels. We must find the probability of each patient’s length of stay using feature variables, which contain the patient’s condition and hospital-level information.

After analysis of different ML models for classification, we have decided to choose three models:

- 1) Naives Bayes Model
- 2) XGboost
- 3) Neural Networks

We have chosen Naïve Bayes model, as the feature variables are ordinal in nature and also Naïve Bayes Model is a perfect multilevel classifier.

So, the model 2 we have decided to use XGBoosting. Boosting is a sequential technique that works on the principle of an ensemble model. It combines the set of weak learners and improves prediction accuracy. The final prediction score of the model is calculated by summing up each and individual score.

The 3rd model we have chosen to be neural network because in NN, increasing the number of neurons from each layer to the other layer, will increase the hypothetical space of the model and try to learn more patterns from the data. So, this can better predict the Length of stay of patients.

Sprint 3 completed successfully