/01

Persistency of a drug: Project review

Name

Wangu Ndungu

Email Address

nwangu349@gmail.com

Country

Kenya

College

Kenyatta University

Specialization

Data Science

Group Name: Feature Transformers

Agenda

Introduction
Data Wrangling Technique
Feature Transformation Strategies
Model Development Options
Model Evaluation Metrics
Best Solution for the project

Introduction

Drug Persistence is one of the main challenges faced by the pharmaceuticals industry according to a study carried out by Lexis Nexis in 2020. ABC, a pharma industry, would like to automate the process of flagging patients as either Persistent or Non-persistent. This will enable pharmacists to conduct proper follow-ups on the latter patients, making sure they're taking proper doses and completing their prescriptions.

The Feature transformers group has come up with a model that is capable of achieving this by analyzing a dataset of already flagged patients. Using the patients' medical characteristics such as age, T-score(a measure of bone density), medical adherence, just to mention a few, we were able to figure out certain trends and figure out which group of patients is more or less likely to be persistent. A supervised classification model was built from this data.

/05

Data wrangling Technique

First things first, our data had to undergo some pre-processing before we could proceed. Evaluation for missing data, outliers and duplicate values was done.
There was no missing data present. We dropped all duplicate values in the dataset.
We checked for outliers using a box-plot visual and a KDE plot. A heavy right skewness was observed in the "Dexa_Freq_During_Rx" column, so we used log transformation on it. it proved to improve skewness and eliminate most outliers. We replaced outliers on the "Count_Of_Risks" with its median value.
As for the categorical columns, outliers were detected using histograms. We however opted to leave them unchanged.

/06

—

Feature Transformation strategy

Ordinal encoding was used on the following ordinal columns: 'Age_Bucket', 'Tscore_Bucket_Prior_Ntm', 'Tscore_Bucket_During_Rx', a numerical value is assigned to each category.

For the binary columns, frequency encoding was used. This is a method in which the frequency of categories is utilized as labels. This approach occupies much less space as only one additional column is created and works just as well as one-hot encoding.

For our target column, we also used ordinal encoding to create only one column that will use 0 and 1 as its labels.

A few options were put into consideration before settling on a final model.

Logistic Regression model

We used Logistic Regression as our base model due to its ability to produce somewhat accurate results with minimal complexities. It outputs predictions about test data points on a binary scale, if it's above 0.5, it'll belong to class 1 and vice versa.

k-Nearest Neighbours model

This classifier finds a predefined number of training samples closest in distance to a test data point and predicts a label from them. It is proven to be a fairly simple and accurate method of prediction

Ensemble Model

Data is trained with different classifiers and each classifier will make its own prediction but the label with the majority vote is the one that will be the final prediction. Soft voting was used, whereby we took the average of the probabilities of each of the labels, and whichever label is having the highest average will be the final prediction.

XGBoost Model

XGBoost is a popular and efficient open-source implementation of the gradient-boosted trees algorithm. Gradient boosting is a supervised learning algorithm, which attempts to accurately predict a target variable by combining the estimates of a set of simpler, weaker models.

/07

Model Development Options

Accuracy

Logistic Regression Model:
0.88
k-Nearest Neighbours Model:
0.88
Ensemble Model:
0.96
XGBoost Model:
0.96

Precision

Logistic Regression Model:
0.8461538461538461
k-Nearest Neighbours Model:
0.8461538461538461
Ensemble Model:
0.9230769230769231
XGBoost Model:
0.9230769230769231

Recall

Logistic Regression Model:
0.9166666666666666
k-Nearest Neighbours Model:
0.9166666666666666
Ensemble Model:
1.0
XGBoost Model:
1.0

ROC-AUC score

Logistic Regression Model:
0.9807692307692307
k-Nearest Neighbours Model:
0.9807692307692308
Ensemble Model:
0.9935897435897436
XGBoost Model:
0.9935897435897436

Best Solution for the Project

Just by glancing at the previous slide, it's pretty obvious which models have better evaluation metrics; the ensemble and XGBoost models prove to be better.
They coincidentally have the same metrics but we chose to use the ensemble model for this project. Although XGBoost is faster, it is not easily scalable.

Its proven that two heads are better than one, making the ensemble model a suitable choice as it overcomes weaknesses experienced by single models such as variance, bias and noise.