UNDERSTANDING PATIENT PERSISTENCY A LOOK AT THE DATA

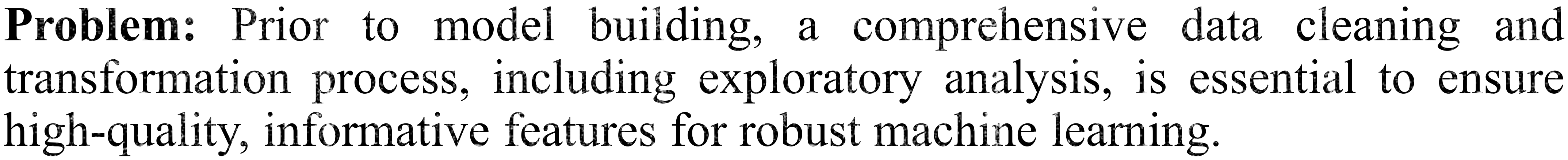
Name: Data Science :: Healthcare - Persistency of a drug:: Group Project

Internship Batch: LISUM30:30 Jan24 - 30 Apr 24

Data intake by: David Paul Nalumenya

Specialization: Data science

Email: nalumenyad@gmail.com



Problem: Prior to model building, a comprehensive data cleaning and transformation process, including exploratory analysis, is essential to ensure high-quality, informative features for robust machine learning.

- Data cleaning techniques were applied to address missing values and ensure data quality.
  - ➢Missing values were checked using .isnull().sum(), non were found
  - ➢Outliers were assessed in the integer columns only, non were found

Note: Therefore our data was properly curated and cleaned.

Statistical fact

- ➢Missing values and outliers can introduce bias into machine learning models, so addressing them is crucial for reliable results.

EXPLORING THE DATA STRUCTURE

- The data consisted of 3424 observations (patients) and 69 features.

- Most features (65 out of 69) were of object data type (categorical information).

- Only 2 features were of integer data type (numerical values).

- However there was an observation, the integer data type was also a representative of a categorical class.

FEATURE IMPORTANCE IN PREDICTING DRUG PERSISTENCY

- Feature engineering will be crucial to prepare the data for machine learning.
  - The patient ID (PTID) was not crucial for the down stream analysis, it was dropped.
- Feature engineering was done on the common feature categories include demographics, medical history, risk, medication information, and scan data.
  - All the features were categorized under five clusters, demographic data, medical risks, comorbidities data including concomitant medication, Count of risks, doctor specialty data.
  - Label encoding, univariate and bivariate analysis was done features in each cluster
  - Class imbalance was noticed in the outcome variable, corrected using SMOTE ()
- Data science fact
  - Feature engineering involves selecting, transforming, and creating new features that are most relevant to the prediction task.

EXPLORATORY DATA ANALYSIS

- Descriptive statistics for demographics identified a higher frequency of:
  - ➢Females
  - ➢Caucasian ethnicity
  - ➢Non-Hispanic origin
  - ➢Midwestern residence
  - ➢Age above 75 years

This demographic group also exhibited a higher prevalence of non-persistence.

- "No" was the most frequent response among the risk factor cluster, indicating a low prevalence of reported risks in the year preceding the initial NTM reaction.

RISK FACTORS, COMORBIDITIES, AND MEDICATION PATTERNS

- Comorbidities analysis showed a frequent lack of reported comorbidities associated with the specific therapy.
  - This could be due to genuine absence of comorbidities or limitations in data capture.
- General Practitioners (GPs) emerged as the most frequent prescribers, followed by the broader category of OB/GYN/Others/PCP/Unknown.
  - This reflects the potential involvement of various specialties in managing the condition.
- Non-use of glucocorticoids before and during NTM reactions, with similar patterns observed for prior DEXA scans, fragility fractures, and risk segment changes.

STANDARDIZATION AND NORMALIZATION

- Prior to model building, the standard scaler was used to scale the data variables within the 0,1 scale.
- This was a classification problem, therefore classification models were more appropriated;
  - Logistic regression model
  - Support vector machine
  - Random forest
  - Decision trees
  - Xgboosting

SMOTE() was used to handle outcome variable class imbalance, this would affect model quality and fit, this method is more preferable if the dataset is smaller, and fewer categories in the feature are present, in this case, they were two

- The Persistence_flag was the outcome variable, with rest of 67 variables as predictor variables
- Model accuracy, F1, precision and recall, AUC/ROC and feature importance for each model will be reported, models were compared and best model was chosen basing on the criteria above.

STANDARDIZATION AND NORMALIZATION

- Prior to model building, the standard scaler was used to scale the data variables within the 0,1 scale.

- This was a classification problem, therefore classification models were more appropriated;
  - Logistic regression model
  - Support vector machine
  - Random forest
  - Decision trees
  - Xgboosting

SMOTE() was used to handle outcome variable class imbalance, this would affect model quality and fit, this method is more preferable if the dataset is smaller, and fewer categories in the feature are present, in this case, they were two

- The Persistence_flag was the outcome variable, with rest of 67 variables as predictor variables
- Model accuracy, F1, precision and recall, AUC/ROC and feature importance for each model were reported, models were compared and best model was chosen basing on the criteria above.

MODEL SELECTION AND COMPARISON BEFORE CORRECTING CLASS IMBALANCE

- Xgboost model performed well on data where the class imbalance was not corrected with SMOTE, the reported accuracy was **0.8145985401459854, better than the rest of the models.**
- Logistic Regression Accuracy: 0.7941605839416058
- Random Forest Accuracy: 0.7985401459854015
- Decision Tree Accuracy: 0.7357664233576642
- SVM Accuracy: 0.7795620437956204

- Random forest and xgboost model had the highest accuracy compared to the other models.

MODEL SELECTION AND COMPARISON AFTER CORRECTING CLASS IMBALANCE

After correcting class imbalance;

- Logistic Regression Accuracy: 0.7775175644028103
- Random Forest Accuracy: 0.8290398126463701
- Decision Tree Accuracy: 0.7353629976580797
- SVM Accuracy: 0.7681498829039812
- XGBoost Accuracy: 0.8138173302107728

- **Two models, random forest, and Xgboost performed well, they gave the highest accuracy.**
- **However its important to note that, after correcting class imbalance, the model accuracy for all the models reduced significantly, suggesting that previously the models were overfit**

CORRECTING CLASS IMBALANCE AND HYPERPARAMETER TUNING FOR RANDOM FOREST MODEL

- There after, random forest and xgboost model were taken further for analysis until we choose the best model.
- SMOTE was maintained on the dependent variable, hyperparameter tuning done for both models and results were as below
- For random forest, hyperparameters gotten, max_depth=8, min_samples_leaf=1, min_samples_split=5, n_estimators=200, gave an accuracy of 0.7775175644028103, ROC AUC area was 0.86.
- The predicted important features were;

Persistency_Flag,Age_Bucket,Region,Dexa_Freq_During_Rx,Dexa_During_Rx,Ntm_Speciality,Comorb_Long_Term_Current_Drug_Therapy,Comorb_Encounter_For_Screening_For_Malignant_Neoplasms,Comorb_Encounter_For_Immunization,Comorb_Encntr_For_General_Exam_W_O_Complaint,_Susp_Or_Reprtd_Dx.

CORRECTING CLASS IMBALANCE AND HYPERPARAMETER TUNING FOR RANDOM FOREST MODEL

- For xgboost, SMOTE was used on the dependent variable, hyperparameter tuning done and results were as below;

- For xgboost, hyperparameters gotten were; learning_rate= 0.1, max_depth = 8, min_child_weight= 1, n_estimators=200.

- These consequently were used to build the model that gave an accuracy of 0.8255269320843092 and a ROC AUC area of 0.89

MODEL CHOICE

- After exploratory data analysis, hyperparameter tuning and feature engineering, xgboost performed better, it fit the data well compared to the other classification models.

Below is the classification report and model accuracy;

- XGBoost Accuracy: 0.8255269320843092
- AUC = 0.89

	Precision	Recall	F1
Non persistent	0.82	0.82	0.82
Persisten	0.83	0.83	0.83