REAL ESTATE PRICE PREDICTION

Ashish Singh[1], Jayanth Vishal Reddy[2], Vipin Kumar[3]

[1]Lovely Professional University, Phagwara, Punjab, 144411, India
[2]Lovely Professional University, Phagwara, Punjab, 144411, India
[3]Assistant Professor, Dept. of Computer Science and Engineering, Lovely Professional University, Punjab, India

---***---

Abstract - *Home prices are a really important indicator of the economic process, and land prices there's tons of interest from both buyers and sellers. during which Assignment or project. Home prices are estimated Detailed variables that contain multiple elements of Residential accommodation. As house prices continue, they will be assessed in several ways Lasso, SVM regression[3], rectilinear regression, and decision tree [4]; According to personal prices, they're Divided into methods that are evaluated in Innocent Bayes, including property classification SVM Division, and Random Forest Planning. We also run PCA to enhance Predictive accuracy. the aim of this project Retrospective model and make one Precisely enabled subdivision model to live the worth of a given home. this may help us to shop for land profitably.*

Key Words: (SVM regression, decision tree, supervised learning, linear regression, Lasso Regression, data visualization, sklearn, relapse model)

1. INTRODUCTION

The dataset is that the prices and features of residential houses in Bangalore, India, obtained from Kaggle. This dataset consists of 9 columns and 13320 rows, it comprises 9 characteristics like the world of the important estate, sorts of floors, the format of the house (2BHK and 3BHK), and numbers of bathrooms. Such high numbers of characteristics allow us to gauge several methods of forecasting land prices. The dataset consists of characteristics in several arrangements. It holds numerical data like prices and numbers of baths/rooms/family room, as strongly as categorical features like 'area type', 'location', 'society', 'Availability'. To execute this data with several arrangements available for our algorithms.

2. Models

We will do three types of supervised learning [5] algorithms such as linear regression, lasso, and decision tree. While it appears feasible to make a retreat as real estate prices continue, categorizing house prices by price will additionally provide practical insight for consumers, and also, this helps us to explore a variety of strategies that may be retrospective-or aligned with certain divisions. With 288 features in the database, adjustments are needed to prevent overheating. In order to determine the standard deviation parameter, for each project in both partition and subdivision, we will first perform K-fold cross verification with k = 5 in most types of custom parameter selection; this has helped us to select common rescue parameters in the training phase. In order to further improve our models, we also make a key core pipeline for all models and mix up a guaranteed number of elements that fit each model to deliver optimized results.

2.1 Classification

Home costs are grouped into value cans. In view of the house value dispersion in the informational collection, the value cans are as per the following: [0, 100k), [100k, 150k), [150k, 200k), [200k, 250k), [250k, 300k,] 300k, 350k), [350k, and), and we need to do a multi-class grouping to appraise the house costs in these seven cans. The exhibition of each model can be characterized by the precision rate, which is the quantity of experiments accurately grouped over the all out number of cases.

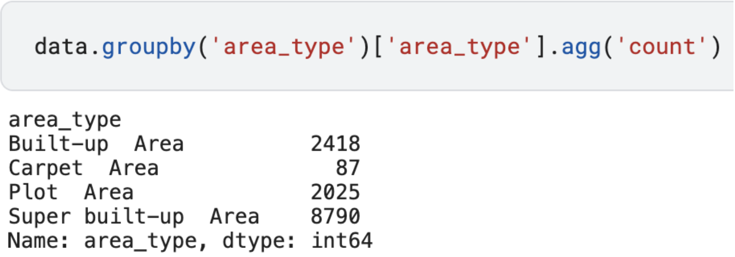
2.2 Models and Results

Accuracy while the multinomial nav base has 51% accuracy. Alternatively, the Gaussian Innocent Bayes model also performed better than the random estimate (14% or 1/7 with 7 price bucks). Furthermore, it is good to see how the multinomial nave bases are declining, we index them on the value buckets according to their commands and calculate the average absolute difference between the indexed indices of all cases and the computed indices. Root Mean Square may be an error. The mean difference of the multinomial innocent bale was 0.689, i.e. the median sequences were incorrectly modeled less than 1.

To improve our scientific categorization, we went to Multinomial Calculated Relapse in the equivalent dataset. We tuned the L2 regularization boundaries utilizing multiple times cross approval (we will fix this later with more subtleties); We additionally set up a blockade in the offices. Notwithstanding, its exhibition is really like that of the worldwide nave base; It has half exactness contrasted with 51% multinomial blameless inlet. Subsequent to tuning the boundaries, it was tracked down that the presentation of both Nive Bayes and Multinomial Calculated Relapse was at half. We kept on investigating different models for our multiscale arrangement. An option is the Help Vector Machine Arrangement (SVC), and we picked the Gaussian part alongside the straight portion. Like multimonial calculated relapse, we added the L2 regularization boundary and tuned it utilizing cross changeability. We tracked down that the SVC with the straight bit outflanked our past model with 63% exactness, while the SVC with the Gaussian bit had just 41% precision. At last, our last decision of scientific classification model is irregular backwoods scientific categorization. A significant boundary to power over-fitting is the greatest profundity at which we permit trees to develop; subsequently, we performed cross-approval to tune these most extreme profundity measures for regularization, like the multinomial strategic relapse and SVC's L2 regularization boundaries. In the wake of tuning, we accomplish 67% precision, which is like SVC with direct bits. Up until this point, SVC has performed best with 67% exactness, straight portion and irregular timberland grouping.

3. Regression

Before we fit into the relapse model, we pre-handled the information with a log-transformer on the contorted highlights, including the objective variable deal cost, to have a typical appropriation. We use groupby() work which is utilized to part the information into bunches dependent on certain rules and agg() work shortened form of total is utilized to characterize how we need to manage the gathered information.



```
data.groupby('area_type')['area_type'].agg('count')
```

```
area_type
Built-up  Area          2418
Carpet  Area              87
Plot  Area              2025
Super built-up  Area    8790
Name: area_type, dtype: int64
```

Fig -1: groupby( ) function

Then we drop NA values and useless columns and write a function to convert values into flot types.

3.1 Models and Results

For the relapse model [8], we will attempt to take care of the accompanying issue: In light of the rundown of handled highlights for the house, we need to appraise its potential deal cost. A direct relapse is a characteristic option in contrast to the standard model for relapse issues. Subsequently, we originally executed direct relapse 28 including all highlights utilizing 288 highlights and 1000 preparing models. This model is utilized to appraise the deal costs of homes given in our test information and contrasted with the genuine deal costs of homes given in the test informational collection. Results The wellspring of the outcomes acquired is the normal square blunder (rmse) and the presentation is estimated by the real outcomes. Our standard model created 0.5501 breaks. Since the objective variable is the log-transformer before the deal value model fitting, the subsequent rmse relies upon the distinction in log-changed selling costs, which are little qualities of rmse for the relapse model.

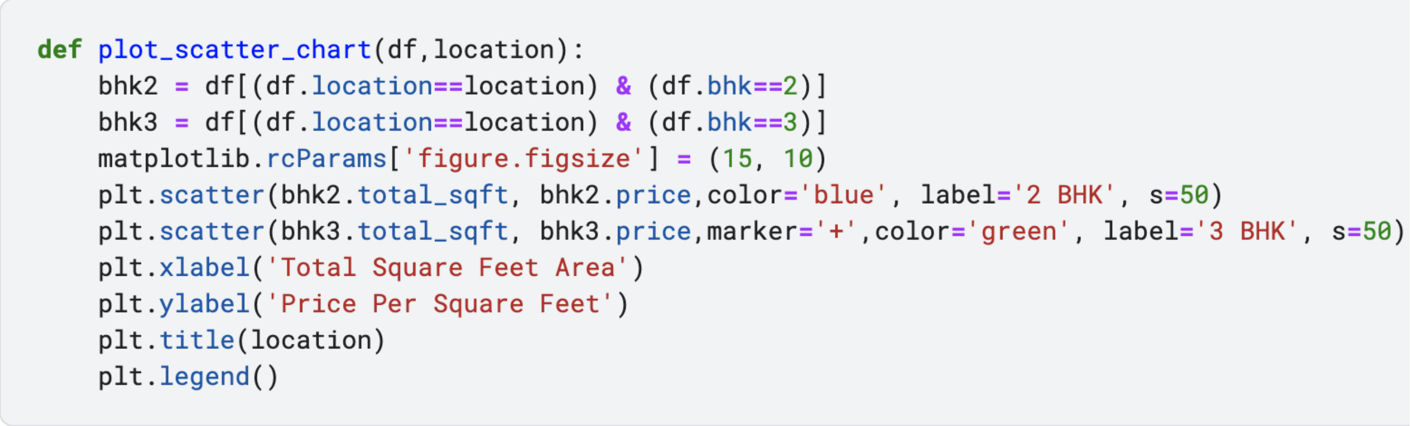
After utilizing the direct relapse model as the benchmark model, we added regularization boundaries to the straight relapse model to lessen overfitting. Direct relapse with Tether after multiple times cross-confirmation gave 0.5418 hole, which is superior to our pipeline model. Moreover, Straight Relapse with Rope naturally chooses 110 factors and eliminates the other 178 factors to suit the mode. Plat and Tether on the chose model will be combined with their loads To a limited extent 6 of the standard model.

Notwithstanding the Tether Regularizer, we likewise applied the Edge Regularizer with cross-approval to our standard relapse model, which delivers a REM of 0.5448. This rmse is far better than our pattern model, which implies normal direct relapse overfitting.

Backing vector relapse (SVR) with Gaussian and straight parts is additionally incorporated into the highlights. The CS boundary of the two models is approved to choose the best exhibition boundaries. The SVR created a 0.5271 shaft with the Gaussian piece model, and the straight part delivered a 5.503 higher pillar. The SVR with the Gaussian bit is 4% better than our gauge model. Since the piece is ineligible with the dataset, the SVR with the direct portion delivers generally high rmse for this situation.

3. Data Visualization

Information perception is the graphical portrayal of data and information. By utilizing visual components like diagrams, charts, and guides, information perception instruments give an open method to see and get patterns, exceptions, and examples in information. Time to Visualize Data for that we write a function that will help us to visualize the data of the different locations. To visualise the data we write a function plot_scatter_chart.



```
def plot_scatter_chart(df,location):
    bhk2 = df[(df.location==location) & (df.bhk==2)]
    bhk3 = df[(df.location==location) & (df.bhk==3)]
    matplotlib.rcParams['figure.figsize'] = (15, 10)
    plt.scatter(bhk2.total_sqft, bhk2.price,color='blue', label='2 BHK', s=50)
    plt.scatter(bhk3.total_sqft, bhk3.price,marker='+',color='green', label='3 BHK', s=50)
    plt.xlabel('Total Square Feet Area')
    plt.ylabel('Price Per Square Feet')
    plt.title(location)
    plt.legend()
```

The Fig -2 Shows Price per square Feet vs number of houses.

The Fig -3 Number of Bathrooms vs number of houses.

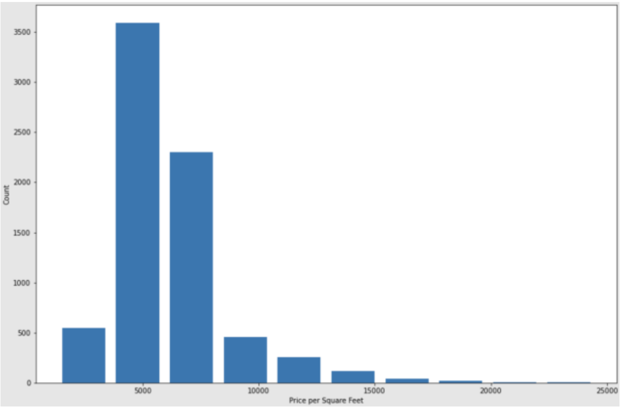


Fig -2: Price Per Square Feet

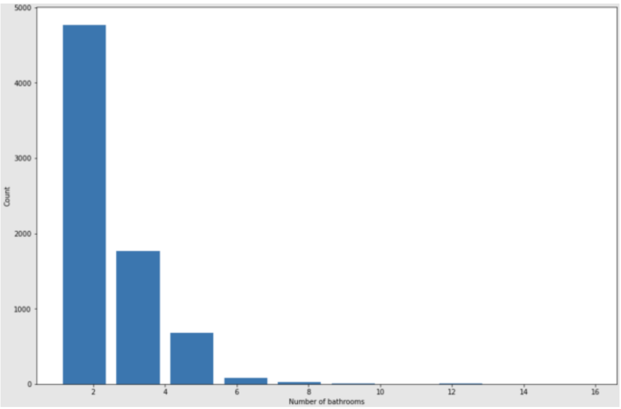


Fig -3: Number of Bathrooms

This graph shows no of houses with number of bathrooms. As we clearly see no of houses is more with 2 bathrooms.

Common general types of data visualization [6]:

- Charts
- Tables
- Graphs
- Maps
- Infographics
- Scatter

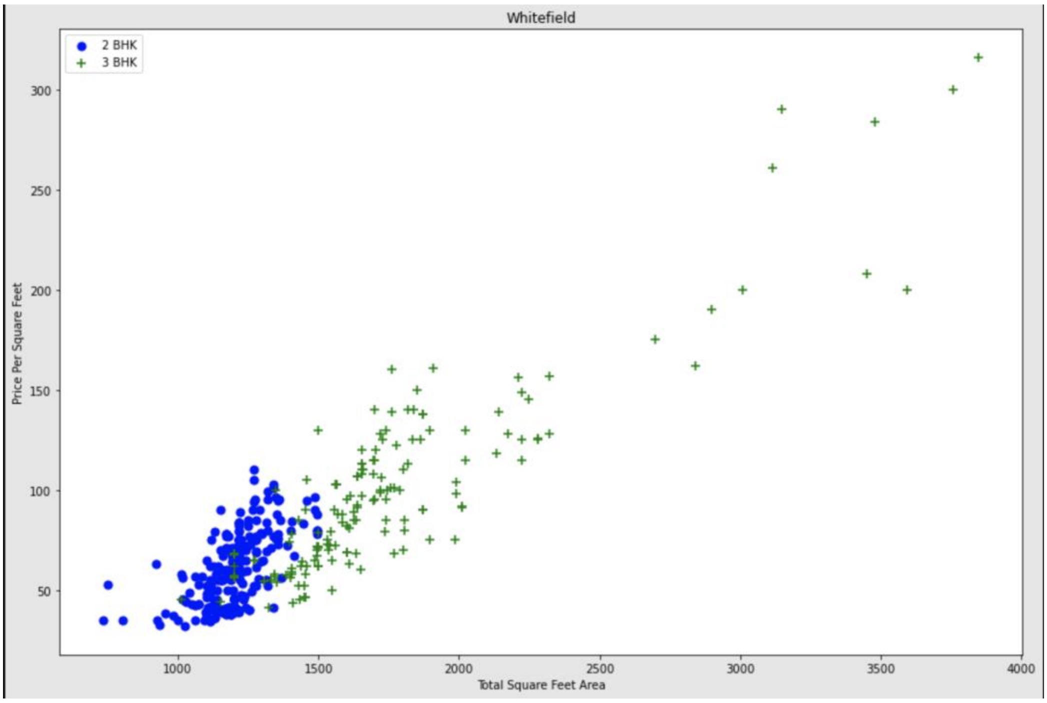


Fig -4: Total Square Feet Area vs Price per Square Feet in Whitefield.

Location :- Whitefield
By this we Clearly see that price of 3BHK is more that price of 2BHK because price per square feet in 3BHK is more than price per square feet in 2BHK that's why overall price is more than 2BHK.

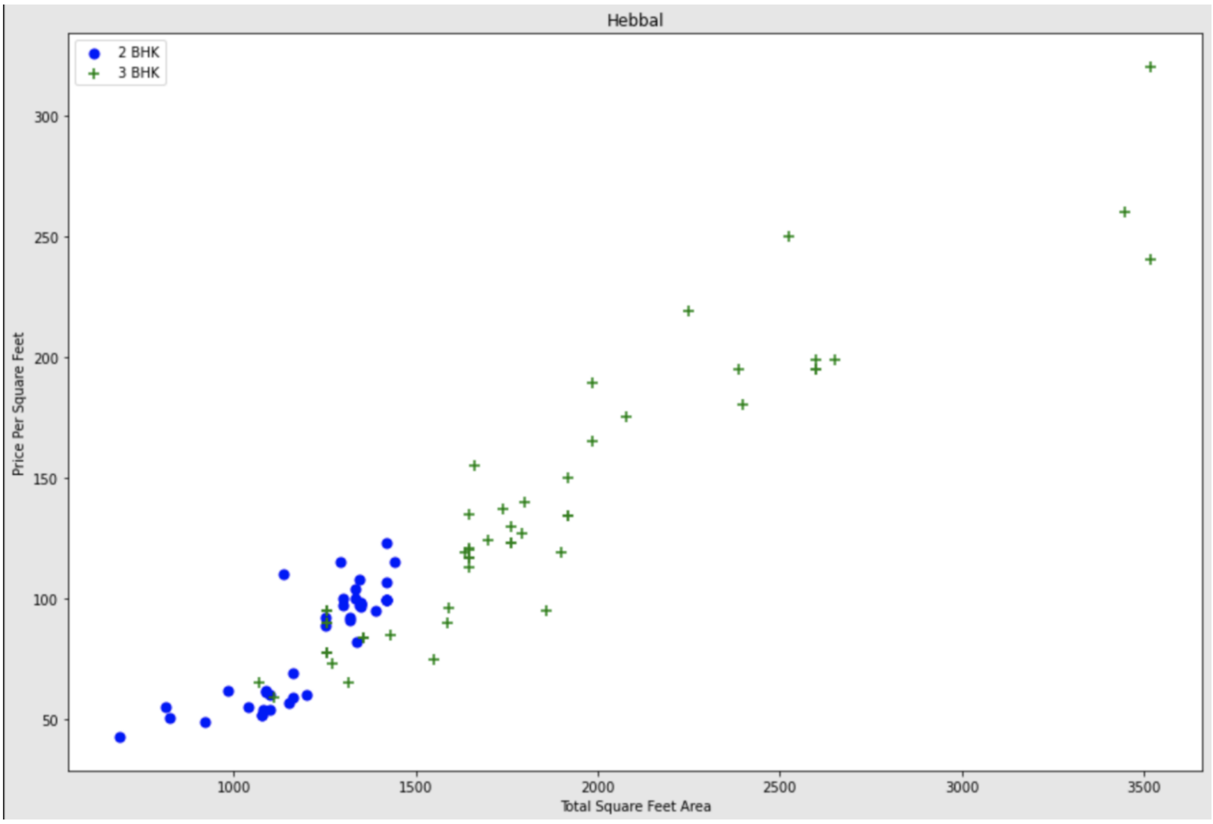


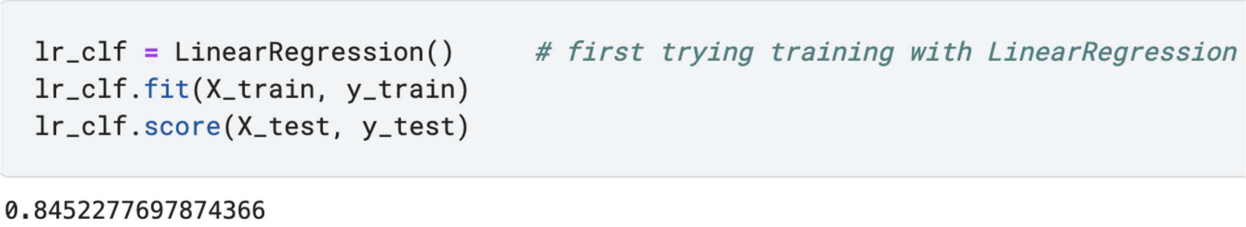
Fig -5: Total Square Feet Area vs Price per Square Feet in Hebbal.

In Hebbal 3BHK have more price than 2BHK and Number of 3BHK is also more than 2BHK.

4. Training and Testing

Before Training the model, it is required to split the data into train and test data. For this we will use, sklearn's [7] train_test_spilit which splits data into 80% and 20%. 80% for Training and 20% for testing.

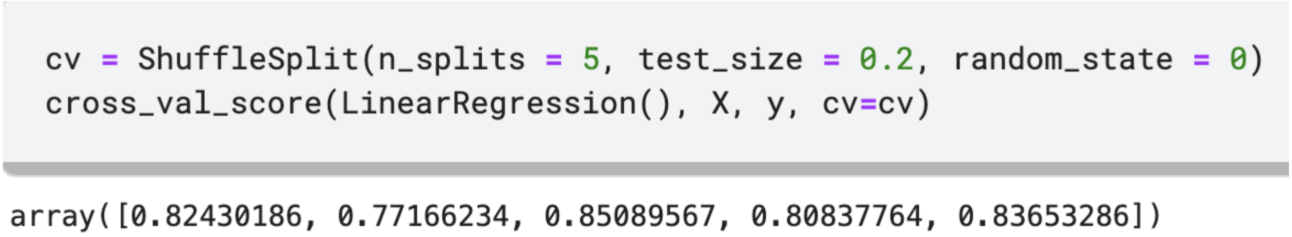
Preciesly, we will be trying more than one model, therefore, Training the Models



```
lr_clf = LinearRegression()     # first trying training with LinearRegression
lr_clf.fit(X_train, y_train)
lr_clf.score(X_test, y_test)
```

```
0.8452277697874366
```

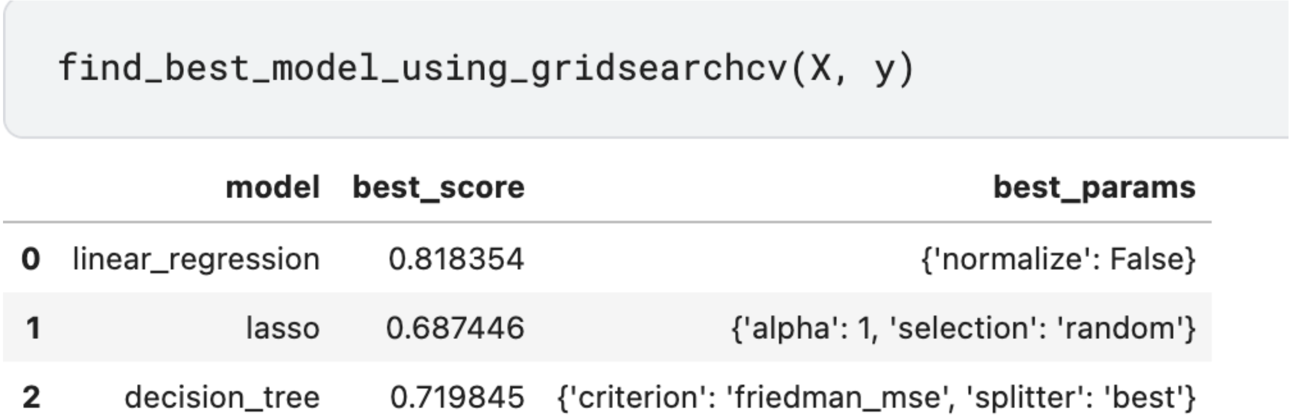
First we train with Linear Regression
With that we achieve 0.845 Score
We Use Cross Validation which is mainly used for the comparison of different models. For each model, we may get the average generalization error on the k validation sets. Then we will be able to choose the model with the lowest average generation error as our optimal model.



```
cv = ShuffleSplit(n_splits = 5, test_size = 0.2, random_state = 0)
cross_val_score(LinearRegression(), X, y, cv=cv)
```

```
array([0.82430186, 0.77166234, 0.85089567, 0.80837764, 0.83653286])
```

Then we use create a function "find_best_model_using_gridsearchcv" which include Linear Regression, Lasso and decision_tress for training.



```
find_best_model_using_gridsearchcv(X, y)
```

	model	best_score	best_params
0	linear_regression	0.818354	{'normalize': False}
1	lasso	0.687446	{'alpha': 1, 'selection': 'random'}
2	decision_tree	0.719845	{'criterion': 'friedman_mse', 'splitter': 'best'}

CONCLUSIONS

For the regression problem, the model that works best with linear regression [1] accuracy of 0.818354 is simple: false It is also the best of all other algorithms.

On the other hand, the Lasso Regression [2] model can provide information about selected properties, which can help us understand the relationship between home features and its selling prices.

According to our analysis, the number of square feet, bathrooms, and neighborhoods in Bangalore is of great statistical importance in estimating the selling price of a home.

REFERENCES

1. Weisberg, S., 2005. *Applied linear regression* (Vol. 528). John Wiley & Sons.
2. Ranstam, J. and Cook, J.A., 2018. LASSO regression. *Journal of British Surgery*, *105*(10), pp.1348-1348.
3. Flake, G.W. and Lawrence, S., 2002. Efficient SVM regression training with SMO. *Machine Learning*, *46*(1), pp.271-290.
4. Safavian, S.R. and Landgrebe, D., 1991. A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, *21*(3), pp.660-674.
5. Caruana, R. and Niculescu-Mizil, A., 2006, June. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning* (pp. 161-168).
6. Friendly, M., 2008. A brief history of data visualization. In *Handbook of data visualization* (pp. 15-56). Springer, Berlin, Heidelberg.
7. Jolly, K., 2018. Machine Learning with scikit-learn Quick Start Guide: Classification, regression, and clustering techniques in Python. Packt Publishing Ltd.
8. Marlatt, G.A. and George, W.H., 1984. Relapse prevention: Introduction and overview of the model. *British journal of addiction*, *79*(4), pp.261-273.