**Introduction**

Dataset selected for the project HealthCare contains information associated with the patients that whether they suffer from heart disease or hypertension. Other information associated with patient data includes marital status, work type, residence, glucose level, BMI, etc.   
Target Variable selected for this dataset for further analysis is “**Stroke”,** dataset contains 43k entries, 12 attributes sourced from kaggle.

Using this, we are trying to achieve prediction for the people who are likely to suffer from heart disease.

We have developed 5 models by keeping two criteria in the dataset selected, which is:   
1. Keeping smoking status column by removing NA values.  
2. Eliminating smoking status column.

**Analysis**

The dataset that we selected contained a total of 12 different attributes out of which 4 were categorical and 8 were numeric. After performing data cleaning, we got 3% missing data value in “bmi” and 31% in “smoking\_status”. The missing values of the “bmi” column were replaced by the mean of the column. We have generated models based on two criteria, one where we have dropped the entire “smoking\_status” column and second where we have considered “smoking\_status” by dropping the NA values of that column.

We converted the categorical variables into numeric variables for obtaining outliers by interquartile range where we found 1% outliers in the “avg\_glucose” and “bmi” which were dropped. We considered “Stroke” as the target variable and have denoted it by Probability density.

The following charts represents the relationship of Stoke with other parameters like “Age”, “hypertension”, “bmi”, “sex” and “marital status”. We observed that: -

We have implemented 5 models on the dataset considering “stroke” as the target variable. The models are implemented twice to acknowledge the impact of smoking-status of individuals over the chances of having a stroke. For this reason, we implemented models first where we have dropped the attribute “smoking status” and the other where smoking status is included into the list of feature columns.

|  |  |  |
| --- | --- | --- |
| **Model** | **Without considering Smoking-status** | **Considering Smoking Status** |
| K-Nearest Neighbor | 98.28% | 98.16% |
| Generalized Linear Model | 78.32% | 74.61% |
| Decision Tree | 98.35% | 98.20% |
| Random Forest | 98.32% | 98.15% |
| Gradient Boosting | 98.02% | 97.91% |

**Conclusion**

By running all the models, we figured out that the Generalized Linear Model is not suitable for prediction because it is not efficient while dealing with complex datasets. We also achieved almost similar accuracies while implementing the other models. We would suggest going with the random forest model because of its flexible nature, it can be used as a classifier as well as regressor. By analyzing the dataset, we can predict that 70-80 is the age that has high chances of getting a stroke whereas 10-20 age group doesn’t have high chances of stroke and the bmi range 25-30 is a critical range. Also, chances of stroke are more for married people as compared to unmarried people and chances of stroke in male is more as better as female.

From the obtained result, we can conclude that smoking status does not affect chances of acquiring stroke.

Comparing all the modules, smoking status is not a major issue for getting stroke. Also, important observations that need to be taken under consideration.

**References**

* A gentle introduction to XGBoost for applied machine learning. (2020, April 21). Machine Learning Mastery. <https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/>
* W3cubDocs. (n.d.). Retrieved October 20, 2020, from <https://docs.w3cub.com/statsmodels/generated/statsmodels.genmod.generalized_linear_model.glm.predict/>
* Yadav, D. (2019, December 09). Categorical encoding using Label-Encoding and One-Hot-Encoder. Retrieved October 20, 2020, from <https://towardsdatascience.com/categorical-encoding-using-label-encoding-and-one-hot-encoder-911ef77fb5bd>
* Donges, N. (n.d.). A complete guide to the random forest algorithm. Retrieved October 20, 2020, from <https://builtin.com/data-science/random-forest-algorithm>