# PRICE PREDICTION OF HOUSES USING REGRESSION TECHNIQUES

Under the guidance of:
Dr. V.K Jha

Presented By:
- Mohit Toppo (BE/10290/16)
- Pratyush Nigel Baxla (BE/10469/16)
- Animesh Purty (BE/10556/16)

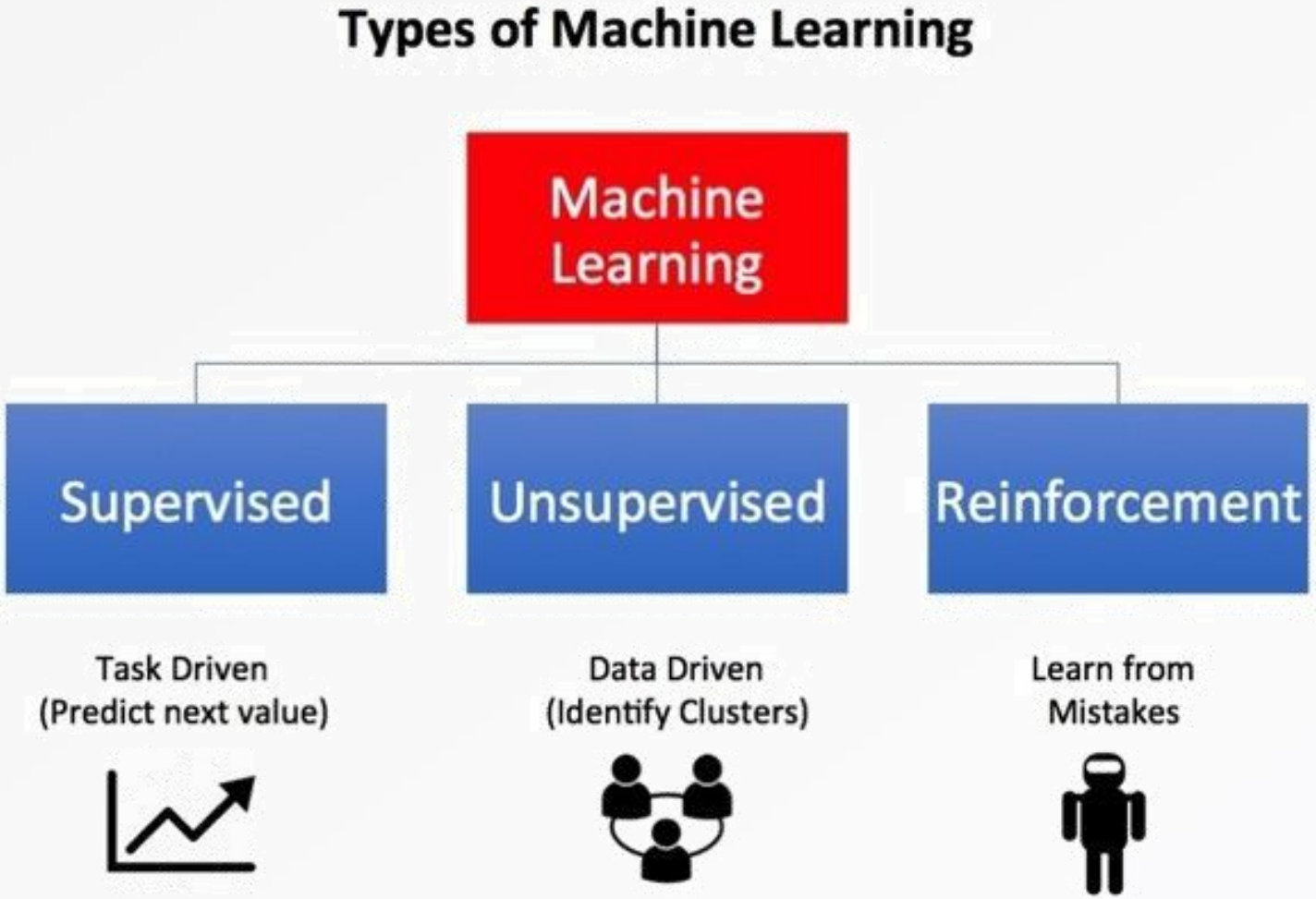
# Introduction

**Machine learning** (**ML**):

Machine learning is the science of getting computer to act without being explicitly programmed.Its a subset of Artificial Intellegence(AI).

**Machine Learning** has given us self-driving cars, practical speech recognition, effective web search, and a vastly improved understanding of the human genome.

# Types of Machine Learning



# Objective

In this project we have taken the Housing dataset which contains information about different houses in Bengaluru.

The **objective** is to build a model with the given 9 features(categorical and continuous) to predict the prices of the houses using Linear, Ridge and Lasso Regression Tecniques, check the accuracies and compare them.

**Features:**

1. Area_type – describes the area
2. Availability – when it can be possessed or when it is ready(categorical and time-series)
3. Location – where it is located in Bengaluru (Area name)
4. Size – in BHK or Bedroom (1-10 or more)
5. Society – to which society it belongs
6. Total_sqft – size of the property in sq.ft
7. Bath – No. of bathrooms
8. Balcony – No. of the balcony

# WORKING & METHODOLOGY

**Data Cleaning**

- Removal of unwanted observations
- Handling Missing Data
- Fixing Structural Errors

**Feature Engineering**

- Adding new features
- Dimensionality Reduction
- Dealing with categorical values

**Outlier Reduction**

- Removing of extreme values that deviate from other observations on data

**Building a model**

- Forming a Training and Testing Dataset

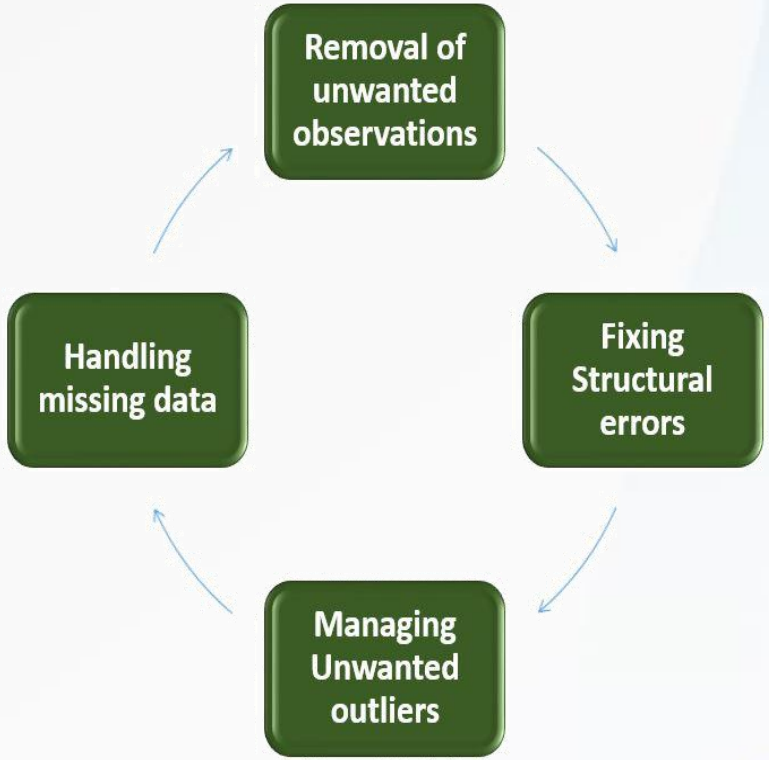
**Train the Model**

- Using Machine Learning Algorithm which in this case Multiple Linear Regression
- Based on the generated graphs, predict the prices of the house

# Data Cleaning

**Removal of unwanted observations** like:

- **Handling Missing Data**: Completely remove the row consisting of NA values or by taking the median of rest of the data.
- **Fixing Structural Errors**: Fixing Typos or Inconsistent capitalization
- **Managing Unwanted Outliers**: Removing Outliers i.e Extreame Values that are odd/not fitting in the general pattern of the dataset

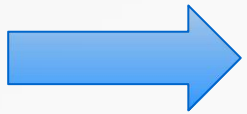


# Feature Engineering

**Feature Engineering** is the process of using domain knowledge to extract features from raw data via data mining techniques. These features can be used to improve the performance of machine learning algorithms

Ex: Adding new feature(integer) for bhk (Bedrooms Hall Kitchen) to remove inconsistencies

| size | total_sqft | bath | balcony | price |
|---|---|---|---|---|
| 2 BHK | 1056 | 2.0 | 1.0 | 39.07 |
| 4 Bedroom | 2600 | 5.0 | 3.0 | 120.00 |
| 3 BHK | 1440 | 2.0 | 3.0 | 62.00 |
| 3 BHK | 1521 | 3.0 | 1.0 | 95.00 |
| 2 BHK | 1200 | 2.0 | 1.0 | 51.00 |



| size | total_sqft | bath | balcony | price | bhk |
|---|---|---|---|---|---|
| 2 BHK | 1056 | 2.0 | 1.0 | 39.07 | 2 |
| 4 Bedroom | 2600 | 5.0 | 3.0 | 120.00 | 4 |
| 3 BHK | 1440 | 2.0 | 3.0 | 62.00 | 3 |
| 3 BHK | 1521 | 3.0 | 1.0 | 95.00 | 3 |
| 2 BHK | 1200 | 2.0 | 1.0 | 51.00 | 2 |

# Correlation Matrix

- we create a correlation matrix that measures the linear relationships between the variables. The correlation matrix can be formed by using the corr function from the pandas dataframe library. We will use the heatmap function from the seaborn library to plot the correlation matrix.


- The correlation coefficient ranges from -1 to 1. If the value is close to 1, it means that there is a strong positive correlation between the two variables. When it is close to -1, the variables have a strong negative correlation.

# Outlier Removal

- When modeling, it is important to clean the data sample to ensure that the observations best represent the problem.
- Sometimes a dataset can contain extreme values that are outside the range of what is expected and unlike the other data. These are called outliers and often machine learning modeling and model skill in general can be improved by understanding and even removing these outlier values.

**Outlier Removal Using Business Logic**

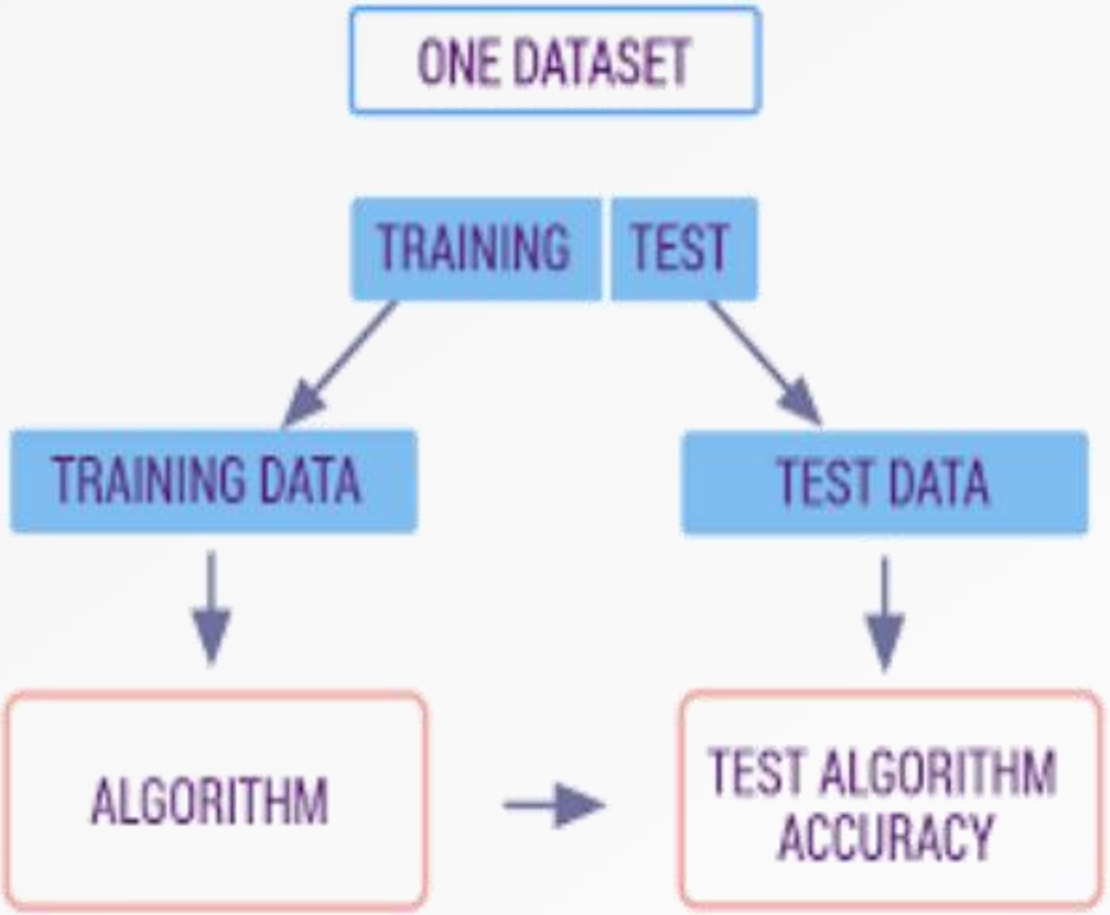
- As a data scientist when you have a conversation with your business manager (who has expertise in real estate), he will tell you that normally square ft per bedroom is 300.

  We will remove such outliers by keeping our minimum thresold per bhk to be 300 sqft

**Outlier Removal Using Standard Deviation and Mean**

- Here we find that min price per sqft is Rs.267 whereas max is Rs.1,20,00,000, this shows a wide variation in property prices. We should remove outliers per location using mean and one standard deviation

# Training And Testing Data



# Linear Regression

**Linear regression** is the simplest and most widely used statistical technique for predictive modeling. It basically gives us an equation, where we have our features as independent variables, on which our target variable is dependent upon.
Linear regression equation looks like this:

$$Y = \theta_1 X_1 + \theta_2 X_2 + \dots \theta_n X_n$$

Here, we have Y as our dependent variable, X's are the independent variables and all thetas are the coefficients. Coefficients are basically the weights assigned to the features, based on their importance.

# Ridge Regression

**Ridge Regression** is a technique used when the data suffers from multicollinearity (independent variables are highly correlated). In multicollinearity, even though the least squares estimates (OLS) are unbiased, their variances are large which deviates the observed value far from the true value. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

$$\text{The sum of the squared residuals} + \lambda \times \text{the slope}^2$$

**Important Points:**

- The assumptions of this regression is same as least squared regression except normality is not to be assumed
- Ridge regression shrinks the value of coefficients but doesn't reaches zero, which suggests no feature selection feature
- This is a regularization method and uses  l2 regularization

# Lasso Regression

**Lasso** performs a so called  l1 regularization(a process of introducing additional information in order to prevent overfitting), i.e. adds penalty equivalent to absolute value of the magnitude of coefficients.
In particular, the minimization objective does not only include the residual sum of squares (RSS) - like in the OLS regression setting - but also the sum of the absolute value of coefficients.

The sum of squared residuals + λ x |the slope|

- The assumptions of lasso regression is same as least squared regression except normality is not to be assumed
- Lasso Regression shrinks coefficients to zero (exactly zero), which certainly helps in feature selection
- Lasso is a regularization method and uses l1 regularization
- If group of predictors are highly correlated, lasso picks only one of them and shrinks the others to zero

# Comparision between Linear, Ridge and Lasso Regression

**Linear regression** is a simple technique to implement. In this technique, the dependent variable is continuous, independent variable(s) can be continuous or discrete, and nature of regression line is linear. Linear Regression establishes a relationship between dependent variable (Y) and one or more independent variables (X)using a best fit straight line (also known as regression line).

```
[80]: from sklearn.linear_model import LinearRegression
      lin_model=LinearRegression()
      lin_model.fit(X_train, Y_train)
      lin_model.score(X_test, Y_test)
```

```
[80]: 0.8483032790645954
```

**Ridge Regression** is a technique used when the data suffers from multicollinearity ( independent variables are highly correlated). In multicollinearity, even though the least squares estimates (OLS) are unbiased, their variances are large which deviates the observed value far from the true value. By adding a degree of bias to the regression estimates, ridge regression reduces the standard errors.

```
[106]: from sklearn.linear_model import Ridge
       rr = Ridge(alpha=0.01)
       rr.fit(X_train, Y_train)
       rr.score(X_test, Y_test)
       #(0.001, 0.01, 0.1, 0.5, 1, 2, 10)
```

```
[106]: 0.8482536690954904
```

**Lasso (Least Absolute Shrinkage and Selection Operator)**differs from ridge regression in a way that it uses absolute values in the penalty function, instead of squares. This leads to penalizing values which causes some of the parameter estimates to turn out exactly zero. Larger the penalty applied, further the estimates get shrunk towards absolute zero. This results to variable selection out of given n variables.

```
[115]: from sklearn.linear_model import Lasso
       model_lasso = Lasso(alpha=0.1)
       model_lasso.fit(X_train, Y_train)
       model_lasso.score(X_test, Y_test)
```

```
[115]: 0.8079223356257808
```

# Conclusion

The results shows that even though **Linear Regression** and **Ridge Regression** having same accuracy, **84.83%** and **84.82%** respectively, predict different prices for same property with just slight difference from each other. While on the other hand **Lasso Regression** having accuracy of **80.79%** predict a very different prices, that too with large differences with respect to other regression techniques.

| | Actual Price | linear predicted price | ridge predicted price | lasso predicted price |
|---|---|---|---|---|
| **8661** | 82.0 | 123.299652 | 123.680680 | 119.981175 |
| **9118** | 200.0 | 184.952953 | 186.266744 | 182.897526 |
| **9464** | 63.0 | 69.070756 | 66.826242 | 60.559468 |
| **3626** | 79.0 | 73.764612 | 73.093385 | 65.428607 |
| **4102** | 130.0 | 139.572500 | 142.460641 | 149.410719 |
| **9109** | 165.0 | 95.907371 | 94.178321 | 87.994527 |
| **6527** | 60.0 | 68.885711 | 71.155844 | 75.222497 |
| **8892** | 156.0 | 93.543293 | 94.053794 | 91.596742 |
| **901** | 69.0 | 61.959328 | 62.151068 | 67.162682 |
| **10220** | 101.0 | 122.532451 | 123.619941 | 122.431356 |

# Future Scope

- Increasing and improving data set.
- Changing the dataset and finding the better topic model according to new dataset.
- Implementing different topic models.
- It can be used in Automobile Industries, Real Estate, Travel Agencies etc.

# Thank You!