**Introduction**

An accurate prediction on the house price is important to prospective homeowners, developers, investors, appraisers, tax assessors and other real estate market participants, such as, mortgage lenders and insurers.

**Dataset**

Dataset consists of historical house prices of residential homes in Ames, Iowa, with a total of 81 exploratory features and 1460 observations. The dataset is extracted from Kaggle website.

<https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data>

The data set contains every minute detail of the house. Some of the major features in this data set are:

1. Lot Area
2. Neighborhood
3. House Style
4. Quality of the house
5. Overall condition of the house
6. Year built
7. Year remodeled
8. Foundation
9. Basement Condition
10. Total basement square feet
11. 1st floor square feet
12. 2nd floor square feet
13. Above ground living area in square feet
14. Full bathrooms above ground
15. Bedrooms above grade
16. Total rooms above grade
17. Garage size in square feet
18. Garage quality

However, it is good idea to explore the data set from Kaggle to get good idea on the data.

**Data Wrangling**

Data Wrangling is an extremely important step for any data analysis. It is very crucial for data to be organized. This process typically includes manually converting/mapping data from one raw form into another format to allow for more convenient consumption and organization of the data.

Data Cleaning steps carried out in this project are:

1. Handling missing data
2. Handling inconsistent data in a few variables

House Prices data set information:

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 1460 entries, 0 to 1459

Data columns (total 81 columns):

Id 1460 non-null int64

MSSubClass 1460 non-null int64

MSZoning 1460 non-null object

LotFrontage 1201 non-null float64

LotArea 1460 non-null int64

Street 1460 non-null object

Alley 91 non-null object

LotShape 1460 non-null object

LandContour 1460 non-null object

Utilities 1460 non-null object

LotConfig 1460 non-null object

LandSlope 1460 non-null object

Neighborhood 1460 non-null object

Condition1 1460 non-null object

Condition2 1460 non-null object

BldgType 1460 non-null object

HouseStyle 1460 non-null object

OverallQual 1460 non-null int64

OverallCond 1460 non-null int64

YearBuilt 1460 non-null int64

YearRemodAdd 1460 non-null int64

RoofStyle 1460 non-null object

RoofMatl 1460 non-null object

Exterior1st 1460 non-null object

Exterior2nd 1460 non-null object

MasVnrType 1452 non-null object

MasVnrArea 1452 non-null float64

ExterQual 1460 non-null object

ExterCond 1460 non-null object

Foundation 1460 non-null object

BsmtQual 1423 non-null object

BsmtCond 1423 non-null object

BsmtExposure 1422 non-null object

BsmtFinType1 1423 non-null object

BsmtFinSF1 1460 non-null int64

BsmtFinType2 1422 non-null object

BsmtFinSF2 1460 non-null int64

BsmtUnfSF 1460 non-null int64

TotalBsmtSF 1460 non-null int64

Heating 1460 non-null object

HeatingQC 1460 non-null object

CentralAir 1460 non-null object

Electrical 1459 non-null object

1stFlrSF 1460 non-null int64

2ndFlrSF 1460 non-null int64

LowQualFinSF 1460 non-null int64

GrLivArea 1460 non-null int64

BsmtFullBath 1460 non-null int64

BsmtHalfBath 1460 non-null int64

FullBath 1460 non-null int64

HalfBath 1460 non-null int64

BedroomAbvGr 1460 non-null int64

KitchenAbvGr 1460 non-null int64

KitchenQual 1460 non-null object

TotRmsAbvGrd 1460 non-null int64

Functional 1460 non-null object

Fireplaces 1460 non-null int64

FireplaceQu 770 non-null object

GarageType 1379 non-null object

GarageYrBlt 1379 non-null float64

GarageFinish 1379 non-null object

GarageCars 1460 non-null int64

GarageArea 1460 non-null int64

GarageQual 1379 non-null object

GarageCond 1379 non-null object

PavedDrive 1460 non-null object

WoodDeckSF 1460 non-null int64

OpenPorchSF 1460 non-null int64

EnclosedPorch 1460 non-null int64

3SsnPorch 1460 non-null int64

ScreenPorch 1460 non-null int64

PoolArea 1460 non-null int64

PoolQC 7 non-null object

Fence 281 non-null object

MiscFeature 54 non-null object

MiscVal 1460 non-null int64

MoSold 1460 non-null int64

YrSold 1460 non-null int64

SaleType 1460 non-null object

SaleCondition 1460 non-null object

SalePrice 1460 non-null int64

dtypes: float64(3), int64(35), object(43)

The output above is produced from **info()** function. There are a few categorical and numerical variables with missing values.

1. **Handling Missing Data:**

* **Categorical Data:** The categorical variables with missing values are ‘MasVnrType’ and ‘Electrical’. Python provides many methods like fillna, forward/ backward filling, dropna etc. for handling missing data. I introduced another category called ‘**missing**’ to all the null values. This way I am retaining the original information of the data and not guessing anything.
* **Numerical Data:** The most popular method to handle missing numerical data is **Mean Imputation**. I applied the same on my numerical data. Mean imputation is a method in which the missing value on a certain variable is replaced by the mean of the available cases. This is a reliable method for handling missing numerical data.

1. **Handling inconsistent data:**

There are a few null values in the data set which are not actually nulls but are entered wrongly as nulls. Referring to the actual data set description file (data\_description.txt) from Kaggle, a few values were coded as ‘NA’ if a feature was not present in the house, but these NA values were entered as Nan in the .csv file. I decoded these misinterpreted values as ‘No feature\_name’ (feature\_name being name of the feature not present in the house).