

### Sprint 3

Date	13.11.22
Team ID	PNT2022TMID53212
Project Title	Analytics for Hospitals' Health-Care Data
Team Members	Jaikishore R, Chitiprolu Prathyusha, Duvicksha U, Kapireddy Charitha

#### 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.

```
Healthcare analytics .ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Feature Engineering

[34] def get_countid_enocde(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(temp[name]), inplace = True)
      test[name].fillna(np.median(temp2[name]), inplace = True)
      return train, test

[35] train, test = get_countid_enocde(train, test, ['patientid'], name = 'count_id_patient')
      train, test = get_countid_enocde(train, test,
      ['patientid', 'Hospital_region_code'], name = 'count_id_patient_hospitalCode')
      train, test = get_countid_enocde(train, test,
      ['patientid', 'Ward_Facility_Code'], name = 'count_id_patient_wardfacilityCode')
```

```
Healthcare analytics .ipynb
File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

[36] # Dropping duplicate columns
      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)

# Splitting train data for Naive Bayes and XGBoost
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)
```

## Choosing ML Model 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.

## JIRA Spring 3 Tracking

The screenshot shows the Jira Software interface for the 'IBM\_53212' project. The left sidebar contains navigation options: 'PLANNING' (Roadmap, Backlog, Board) and 'DEVELOPMENT' (Code, Project pages, Add shortcut, Project settings). The main area displays the 'I5 Sprint 3' board. The board has three columns: 'TO DO 3 ISSUES', 'IN PROGRESS', and 'DONE'. The 'TO DO' column contains three issues: 'Feature engineering of dataset' (ID 15-36), 'Model Analysis' (ID 15-37), and 'Choosing preferred model for analysis' (ID 15-38). The 'IN PROGRESS' and 'DONE' columns are currently empty. The board is set to 'GROUP BY None' and has a '14 days remaining' timer.

The screenshot shows the Jira Software interface for the 'IBM\_53212' project. The left sidebar contains navigation options: 'PLANNING' (Roadmap, Backlog, Board) and 'DEVELOPMENT' (Code, Project pages, Add shortcut, Project settings). The main area displays the 'I5 Sprint 3' board. The board has three columns: 'TO DO 1 ISSUE', 'IN PROGRESS 1 ISSUE', and 'DONE 1 ISSUE'. The 'TO DO' column contains one issue: 'Choosing preferred model for analysis' (ID 15-38). The 'IN PROGRESS' column contains one issue: 'Model Analysis' (ID 15-37). The 'DONE' column contains one issue: 'Feature engineering of dataset' (ID 15-36). The board is set to 'GROUP BY None' and has a '10 days remaining' timer.

## Sprint 3 Completed Successfully

The screenshot shows the Jira Software interface for a project named 'IBM\_53212'. The 'I5 Sprint 3' board is displayed, showing a Kanban-style workflow with columns for 'TO DO', 'IN PROGRESS', and 'DONE 3 ISSUES'. The 'DONE 3 ISSUES' column contains three completed issues: 'Feature engineering of dataset' (ID 15-36), 'Model Analysis' (ID 15-37), and 'Choosing preferred model for analysis' (ID 15-38). The interface includes a sidebar with navigation options like 'Roadmap', 'Backlog', and 'Board'. The top navigation bar shows 'Your work' and 'Projects' tabs. The bottom status bar indicates 'You're in a team-managed project'.

The screenshot shows the same Jira Software interface, but with a modal dialog box titled 'Complete I5 Sprint 3' overlaid. The dialog box contains a blue ribbon icon and the text: 'This sprint contains 3 completed issues. That's all of them - well done!'. At the bottom of the dialog, there are two buttons: 'Complete sprint' and 'Cancel'. The background interface is dimmed, showing the same 'I5 Sprint 3' board and sidebar.