MACHINE LEARNING

PART PRESENTATION

CHOICE OF MODEL

SUPERVISED LEARNING-

Supervised learning deals with labeled data.

The reason to choose this is we have labeled data (njhome_floodsummary.csv)

We are Predicting 2011-2022 Precipitation(Anomaly_pct) vs House _Price drop

The Prediction is “Y” and “N”.

So its Binary Classification

FIRST SEGMENT-RUBRIC

(Decision for binary classification Algorithms)

REGRESSION-**regression** is used to predict continuous variables

LOGISTIC REGRESSION-**Logistic regression** predicts binary outcomes, meaning that there are only two possible outcomes.

CLASSIFICATION-**Classification**, on the other hand, is used to predict discrete outcomes

The outcome, in this case, is whether the person will vote "Yes" or "No." The classification model's algorithms would attempt to learn patterns from the data, and if the model is successful, gain the ability to make accurate predictions for new voters.

1.DECISSION TREE

2.RANDOM FOREST(ENSEMBLE)

SECOND SEGMENT-RUBRIC

(DECISION TO CREATE A MACHINE LEARNING WITH AVAILABLE DATA)

Converted all object variable to float/Integer

Result for all algorithm accuracy as below

1.logistic Regression gave 0.5988372093023255 accuracy

2.Decision Tree gave 0.7848837209302325

3.Random forest gave Accuracy Score : 1.0 -(which is to the perfection but in reality no data can be so perfect ,so ignoring this Model.

THIRD SEGMENT RUBRIC

THE .CSV was updated fro 2 CITIES AND 2 COUNTIES

The whole df(with X_train all 16 columns) was used

And it gave

Results for 3 ALGORITHMS as below-

1.LOGISTIC REGRESSION-0.5766423357664233

2.DECISION TREE-0.7883211678832117

3.RANDOM FOREST -1.0 (Using the equation (TP + TN) / Total, we can determine our accuracy)-but thisis unrealistic so droping this model

NOTHING CHANGED MUCH -and decided to go with DECISION TREE

X_train shape selection- when ran all columns

In [71]:
```
# We can sort the features by their importance.
sorted(zip(rf_model.feature_importances_, X.columns), reverse=True)
```

Out[71]:
```
[(0.7985322390260998, 'price_drop_amt'),
 (0.05593839549998928, 'year'),
 (0.03440773750384153, 'avghomeprice_month'),
 (0.02272429981367929, 'city_max_day_rain'),
 (0.021444610569882075, 'city_avg_daily_rain'),
 (0.020375421999981193, 'Anomalypct'),
 (0.019108115818803586, 'city_month_total_rain'),
 (0.012722273931325145, 'month'),
 (0.006012076323998551, 'SizeRank'),
 (0.0031114961841169977, 'zipcode'),
 (0.001027733731375137, 'CountyName'),
 (0.001006643330261235, 'LONGITUDE'),
 (0.0009991019917467102, 'ELEVATION'),
 (0.0009764576405017474, 'CITY'),
 (0.0008949748195014253, 'LATITUDE'),
 (0.0007184218148963946, 'Anomaly'),
 (0.0, 'State')]
```

To get better Algorithm results- X_train reframed

Rank the Importance of Features#

In [51]:
```
# Calculate feature importance in the Random Forest model.
importances = rf_model.feature_importances_
importances
```

Out[51]:
```
array([0.01233577, 0.01042029, 0.11632184, 0.24015627, 0.18402527,
       0.19950475, 0.23723581])
```

In [52]:
```
# We can sort the features by their importance.
sorted(zip(rf_model.feature_importances_, X.columns), reverse=True)
```

Out[52]:
```
[(0.24015627453179084, 'year'),
 (0.23723581138264804, 'avghomeprice_month'),
 (0.19950474894836076, 'Anomalypct'),
 (0.18402526930943405, 'city_month_total_rain'),
 (0.11632184064232612, 'month'),
 (0.012335769961870177, 'CITY'),
 (0.010420285223569928, 'ELEVATION')]
```

FINAL RUBRIC

The njhome_floodsummary csv was updated

and we got CURRENTACCURACY SCORE for VARIOUS alogorithms

by minimising/reframing on X_train.shape

X = X = df [['CITY', 'ELEVATION', 'month', 'year',

'city_month_total_rain',

'Anomalypct',

'avghomeprice_month']]-7 columns -projected in Feature importance

1.LOGISTIC REGRESSION-0.5912408759124088

2.DECISION TREE-0.8029197080291971

3.RANDOM FOREST -0.8321167883211679 (Using the equation (TP + TN) / Total, we can determine our accuracy)- this gave more accuracy m

The final algorithm for the project is RANDOM FOREST with highest accuracy.

MACHINE LEARNING FINAL RESULT

It showed that Random forest gave best accuracy result

If compared Final JUPYTER NOTEBOOK -ML_Pricedrop.ipynb

RANDOM FOREST and DECISION TREE showed same Features with different ratio of importance

The Random forest has almost the same hyperparameters as a decision tree. Its ensemble method of decision trees is generated on randomly split data.

-continued next page

RANDOM FOREST OVER DECISION TREE

DECISION TREE-

-A decision tree is a tree-like model of decisions along with possible outcomes.

-There is always a scope for overfitting, caused due to the presence of variance.

-The results are not accurate.

RANDOM FOREST-

-A classification algorithm consisting of many decision trees combined to get a more accurate result as compared to a single tree.

-Random forest algorithm avoids and prevents overfitting by using multiple trees.

-This gives accurate and precise results.

SO RANDOM FOREST BEST FOR THIS MODEL TO PREDICT PRICEDROP