Project Name: House Prices: Advanced Regression Techniques

Problem Statement

Ask a home buyer to describe their dream house, and they probably won't begin with the height of the basement ceiling or the proximity to an east-west railroad.But this playground competition's dataset proves that much more influences price negotiations than the number of bedrooms or a white-picket fence.

With 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, this competition challenges you to predict the final price of each home.

The main aim of this project is to predict the house price based on various features which we will discuss as we go ahead

Dataset to downloaded from the below link

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

```
##3 Importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

pd.pandas.set_option('display.max_columns',None)

train = pd.read_csv("train.csv")
test = pd.read_csv("test.csv")
df_train = pd.concat((train, test))
df_train.head()
```

```
   Id  MSSubClass MSZoning  LotFrontage  LotArea Street Alley LotShape  \
0   1          60       RL         65.0     8450   Pave   NaN      Reg

1   2          20       RL         80.0     9600   Pave   NaN      Reg

2   3          60       RL         68.0    11250   Pave   NaN      IR1

3   4          70       RL         60.0     9550   Pave   NaN      IR1

4   5          60       RL         84.0    14260   Pave   NaN      IR1


  LandContour Utilities LotConfig LandSlope Neighborhood Condition1  \
0         Lvl    AllPub    Inside       Gtl      CollgCr       Norm
1         Lvl    AllPub       FR2       Gtl      Veenker      Feedr
2         Lvl    AllPub    Inside       Gtl      CollgCr       Norm
3         Lvl    AllPub    Corner       Gtl      Crawfor       Norm
```

```
4         Lvl    AllPub       FR2       Gtl      NoRidge       Norm

  Condition2 BldgType HouseStyle  OverallQual  OverallCond  YearBuilt
\
0       Norm     1Fam     2Story            7            5       2003

1       Norm     1Fam     1Story            6            8       1976

2       Norm     1Fam     2Story            7            5       2001

3       Norm     1Fam     2Story            7            5       1915

4       Norm     1Fam     2Story            8            5       2000


   YearRemodAdd RoofStyle RoofMatl Exterior1st Exterior2nd MasVnrType
\
0          2003     Gable  CompShg     VinylSd     VinylSd    BrkFace

1          1976     Gable  CompShg     MetalSd     MetalSd       None

2          2002     Gable  CompShg     VinylSd     VinylSd    BrkFace

3          1970     Gable  CompShg     Wd Sdng     Wd Shng       None

4          2000     Gable  CompShg     VinylSd     VinylSd    BrkFace


   MasVnrArea ExterQual ExterCond Foundation BsmtQual BsmtCond
BsmtExposure  \
0       196.0        Gd        TA      PConc       Gd       TA
No
1         0.0        TA        TA     CBlock       Gd       TA
Gd
2       162.0        Gd        TA      PConc       Gd       TA
Mn
3         0.0        TA        TA     BrkTil       TA       Gd
No
4       350.0        Gd        TA      PConc       Gd       TA
Av

  BsmtFinType1  BsmtFinSF1 BsmtFinType2  BsmtFinSF2  BsmtUnfSF
TotalBsmtSF  \
0          GLQ       706.0          Unf         0.0      150.0
856.0
1          ALQ       978.0          Unf         0.0      284.0
1262.0
2          GLQ       486.0          Unf         0.0      434.0
920.0
```

```
3          ALQ       216.0         Unf         0.0      540.0
756.0
4          GLQ       655.0         Unf         0.0      490.0
1145.0

  Heating HeatingQC CentralAir Electrical  1stFlrSF  2ndFlrSF
LowQualFinSF  \
0    GasA        Ex          Y      SBrkr       856       854
0
1    GasA        Ex          Y      SBrkr      1262         0
0
2    GasA        Ex          Y      SBrkr       920       866
0
3    GasA        Gd          Y      SBrkr       961       756
0
4    GasA        Ex          Y      SBrkr      1145      1053
0

   GrLivArea  BsmtFullBath  BsmtHalfBath  FullBath  HalfBath
BedroomAbvGr  \
0       1710           1.0           0.0         2         1
3
1       1262           0.0           1.0         2         0
3
2       1786           1.0           0.0         2         1
3
3       1717           1.0           0.0         1         0
3
4       2198           1.0           0.0         2         1
4

   KitchenAbvGr KitchenQual  TotRmsAbvGrd Functional  Fireplaces
FireplaceQu  \
0             1          Gd             8        Typ           0
NaN
1             1          TA             6        Typ           1
TA
2             1          Gd             6        Typ           1
TA
3             1          Gd             7        Typ           1
Gd
4             1          Gd             9        Typ           1
TA

  GarageType  GarageYrBlt GarageFinish  GarageCars  GarageArea
GarageQual  \
0     Attchd       2003.0          RFn         2.0       548.0
TA
1     Attchd       1976.0          RFn         2.0       460.0
TA
```

```
2     Attchd       2001.0          RFn         2.0       608.0
TA
3     Detchd       1998.0          Unf         3.0       642.0
TA
4     Attchd       2000.0          RFn         3.0       836.0
TA

  GarageCond PavedDrive  WoodDeckSF  OpenPorchSF  EnclosedPorch
3SsnPorch  \
0         TA          Y           0           61              0
0
1         TA          Y         298            0              0
0
2         TA          Y           0           42              0
0
3         TA          Y           0           35            272
0
4         TA          Y         192           84              0
0

   ScreenPorch  PoolArea PoolQC Fence MiscFeature  MiscVal  MoSold
YrSold  \
0            0         0    NaN   NaN         NaN        0       2
2008
1            0         0    NaN   NaN         NaN        0       5
2007
2            0         0    NaN   NaN         NaN        0       9
2008
3            0         0    NaN   NaN         NaN        0       2
2006
4            0         0    NaN   NaN         NaN        0      12
2008

  SaleType SaleCondition  SalePrice
0       WD        Normal   208500.0
1       WD        Normal   181500.0
2       WD        Normal   223500.0
3       WD       Abnorml   140000.0
4       WD        Normal   250000.0
```

```
df_train.tail()
```

```
        Id  MSSubClass MSZoning  LotFrontage  LotArea Street Alley
LotShape  \
1454  2915         160       RM         21.0     1936   Pave   NaN
Reg
1455  2916         160       RM         21.0     1894   Pave   NaN
Reg
1456  2917          20       RL        160.0    20000   Pave   NaN
Reg
```

```
1457  2918          85       RL          62.0    10441   Pave   NaN
Reg
1458  2919          60       RL          74.0     9627   Pave   NaN
Reg

     LandContour Utilities LotConfig LandSlope Neighborhood Condition1
\
1454         Lvl    AllPub    Inside       Gtl      MeadowV       Norm

1455         Lvl    AllPub    Inside       Gtl      MeadowV       Norm

1456         Lvl    AllPub    Inside       Gtl      Mitchel       Norm

1457         Lvl    AllPub    Inside       Gtl      Mitchel       Norm

1458         Lvl    AllPub    Inside       Mod      Mitchel       Norm


     Condition2 BldgType HouseStyle  OverallQual  OverallCond
YearBuilt  \
1454       Norm    Twnhs     2Story            4            7
1970
1455       Norm   TwnhsE     2Story            4            5
1970
1456       Norm     1Fam     1Story            5            7
1960
1457       Norm     1Fam     SFoyer            5            5
1992
1458       Norm     1Fam     2Story            7            5
1993

      YearRemodAdd RoofStyle RoofMatl Exterior1st Exterior2nd
MasVnrType  \
1454          1970     Gable  CompShg     CemntBd     CmentBd
None
1455          1970     Gable  CompShg     CemntBd     CmentBd
None
1456          1996     Gable  CompShg     VinylSd     VinylSd
None
1457          1992     Gable  CompShg     HdBoard     Wd Shng
None
1458          1994     Gable  CompShg     HdBoard     HdBoard
BrkFace

      MasVnrArea ExterQual ExterCond Foundation BsmtQual BsmtCond  \
1454         0.0        TA        TA     CBlock       TA       TA
1455         0.0        TA        TA     CBlock       TA       TA
1456         0.0        TA        TA     CBlock       TA       TA
1457         0.0        TA        TA      PConc       Gd       TA
```

```
1458        94.0        TA        TA      PConc       Gd       TA

     BsmtExposure BsmtFinType1  BsmtFinSF1 BsmtFinType2  BsmtFinSF2  \
1454           No          Unf         0.0          Unf         0.0
1455           No          Rec       252.0          Unf         0.0
1456           No          ALQ      1224.0          Unf         0.0
1457           Av          GLQ       337.0          Unf         0.0
1458           Av          LwQ       758.0          Unf         0.0

      BsmtUnfSF  TotalBsmtSF Heating HeatingQC CentralAir
Electrical  \
1454      546.0        546.0    GasA        Gd          Y      SBrkr

1455      294.0        546.0    GasA        TA          Y      SBrkr

1456        0.0       1224.0    GasA        Ex          Y      SBrkr

1457      575.0        912.0    GasA        TA          Y      SBrkr

1458      238.0        996.0    GasA        Ex          Y      SBrkr


      1stFlrSF  2ndFlrSF  LowQualFinSF  GrLivArea  BsmtFullBath
BsmtHalfBath  \
1454       546       546             0       1092           0.0
0.0
1455       546       546             0       1092           0.0
0.0
1456      1224         0             0       1224           1.0
0.0
1457       970         0             0        970           0.0
1.0
1458       996      1004             0       2000           0.0
0.0

      FullBath  HalfBath  BedroomAbvGr  KitchenAbvGr KitchenQual  \
1454         1         1             3             1          TA
1455         1         1             3             1          TA
1456         1         0             4             1          TA
1457         1         0             3             1          TA
1458         2         1             3             1          TA

      TotRmsAbvGrd Functional  Fireplaces FireplaceQu GarageType
GarageYrBlt  \
1454             5        Typ           0         NaN        NaN
NaN
1455             6        Typ           0         NaN    CarPort
1970.0
1456             7        Typ           1          TA     Detchd
```

```
1960.0
1457             6        Typ           0         NaN        NaN
NaN
1458             9        Typ           1          TA     Attchd
1993.0

     GarageFinish  GarageCars  GarageArea GarageQual GarageCond
PavedDrive  \
1454          NaN         0.0         0.0        NaN        NaN
Y
1455          Unf         1.0       286.0         TA         TA
Y
1456          Unf         2.0       576.0         TA         TA
Y
1457          NaN         0.0         0.0        NaN        NaN
Y
1458          Fin         3.0       650.0         TA         TA
Y

      WoodDeckSF  OpenPorchSF  EnclosedPorch  3SsnPorch
ScreenPorch  \
1454           0            0              0          0            0

1455           0           24              0          0            0

1456         474            0              0          0            0

1457          80           32              0          0            0

1458         190           48              0          0            0


      PoolArea PoolQC  Fence MiscFeature  MiscVal  MoSold  YrSold
SaleType  \
1454         0    NaN    NaN         NaN        0       6    2006
WD
1455         0    NaN    NaN         NaN        0       4    2006
WD
1456         0    NaN    NaN         NaN        0       9    2006
WD
1457         0    NaN  MnPrv        Shed      700       7    2006
WD
1458         0    NaN    NaN         NaN        0      11    2006
WD

     SaleCondition  SalePrice
1454        Normal        NaN
1455       Abnorml        NaN
1456       Abnorml        NaN
```

```
1457        Normal        NaN
1458        Normal        NaN
```

```
df_train.shape
```

```
(2919, 81)
```

```
train.shape
```

```
(1460, 81)
```

```
test.shape
```

```
(1459, 80)
```

```
df_train.tail()
```

```
        Id  MSSubClass MSZoning  LotFrontage  LotArea Street Alley
LotShape  \
1454  2915         160       RM         21.0     1936   Pave   NaN
Reg
1455  2916         160       RM         21.0     1894   Pave   NaN
Reg
1456  2917          20       RL        160.0    20000   Pave   NaN
Reg
1457  2918          85       RL         62.0    10441   Pave   NaN
Reg
1458  2919          60       RL         74.0     9627   Pave   NaN
Reg

     LandContour Utilities LotConfig LandSlope Neighborhood Condition1
\
1454         Lvl    AllPub    Inside       Gtl      MeadowV       Norm

1455         Lvl    AllPub    Inside       Gtl      MeadowV       Norm

1456         Lvl    AllPub    Inside       Gtl      Mitchel       Norm

1457         Lvl    AllPub    Inside       Gtl      Mitchel       Norm

1458         Lvl    AllPub    Inside       Mod      Mitchel       Norm


     Condition2 BldgType HouseStyle  OverallQual  OverallCond
YearBuilt  \
1454       Norm    Twnhs     2Story            4            7
1970
1455       Norm   TwnhsE     2Story            4            5
1970
1456       Norm     1Fam     1Story            5            7
1960
```

```
1457       Norm     1Fam     SFoyer           5           5
1992
1458       Norm     1Fam     2Story           7           5
1993

      YearRemodAdd RoofStyle RoofMatl Exterior1st Exterior2nd
MasVnrType  \
1454          1970     Gable  CompShg     CemntBd     CmentBd
None
1455          1970     Gable  CompShg     CemntBd     CmentBd
None
1456          1996     Gable  CompShg     VinylSd     VinylSd
None
1457          1992     Gable  CompShg     HdBoard     Wd Shng
None
1458          1994     Gable  CompShg     HdBoard     HdBoard
BrkFace

      MasVnrArea ExterQual ExterCond Foundation BsmtQual BsmtCond  \
1454         0.0        TA        TA     CBlock       TA       TA
1455         0.0        TA        TA     CBlock       TA       TA
1456         0.0        TA        TA     CBlock       TA       TA
1457         0.0        TA        TA      PConc       Gd       TA
1458        94.0        TA        TA      PConc       Gd       TA

     BsmtExposure BsmtFinType1  BsmtFinSF1 BsmtFinType2  BsmtFinSF2  \
1454           No          Unf         0.0          Unf         0.0
1455           No          Rec       252.0          Unf         0.0
1456           No          ALQ      1224.0          Unf         0.0
1457           Av          GLQ       337.0          Unf         0.0
1458           Av          LwQ       758.0          Unf         0.0

      BsmtUnfSF  TotalBsmtSF Heating HeatingQC CentralAir
Electrical  \
1454      546.0        546.0    GasA        Gd          Y      SBrkr

1455      294.0        546.0    GasA        TA          Y      SBrkr

1456        0.0       1224.0    GasA        Ex          Y      SBrkr

1457      575.0        912.0    GasA        TA          Y      SBrkr

1458      238.0        996.0    GasA        Ex          Y      SBrkr


      1stFlrSF  2ndFlrSF  LowQualFinSF  GrLivArea  BsmtFullBath
BsmtHalfBath  \
1454       546       546             0       1092           0.0
0.0
```

```
1455       546       546             0       1092           0.0
0.0
1456      1224         0             0       1224           1.0
0.0
1457       970         0             0        970           0.0
1.0
1458       996      1004             0       2000           0.0
0.0

      FullBath  HalfBath  BedroomAbvGr  KitchenAbvGr KitchenQual  \
1454         1         1             3             1          TA
1455         1         1             3             1          TA
1456         1         0             4             1          TA
1457         1         0             3             1          TA
1458         2         1             3             1          TA

      TotRmsAbvGrd Functional  Fireplaces FireplaceQu GarageType
GarageYrBlt  \
1454             5        Typ           0         NaN        NaN
NaN
1455             6        Typ           0         NaN    CarPort
1970.0
1456             7        Typ           1          TA     Detchd
1960.0
1457             6        Typ           0         NaN        NaN
NaN
1458             9        Typ           1          TA     Attchd
1993.0

     GarageFinish  GarageCars  GarageArea GarageQual GarageCond
PavedDrive  \
1454          NaN         0.0         0.0        NaN        NaN
Y
1455          Unf         1.0       286.0         TA         TA
Y
1456          Unf         2.0       576.0         TA         TA
Y
1457          NaN         0.0         0.0        NaN        NaN
Y
1458          Fin         3.0       650.0         TA         TA
Y

      WoodDeckSF  OpenPorchSF  EnclosedPorch  3SsnPorch
ScreenPorch  \
1454           0            0              0          0            0

1455           0           24              0          0            0

1456         474            0              0          0            0
```

```
1457          80           32               0          0            0

1458         190           48               0          0            0


      PoolArea PoolQC  Fence MiscFeature  MiscVal  MoSold  YrSold 
SaleType  \
1454         0    NaN    NaN         NaN        0       6    2006       
WD
1455         0    NaN    NaN         NaN        0       4    2006       
WD
1456         0    NaN    NaN         NaN        0       9    2006       
WD
1457         0    NaN  MnPrv        Shed      700       7    2006       
WD
1458         0    NaN    NaN         NaN        0      11    2006       
WD

     SaleCondition  SalePrice
1454        Normal        NaN
1455       Abnorml        NaN
1456       Abnorml        NaN
1457        Normal        NaN
1458        Normal        NaN
```

EDA and Feature Engineering

```
duplicate = df_train[df_train.duplicated()]
print(duplicate)

Empty DataFrame
Columns: [Id, MSSubClass, MSZoning, LotFrontage, LotArea, Street, 
Alley, LotShape, LandContour, Utilities, LotConfig, LandSlope, 
Neighborhood, Condition1, Condition2, BldgType, HouseStyle, 
OverallQual, OverallCond, YearBuilt, YearRemodAdd, RoofStyle, 
RoofMatl, Exterior1st, Exterior2nd, MasVnrType, MasVnrArea, ExterQual,
ExterCond, Foundation, BsmtQual, BsmtCond, BsmtExposure, BsmtFinType1,
BsmtFinSF1, BsmtFinType2, BsmtFinSF2, BsmtUnfSF, TotalBsmtSF, Heating,
HeatingQC, CentralAir, Electrical, 1stFlrSF, 2ndFlrSF, LowQualFinSF, 
GrLivArea, BsmtFullBath, BsmtHalfBath, FullBath, HalfBath, 
BedroomAbvGr, KitchenAbvGr, KitchenQual, TotRmsAbvGrd, Functional, 
Fireplaces, FireplaceQu, GarageType, GarageYrBlt, GarageFinish, 
GarageCars, GarageArea, GarageQual, GarageCond, PavedDrive, 
WoodDeckSF, OpenPorchSF, EnclosedPorch, 3SsnPorch, ScreenPorch, 
PoolArea, PoolQC, Fence, MiscFeature, MiscVal, MoSold, YrSold, 
SaleType, SaleCondition, SalePrice]
Index: []

df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2919 entries, 0 to 1458
Data columns (total 81 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Id             2919 non-null   int64  
 1   MSSubClass     2919 non-null   int64  
 2   MSZoning       2915 non-null   object 
 3   LotFrontage    2433 non-null   float64
 4   LotArea        2919 non-null   int64  
 5   Street         2919 non-null   object 
 6   Alley          198 non-null    object 
 7   LotShape       2919 non-null   object 
 8   LandContour    2919 non-null   object 
 9   Utilities      2917 non-null   object 
 10  LotConfig      2919 non-null   object 
 11  LandSlope      2919 non-null   object 
 12  Neighborhood   2919 non-null   object 
 13  Condition1     2919 non-null   object 
 14  Condition2     2919 non-null   object 
 15  BldgType       2919 non-null   object 
 16  HouseStyle     2919 non-null   object 
 17  OverallQual    2919 non-null   int64  
 18  OverallCond    2919 non-null   int64  
 19  YearBuilt      2919 non-null   int64  
 20  YearRemodAdd   2919 non-null   int64  
 21  RoofStyle      2919 non-null   object 
 22  RoofMatl       2919 non-null   object 
 23  Exterior1st    2918 non-null   object 
 24  Exterior2nd    2918 non-null   object 
 25  MasVnrType     2895 non-null   object 
 26  MasVnrArea     2896 non-null   float64
 27  ExterQual      2919 non-null   object 
 28  ExterCond      2919 non-null   object 
 29  Foundation     2919 non-null   object 
 30  BsmtQual       2838 non-null   object 
 31  BsmtCond       2837 non-null   object 
 32  BsmtExposure   2837 non-null   object 
 33  BsmtFinType1   2840 non-null   object 
 34  BsmtFinSF1     2918 non-null   float64
 35  BsmtFinType2   2839 non-null   object 
 36  BsmtFinSF2     2918 non-null   float64
 37  BsmtUnfSF      2918 non-null   float64
 38  TotalBsmtSF    2918 non-null   float64
 39  Heating        2919 non-null   object 
 40  HeatingQC      2919 non-null   object 
 41  CentralAir     2919 non-null   object 
 42  Electrical     2918 non-null   object 
 43  1stFlrSF       2919 non-null   int64  
 44  2ndFlrSF       2919 non-null   int64  
```

```
 45  LowQualFinSF   2919 non-null   int64
 46  GrLivArea      2919 non-null   int64
 47  BsmtFullBath   2917 non-null   float64
 48  BsmtHalfBath   2917 non-null   float64
 49  FullBath       2919 non-null   int64
 50  HalfBath       2919 non-null   int64
 51  BedroomAbvGr   2919 non-null   int64
 52  KitchenAbvGr   2919 non-null   int64
 53  KitchenQual    2918 non-null   object
 54  TotRmsAbvGrd   2919 non-null   int64
 55  Functional     2917 non-null   object
 56  Fireplaces     2919 non-null   int64
 57  FireplaceQu    1499 non-null   object
 58  GarageType     2762 non-null   object
 59  GarageYrBlt    2760 non-null   float64
 60  GarageFinish   2760 non-null   object
 61  GarageCars     2918 non-null   float64
 62  GarageArea     2918 non-null   float64
 63  GarageQual     2760 non-null   object
 64  GarageCond     2760 non-null   object
 65  PavedDrive     2919 non-null   object
 66  WoodDeckSF     2919 non-null   int64
 67  OpenPorchSF    2919 non-null   int64
 68  EnclosedPorch  2919 non-null   int64
 69  3SsnPorch      2919 non-null   int64
 70  ScreenPorch    2919 non-null   int64
 71  PoolArea       2919 non-null   int64
 72  PoolQC         10 non-null     object
 73  Fence          571 non-null    object
 74  MiscFeature    105 non-null    object
 75  MiscVal        2919 non-null   int64
 76  MoSold         2919 non-null   int64
 77  YrSold         2919 non-null   int64
 78  SaleType       2918 non-null   object
 79  SaleCondition  2919 non-null   object
 80  SalePrice      1460 non-null   float64
dtypes: float64(12), int64(26), object(43)
memory usage: 1.8+ MB
```

```
df_train.describe()
```

```
                Id   MSSubClass  LotFrontage       LotArea
OverallQual  \
count  2919.000000  2919.000000  2433.000000   2919.000000
2919.000000
mean   1460.000000    57.137718    69.305795  10168.114080
6.089072
std     842.787043    42.517628    23.344905   7886.996359
1.409947
min       1.000000    20.000000    21.000000   1300.000000
1.000000
```

```
25%     730.500000    20.000000    59.000000    7478.000000
5.000000
50%    1460.000000    50.000000    68.000000    9453.000000
6.000000
75%    2189.500000    70.000000    80.000000   11570.000000
7.000000
max    2919.000000   190.000000   313.000000  215245.000000
10.000000

       OverallCond    YearBuilt  YearRemodAdd   MasVnrArea
BsmtFinSF1  \
count  2919.000000  2919.000000   2919.000000  2896.000000
2918.000000
mean      5.564577  1971.312778   1984.264474   102.201312
441.423235
std       1.113131    30.291442     20.894344   179.334253
455.610826
min       1.000000  1872.000000   1950.000000     0.000000
0.000000
25%       5.000000  1953.500000   1965.000000     0.000000
0.000000
50%       5.000000  1973.000000   1993.000000     0.000000
368.500000
75%       6.000000  2001.000000   2004.000000   164.000000
733.000000
max       9.000000  2010.000000   2010.000000  1600.000000
5644.000000

        BsmtFinSF2    BsmtUnfSF  TotalBsmtSF     1stFlrSF     2ndFlrSF
\
count  2918.000000  2918.000000  2918.000000  2919.000000  2919.000000

mean     49.582248   560.772104  1051.777587  1159.581706   336.483727

std     169.205611   439.543659   440.766258   392.362079   428.701456

min       0.000000     0.000000     0.000000   334.000000     0.000000

25%       0.000000   220.000000   793.000000   876.000000     0.000000

50%       0.000000   467.000000   989.500000  1082.000000     0.000000

75%       0.000000   805.500000  1302.000000  1387.500000   704.000000

max    1526.000000  2336.000000  6110.000000  5095.000000  2065.000000


       LowQualFinSF    GrLivArea  BsmtFullBath  BsmtHalfBath
FullBath  \
```

```
count   2919.000000  2919.000000   2917.000000   2917.000000
2919.000000
mean       4.694416  1500.759849      0.429894      0.061364
1.568003
std       46.396825   506.051045      0.524736      0.245687
0.552969
min        0.000000   334.000000      0.000000      0.000000
0.000000
25%        0.000000  1126.000000      0.000000      0.000000
1.000000
50%        0.000000  1444.000000      0.000000      0.000000
2.000000
75%        0.000000  1743.500000      1.000000      0.000000
2.000000
max     1064.000000  5642.000000      3.000000      2.000000
4.000000

          HalfBath  BedroomAbvGr  KitchenAbvGr  TotRmsAbvGrd
Fireplaces  \
count  2919.000000   2919.000000   2919.000000   2919.000000
2919.000000
mean      0.380267      2.860226      1.044536      6.451524
0.597122
std       0.502872      0.822693      0.214462      1.569379
0.646129
min       0.000000      0.000000      0.000000      2.000000
0.000000
25%       0.000000      2.000000      1.000000      5.000000
0.000000
50%       0.000000      3.000000      1.000000      6.000000
1.000000
75%       1.000000      3.000000      1.000000      7.000000
1.000000
max       2.000000      8.000000      3.000000     15.000000
4.000000

       GarageYrBlt   GarageCars   GarageArea   WoodDeckSF  OpenPorchSF
\
count  2760.000000  2918.000000  2918.000000  2919.000000  2919.000000

mean   1978.113406     1.766621   472.874572    93.709832    47.486811

std      25.574285     0.761624   215.394815   126.526589    67.575493

min    1895.000000     0.000000     0.000000     0.000000     0.000000

25%    1960.000000     1.000000   320.000000     0.000000     0.000000

50%    1979.000000     2.000000   480.000000     0.000000    26.000000
```

```
75%    2002.000000     2.000000   576.000000   168.000000    70.000000

max    2207.000000     5.000000  1488.000000  1424.000000   742.000000


       EnclosedPorch    3SsnPorch  ScreenPorch     PoolArea
MiscVal  \
count    2919.000000  2919.000000  2919.000000  2919.000000
2919.000000
mean       23.098321     2.602261    16.062350     2.251799
50.825968
std        64.244246    25.188169    56.184365    35.663946
567.402211
min         0.000000     0.000000     0.000000     0.000000
0.000000
25%         0.000000     0.000000     0.000000     0.000000
0.000000
50%         0.000000     0.000000     0.000000     0.000000
0.000000
75%         0.000000     0.000000     0.000000     0.000000
0.000000
max      1012.000000   508.000000   576.000000   800.000000
17000.000000

            MoSold       YrSold      SalePrice
count  2919.000000  2919.000000    1460.000000
mean      6.213087  2007.792737  180921.195890
std       2.714762     1.314964   79442.502883
min       1.000000  2006.000000   34900.000000
25%       4.000000  2007.000000  129975.000000
50%       6.000000  2008.000000  163000.000000
75%       8.000000  2009.000000  214000.000000
max      12.000000  2010.000000  755000.000000
```

Handling numerical Missing values

1. For Continious

```
#missing_values_continious = [feature for feature in df_train.columns
if df_train[feature].dtype != "O" and len(df_train[feature].unique())
>20 and df_train[feature].isnull().sum()>0]
#missing_values_continious

missing_values_continious = []
for feature in df_train.columns:
    if df_train[feature].dtype != "O" and
len(df_train[feature].unique())>20:
        missing_values_continious.append(feature)
missing_values_continious
```

```
['Id',
 'LotFrontage',
 'LotArea',
 'YearBuilt',
 'YearRemodAdd',
 'MasVnrArea',
 'BsmtFinSF1',
 'BsmtFinSF2',
 'BsmtUnfSF',
 'TotalBsmtSF',
 '1stFlrSF',
 '2ndFlrSF',
 'LowQualFinSF',
 'GrLivArea',
 'GarageYrBlt',
 'GarageArea',
 'WoodDeckSF',
 'OpenPorchSF',
 'EnclosedPorch',
 '3SsnPorch',
 'ScreenPorch',
 'MiscVal',
 'SalePrice']
```

```
for feature in missing_values_continious:
    print(feature, round(df_train[feature].isnull().mean(),4)*100)
```

```
Id 0.0
LotFrontage 16.650000000000002
LotArea 0.0
YearBuilt 0.0
YearRemodAdd 0.0
MasVnrArea 0.79
BsmtFinSF1 0.03
BsmtFinSF2 0.03
BsmtUnfSF 0.03
TotalBsmtSF 0.03
1stFlrSF 0.0
2ndFlrSF 0.0
LowQualFinSF 0.0
GrLivArea 0.0
GarageYrBlt 5.45
GarageArea 0.03
WoodDeckSF 0.0
OpenPorchSF 0.0
EnclosedPorch 0.0
3SsnPorch 0.0
ScreenPorch 0.0
MiscVal 0.0
SalePrice 49.980000000000004
```

```
median_value = df_train["LotFrontage"].median()
```

```
median_value
```

```
68.0
```

```
for feature in missing_values_continious:
    if feature == "SalePrice":
        pass
    else:
        median_value = df_train[feature].median()
        df_train[feature].fillna(median_value,inplace=True)
#
```

```
df_train.drop("Id" , inplace=True , axis = 1)
```

2. For Descrete

```
#missing_values_descrete = [feature for feature in df_train.columns if
df_train[feature].dtype != "O" and len(df_train[feature].unique()) <20
and df_train[feature].isnull().sum()>0]
#missing_values_descrete
```

```
if df_train["Alley"].dtype != "object":
    print(True)
else:
    print(False)
```

```
False
```

```
missing_values_descrete = []
for feature in df_train.columns:
    if df_train[feature].dtype != "O" and
len(df_train[feature].unique()) <=20:
        missing_values_descrete.append(feature)
missing_values_descrete
```

```
['MSSubClass',
 'OverallQual',
 'OverallCond',
 'BsmtFullBath',
 'BsmtHalfBath',
 'FullBath',
 'HalfBath',
 'BedroomAbvGr',
 'KitchenAbvGr',
 'TotRmsAbvGrd',
 'Fireplaces',
 'GarageCars',
 'PoolArea',
 'MoSold',
 'YrSold']
```

```
for feature in missing_values_descrete:
    print(feature, round(df_train[feature].isnull().mean(),4)*100)
```

```
MSSubClass 0.0
OverallQual 0.0
OverallCond 0.0
BsmtFullBath 0.06999999999999999
BsmtHalfBath 0.06999999999999999
FullBath 0.0
HalfBath 0.0
BedroomAbvGr 0.0
KitchenAbvGr 0.0
TotRmsAbvGrd 0.0
Fireplaces 0.0
GarageCars 0.03
PoolArea 0.0
MoSold 0.0
YrSold 0.0
```

```
df_train["BsmtFullBath"].mode()[0]
```

```
0.0
```

```
df_train["BsmtFullBath"].unique()
```

```
array([ 1.,  0.,  2.,  3., nan])
```

```
for feature in missing_values_descrete:
    mode_value = df_train[feature].mode()[0]
    df_train[feature].fillna(mode_value,inplace=True)
```

Handling categorical missing values

```
#missing_values_c = [feature for feature in df_train.columns if
df_train[feature].dtype == "O" and df_train[feature].isnull().sum()>0]
#missing_values_c
```

```
missing_values_c = []
for feature in df_train.columns:
    if df_train[feature].dtype == "O" and
df_train[feature].isnull().sum()>0:
        missing_values_c.append(feature)
missing_values_c
```

```
['MSZoning',
 'Alley',
 'Utilities',
 'Exterior1st',
 'Exterior2nd',
 'MasVnrType',
 'BsmtQual',
 'BsmtCond',
 'BsmtExposure',
```

```
 'BsmtFinType1',
 'BsmtFinType2',
 'Electrical',
 'KitchenQual',
 'Functional',
 'FireplaceQu',
 'GarageType',
 'GarageFinish',
 'GarageQual',
 'GarageCond',
 'PoolQC',
 'Fence',
 'MiscFeature',
 'SaleType']
```

```
for feature in missing_values_c:
    print(feature, round(df_train[feature].isnull().mean(),4)*100)
```

```
MSZoning 0.13999999999999999
Alley 93.22
Utilities 0.06999999999999999
Exterior1st 0.03
Exterior2nd 0.03
MasVnrType 0.8200000000000001
BsmtQual 2.77
BsmtCond 2.81
BsmtExposure 2.81
BsmtFinType1 2.71
BsmtFinType2 2.74
Electrical 0.03
KitchenQual 0.03
Functional 0.06999999999999999
FireplaceQu 48.65
GarageType 5.38
GarageFinish 5.45
GarageQual 5.45
GarageCond 5.45
PoolQC 99.66000000000001
Fence 80.44
MiscFeature 96.39999999999999
SaleType 0.03
```

```
for feature in missing_values_c:
    mode_value = df_train[feature].mode()[0]
    df_train[feature].fillna(mode_value,inplace=True)
df_train.drop(["Alley" ,"PoolQC", "Fence" , "MiscFeature"  ,
"FireplaceQu" ] , axis = 1 , inplace = True)
```

```
df_train.isnull().sum().sum()
```

```
1459
```

```
df_train.shape
```

```
(2919, 75)
```

Handling year feature

```
#year = [feature for feature in df_train.columns if "Yr" in feature or
"Year" in feature]
#year
```

```
year = []
for feature in df_train.columns:
    if "Yr" in feature or "Year" in feature:
        year.append(feature)
year
```

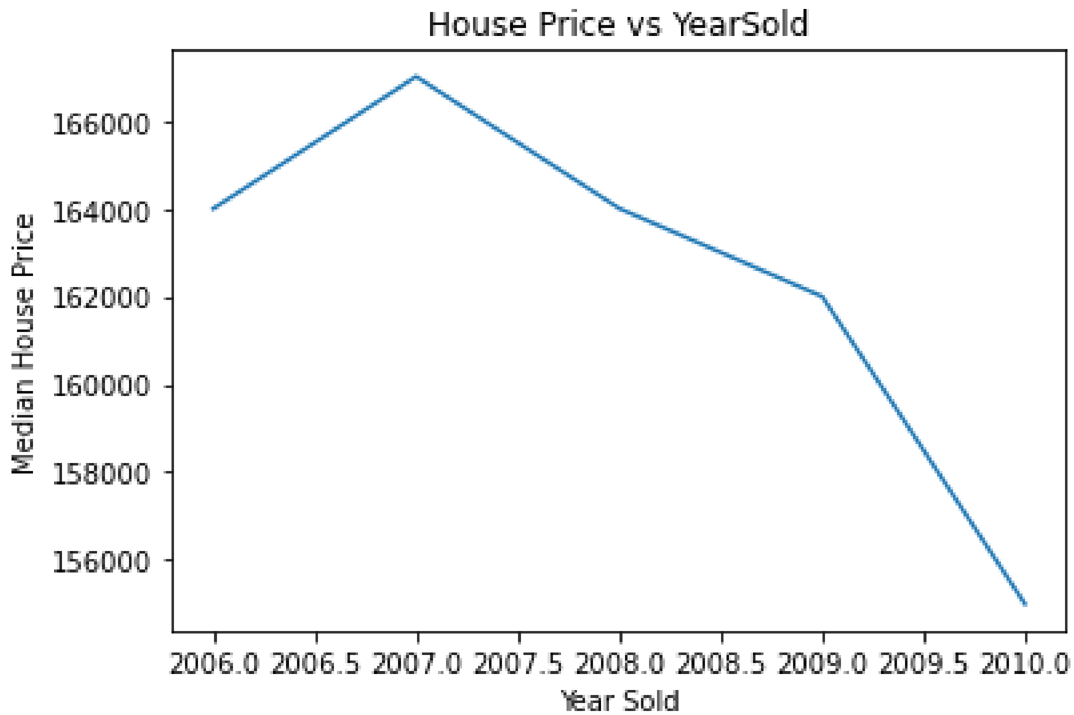
```
['YearBuilt', 'YearRemodAdd', 'GarageYrBlt', 'YrSold']
```

```
for feature in year:
    print(feature, len(df_train[feature].unique()) ,
df_train[feature].dtype)
```

```
YearBuilt 118 int64
YearRemodAdd 61 int64
GarageYrBlt 103 float64
YrSold 5 int64
```

```
df_train.groupby('YrSold')['SalePrice'].median().plot()
plt.xlabel('Year Sold')
plt.ylabel('Median House Price')
plt.title("House Price vs YearSold")
```

```
Text(0.5, 1.0, 'House Price vs YearSold')
```




```
for feature in year:
    df_train[feature]=df_train['YrSold']-df_train[feature]
df_train.drop("YrSold", axis = 1 , inplace = True)

df_train.shape

(2919, 74)

df_train.head()

   MSSubClass MSZoning  LotFrontage  LotArea Street LotShape 
LandContour  \
0          60       RL         65.0     8450   Pave      Reg 
Lvl
1          20       RL         80.0     9600   Pave      Reg 
Lvl
2          60       RL         68.0    11250   Pave      IR1 
Lvl
3          70       RL         60.0     9550   Pave      IR1 
Lvl
4          60       RL         84.0    14260   Pave      IR1 
Lvl

  Utilities LotConfig LandSlope Neighborhood Condition1 Condition2 
BldgType  \
0    AllPub    Inside       Gtl      CollgCr       Norm       Norm 
1Fam
1    AllPub       FR2       Gtl      Veenker      Feedr       Norm 
```

```
1Fam
2    AllPub    Inside       Gtl      CollgCr       Norm       Norm
1Fam
3    AllPub    Corner       Gtl      Crawfor       Norm       Norm
1Fam
4    AllPub       FR2       Gtl      NoRidge       Norm       Norm
1Fam

  HouseStyle  OverallQual  OverallCond  YearBuilt  YearRemodAdd
RoofStyle  \
0     2Story            7            5          5             5
Gable
1     1Story            6            8         31            31
Gable
2     2Story            7            5          7             6
Gable
3     2Story            7            5         91            36
Gable
4     2Story            8            5          8             8
Gable

  RoofMatl Exterior1st Exterior2nd MasVnrType  MasVnrArea ExterQual
ExterCond  \
0  CompShg     VinylSd     VinylSd    BrkFace       196.0        Gd
TA
1  CompShg     MetalSd     MetalSd       None         0.0        TA
TA
2  CompShg     VinylSd     VinylSd    BrkFace       162.0        Gd
TA
3  CompShg     Wd Sdng     Wd Shng       None         0.0        TA
TA
4  CompShg     VinylSd     VinylSd    BrkFace       350.0        Gd
TA

  Foundation BsmtQual BsmtCond BsmtExposure BsmtFinType1
BsmtFinSF1  \
0      PConc       Gd       TA           No          GLQ       706.0

1     CBlock       Gd       TA           Gd          ALQ       978.0

2      PConc       Gd       TA           Mn          GLQ       486.0

3     BrkTil       TA       Gd           No          ALQ       216.0

4      PConc       Gd       TA           Av          GLQ       655.0


  BsmtFinType2  BsmtFinSF2  BsmtUnfSF  TotalBsmtSF Heating
HeatingQC  \
```

```
0         Unf         0.0      150.0        856.0    GasA        Ex

1         Unf         0.0      284.0       1262.0    GasA        Ex

2         Unf         0.0      434.0        920.0    GasA        Ex

3         Unf         0.0      540.0        756.0    GasA        Gd

4         Unf         0.0      490.0       1145.0    GasA        Ex


  CentralAir Electrical  1stFlrSF  2ndFlrSF  LowQualFinSF
GrLivArea  \
0          Y      SBrkr       856       854             0       1710

1          Y      SBrkr      1262         0             0       1262

2          Y      SBrkr       920       866             0       1786

3          Y      SBrkr       961       756             0       1717

4          Y      SBrkr      1145      1053             0       2198


   BsmtFullBath  BsmtHalfBath  FullBath  HalfBath  BedroomAbvGr
KitchenAbvGr  \
0           1.0           0.0         2         1             3
1
1           0.0           1.0         2         0             3
1
2           1.0           0.0         2         1             3
1
3           1.0           0.0         1         0             3
1
4           1.0           0.0         2         1             4
1

  KitchenQual  TotRmsAbvGrd Functional  Fireplaces GarageType
GarageYrBlt  \
0          Gd             8        Typ           0     Attchd
5.0
1          TA             6        Typ           1     Attchd
31.0
2          Gd             6        Typ           1     Attchd
7.0
3          Gd             7        Typ           1     Detchd
8.0
4          Gd             9        Typ           1     Attchd
8.0
```

```
  GarageFinish  GarageCars  GarageArea GarageQual GarageCond 
PavedDrive  \
0          RFn          2.0       548.0         TA         TA 
Y   
1          RFn          2.0       460.0         TA         TA 
Y   
2          RFn          2.0       608.0         TA         TA 
Y   
3          Unf          3.0       642.0         TA         TA 
Y   
4          RFn          3.0       836.0         TA         TA 
Y   

   WoodDeckSF  OpenPorchSF  EnclosedPorch  3SsnPorch  ScreenPorch 
PoolArea  \
0           0           61              0          0            0 
0   
1         298            0              0          0            0 
0   
2           0           42              0          0            0 
0   
3           0           35            272          0            0 
0   
4         192           84              0          0            0 
0   

   MiscVal  MoSold SaleType SaleCondition  SalePrice  
0        0       2       WD        Normal   208500.0  
1        0       5       WD        Normal   181500.0  
2        0       9       WD        Normal   223500.0  
3        0       2       WD       Abnorml   140000.0  
4        0      12       WD        Normal   250000.0  
```

Handling continious values

```
#continious = [feature for feature in df_train.columns if 
len(df_train[feature].unique())>20 and df_train[feature].dtype != "O" 
and feature not in year]
#continious

continious = []
for feature in df_train.columns:
     if df_train[feature].dtype != "O" and 
len(df_train[feature].unique())>20  and feature not in year:
            continious.append(feature)
continious

['LotFrontage',
 'LotArea',
 'MasVnrArea',
```

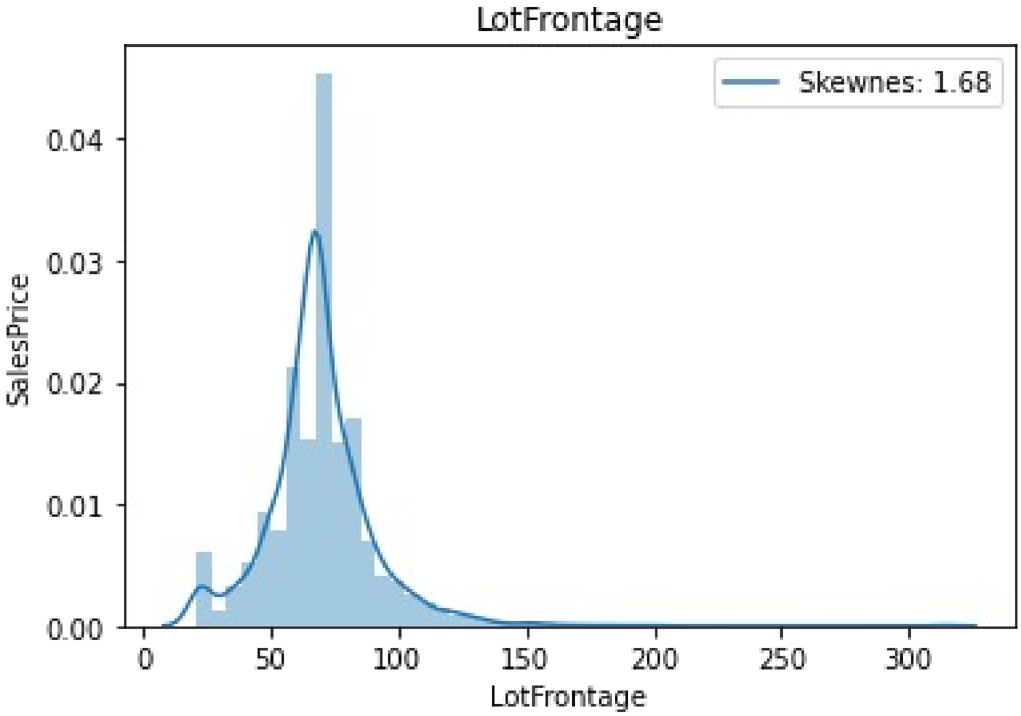
```
 'BsmtFinSF1',
 'BsmtFinSF2',
 'BsmtUnfSF',
 'TotalBsmtSF',
 '1stFlrSF',
 '2ndFlrSF',
 'LowQualFinSF',
 'GrLivArea',
 'GarageArea',
 'WoodDeckSF',
 'OpenPorchSF',
 'EnclosedPorch',
 '3SsnPorch',
 'ScreenPorch',
 'MiscVal',
 'SalePrice']
```

```
data["LotFrontage"].skew()
```

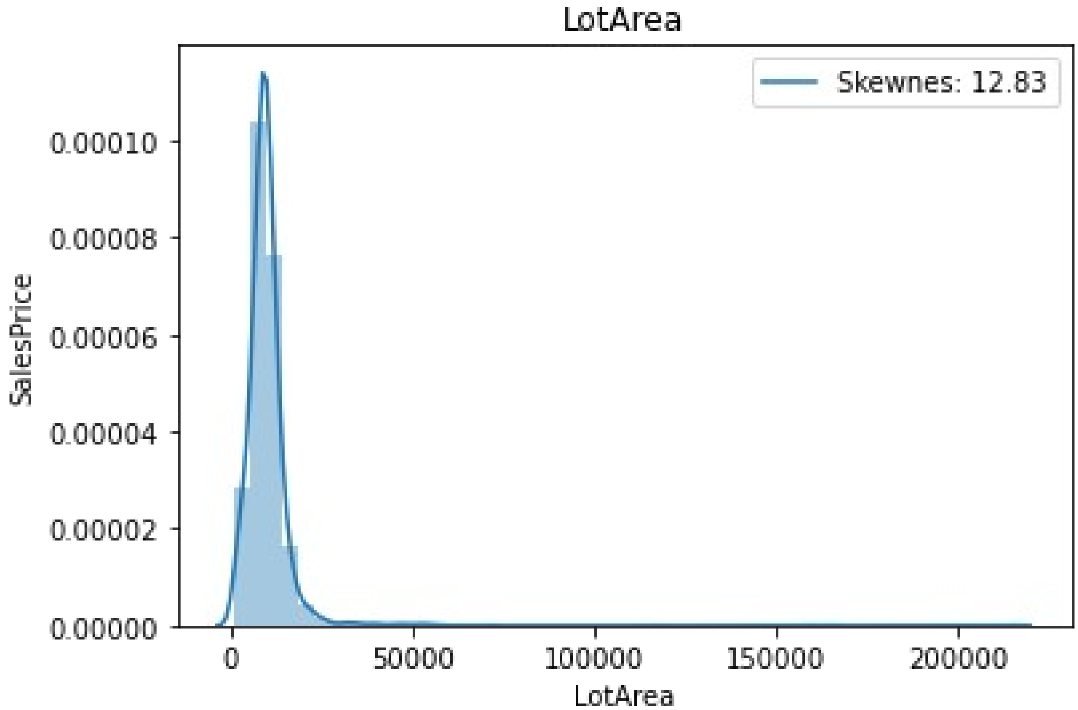
```
-0.9948415692198087
```

```
## We will be using logarithmic transformation
for feature in continious:
    data = df_train.copy()
    #data[feature]=np.log1p(data[feature])
    ax = sns.distplot(data[feature])
    ax.legend(["Skewnes: {:0.2f}".format(data[feature].skew())])
    plt.xlabel(feature)
    plt.ylabel('SalesPrice')
    plt.title(feature)
    plt.show()
```

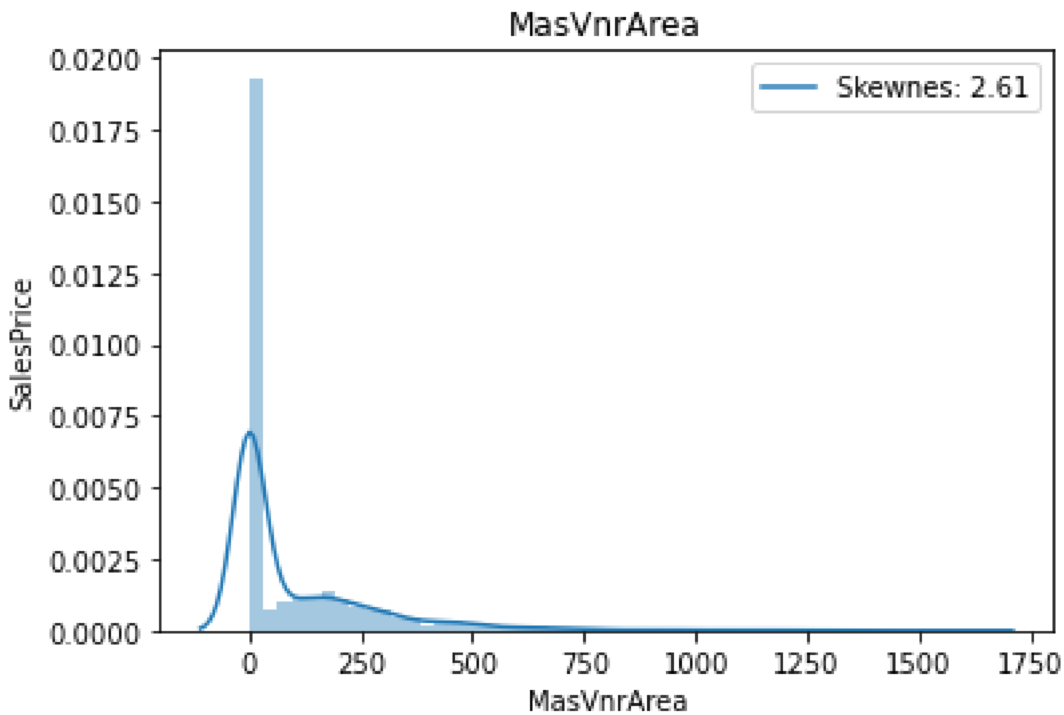
```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



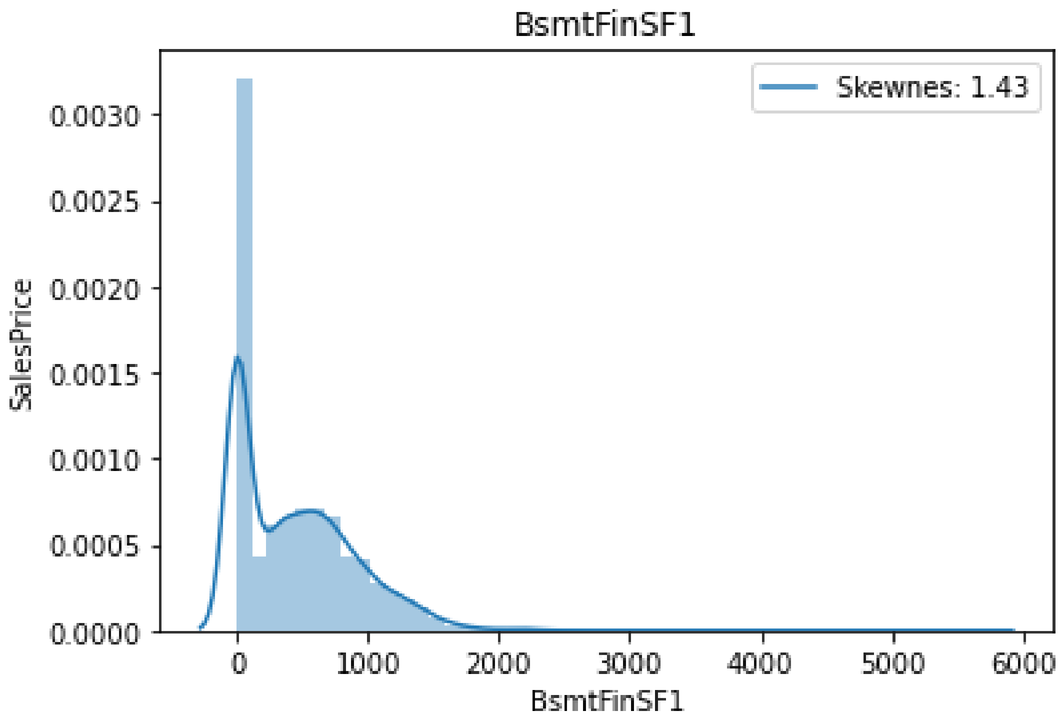

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



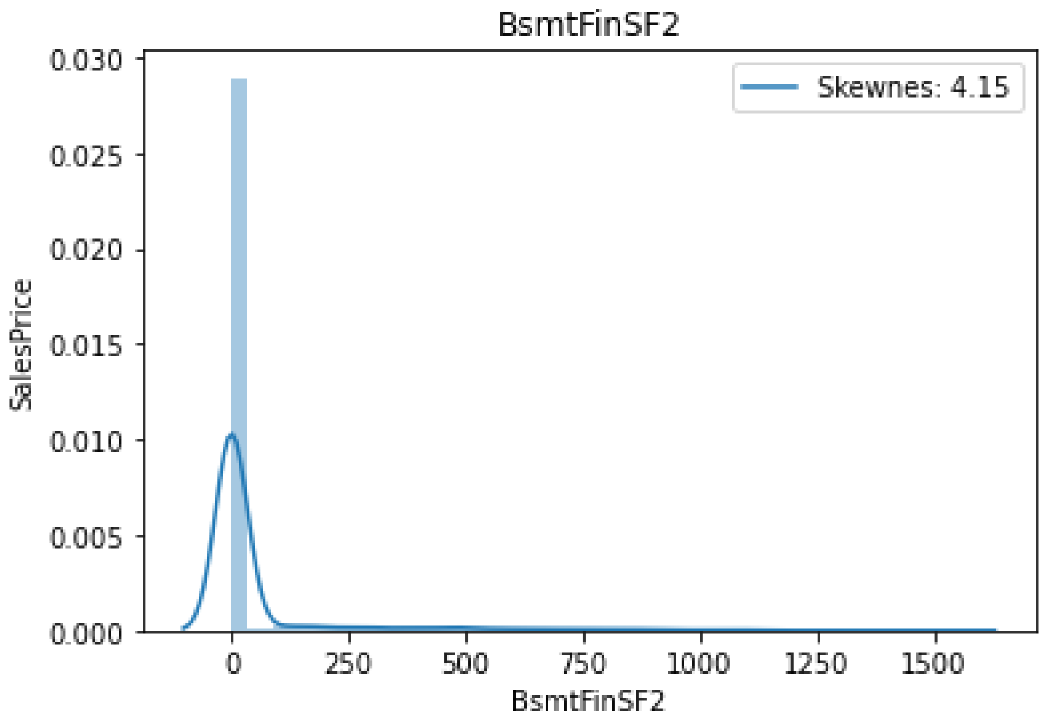

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



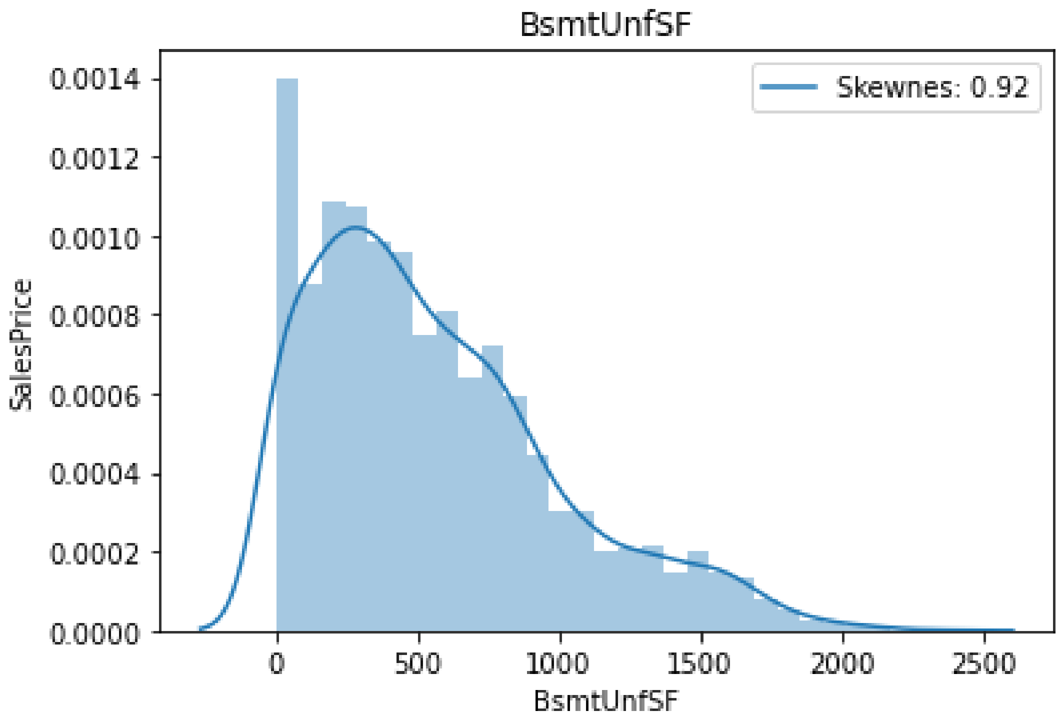

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



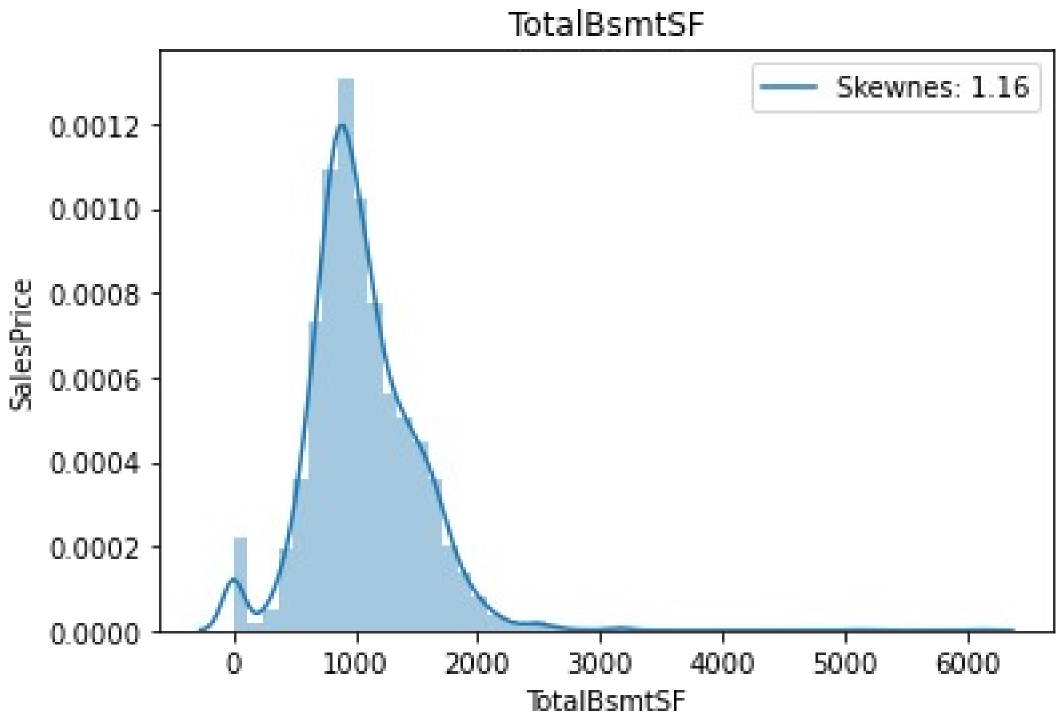

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



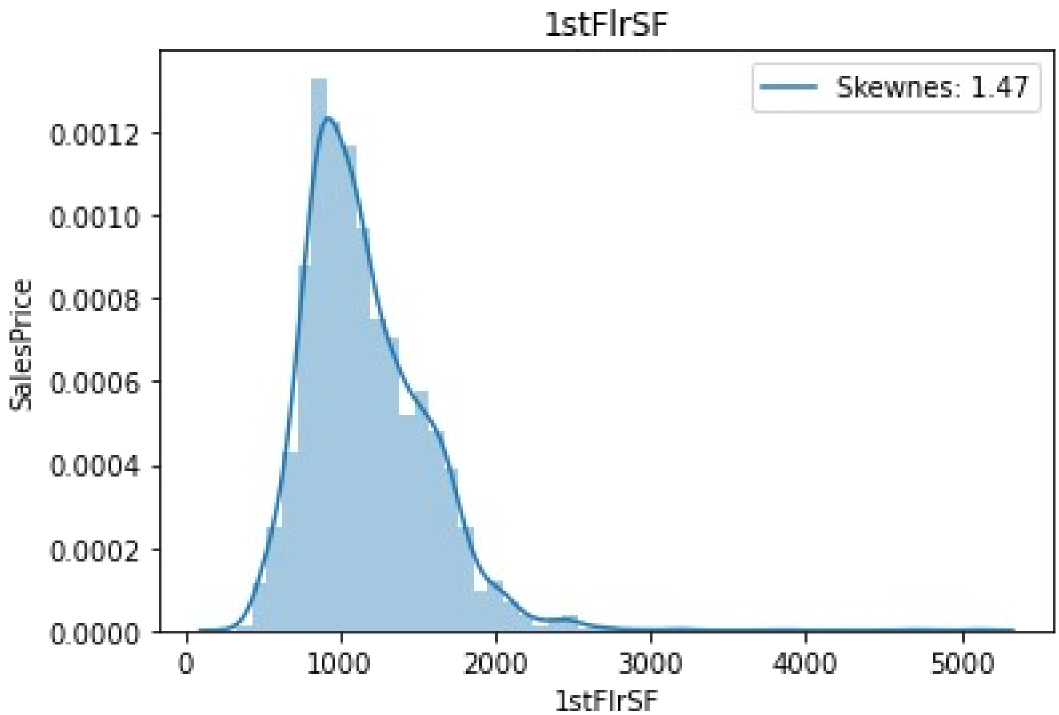

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



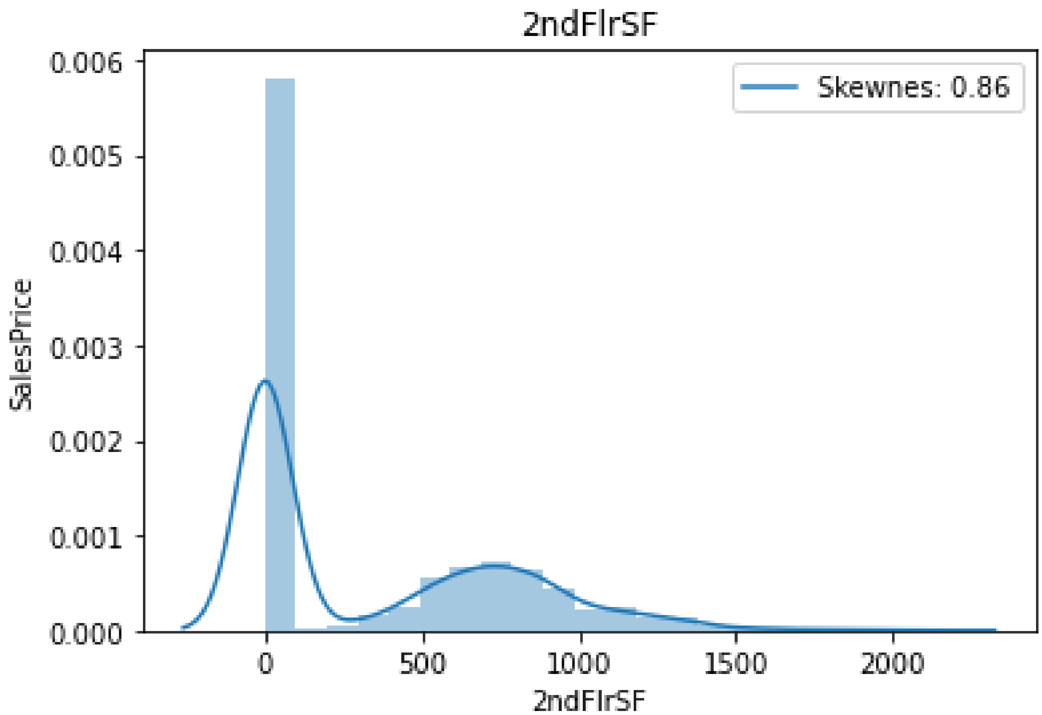

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



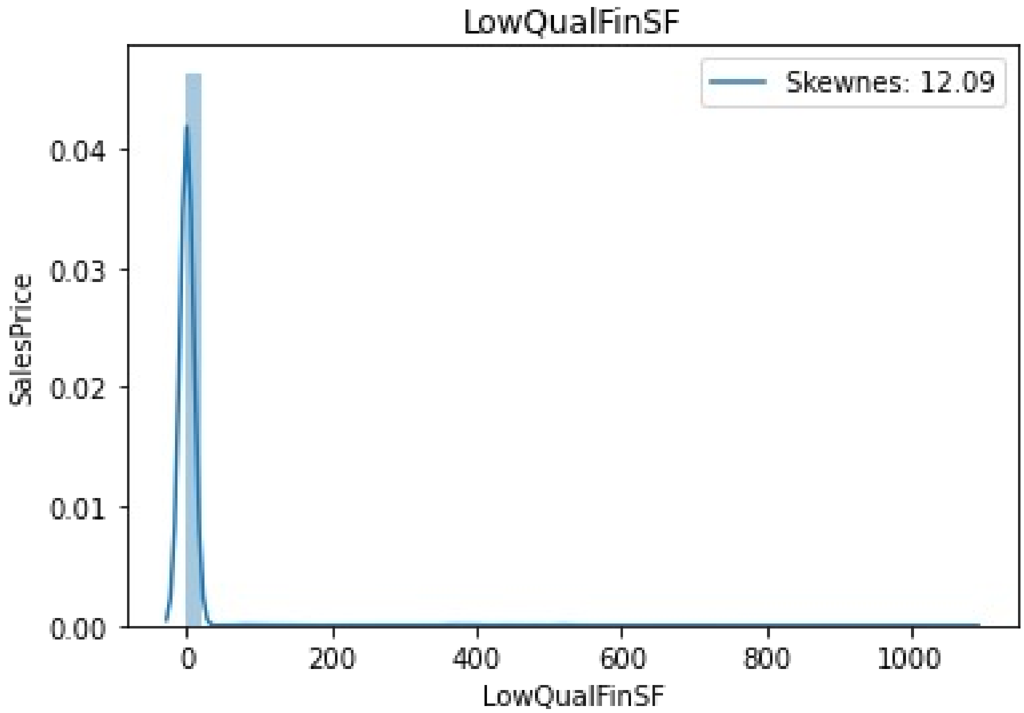

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



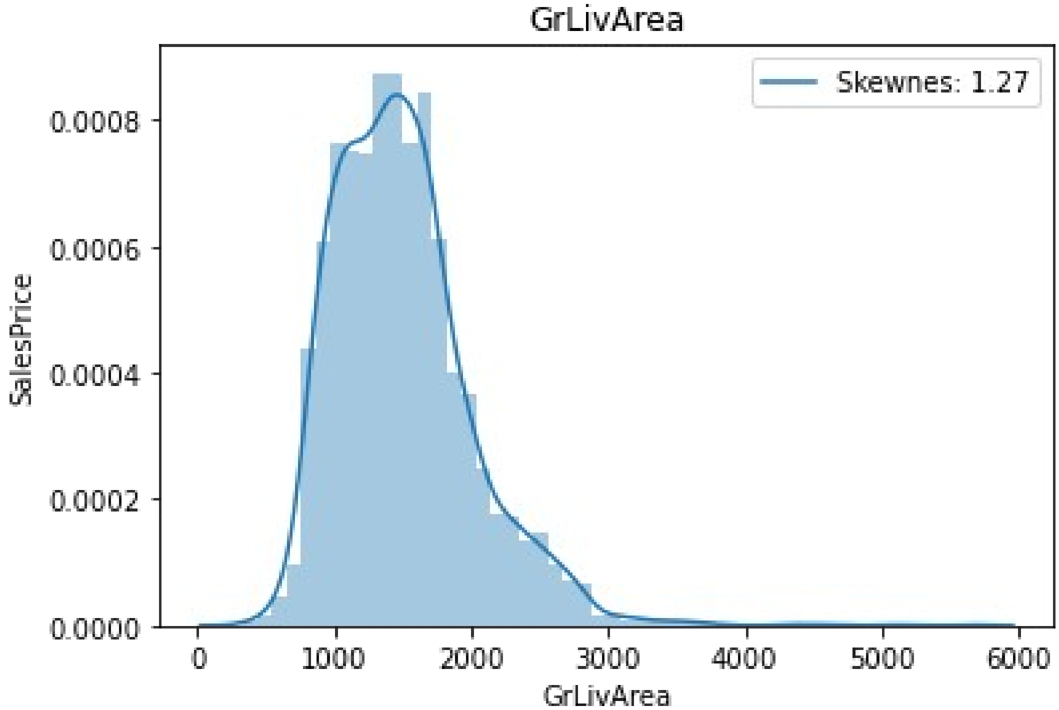

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



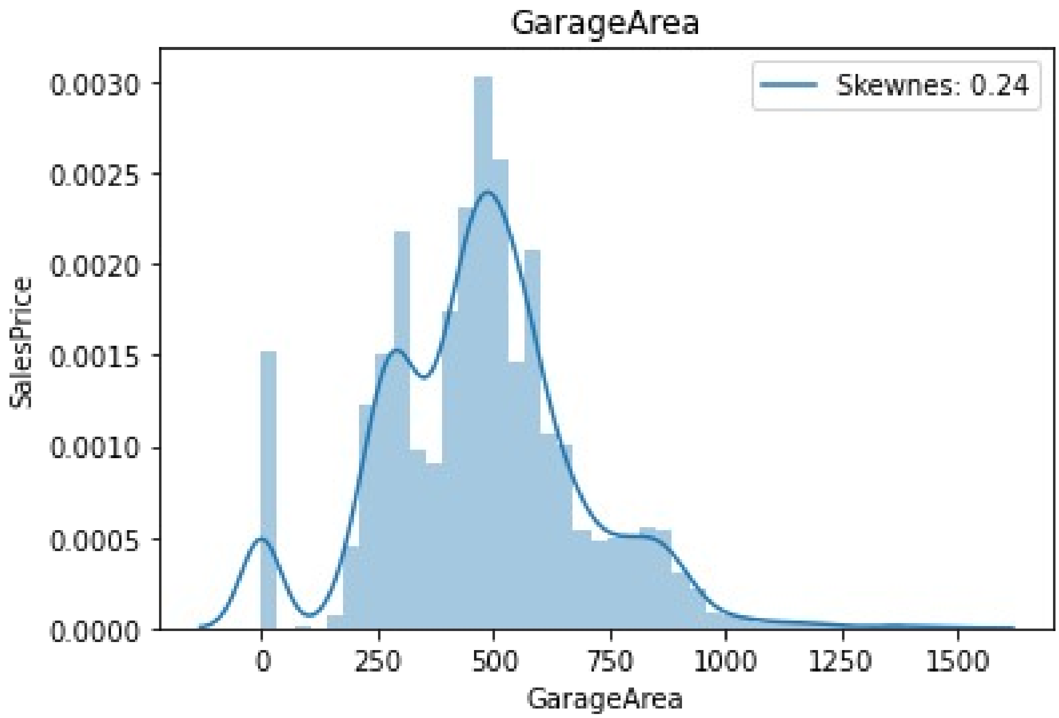

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



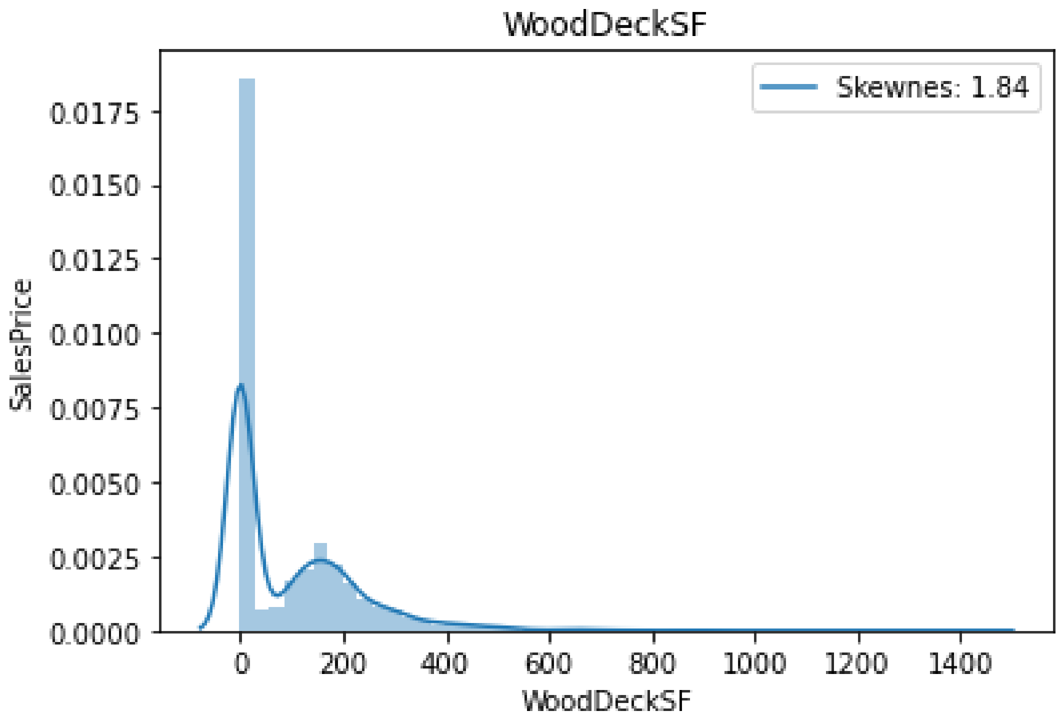

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



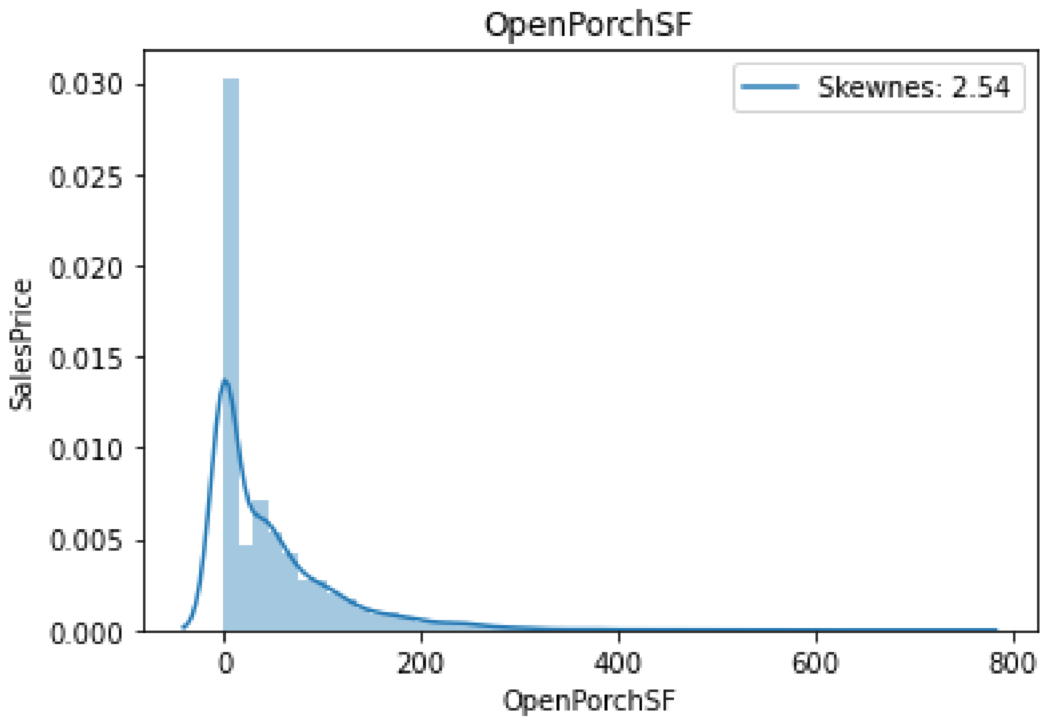

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



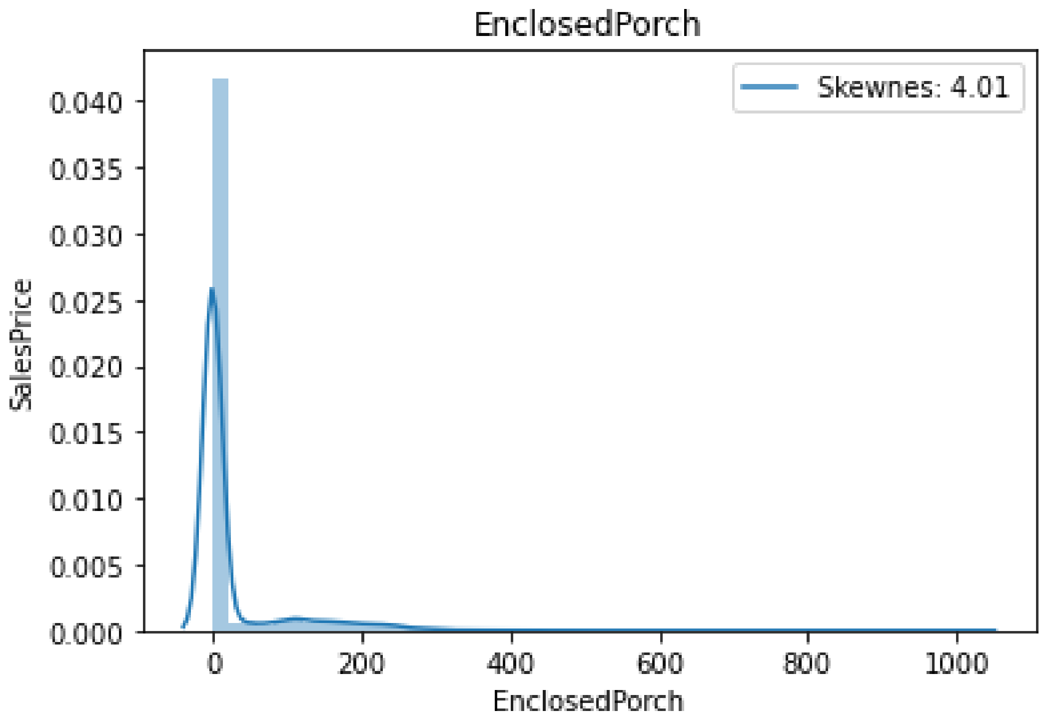

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



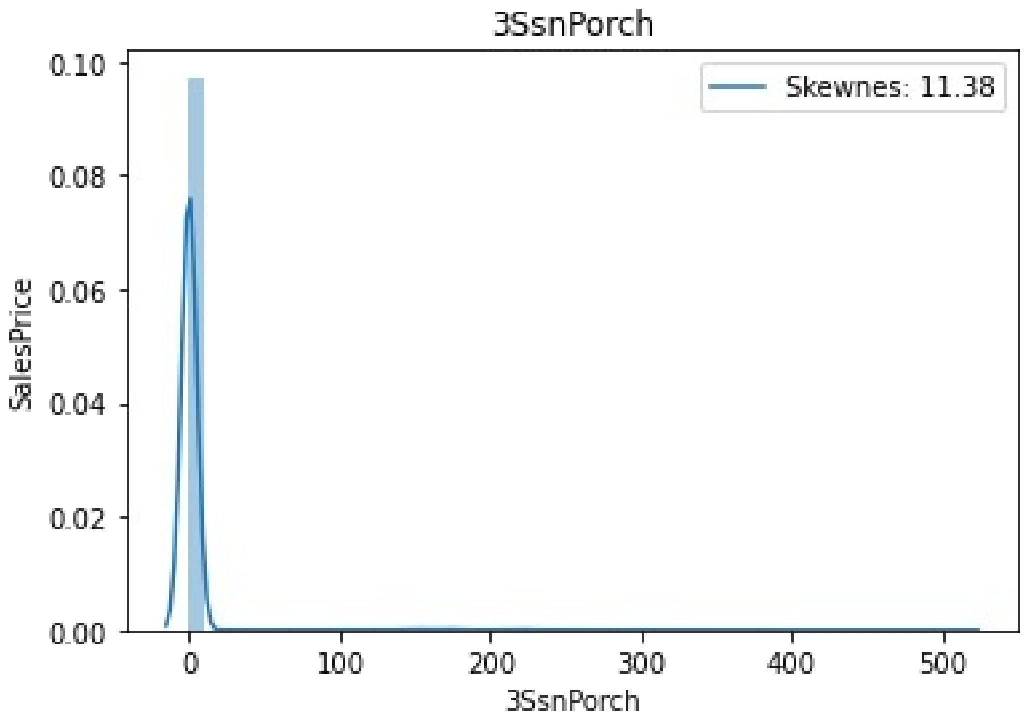

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



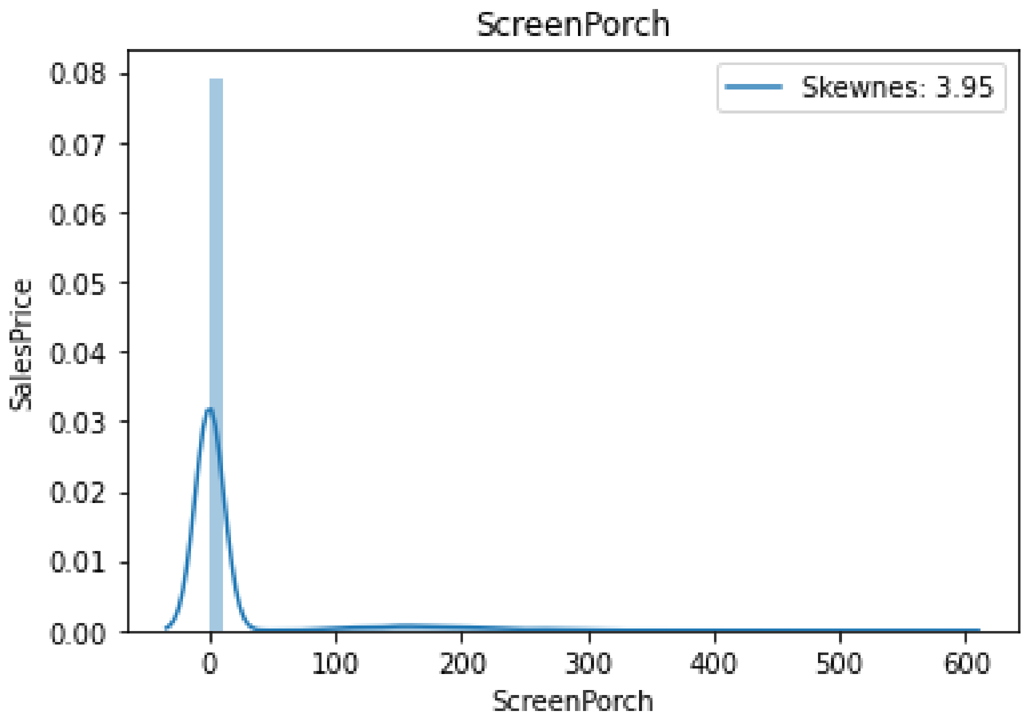

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



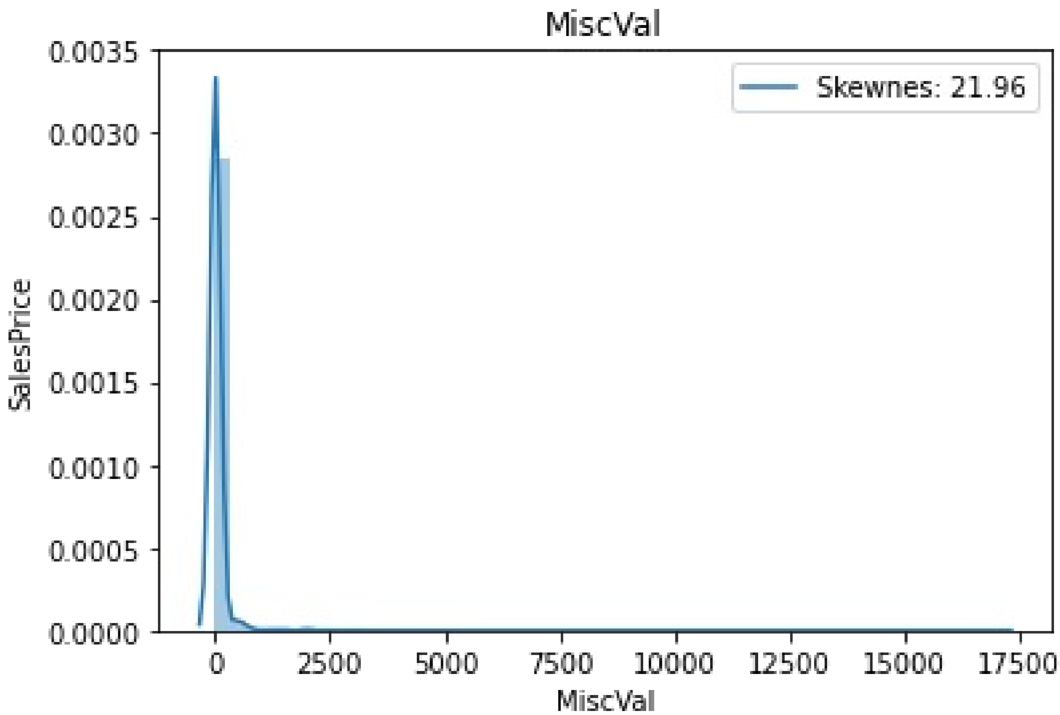

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



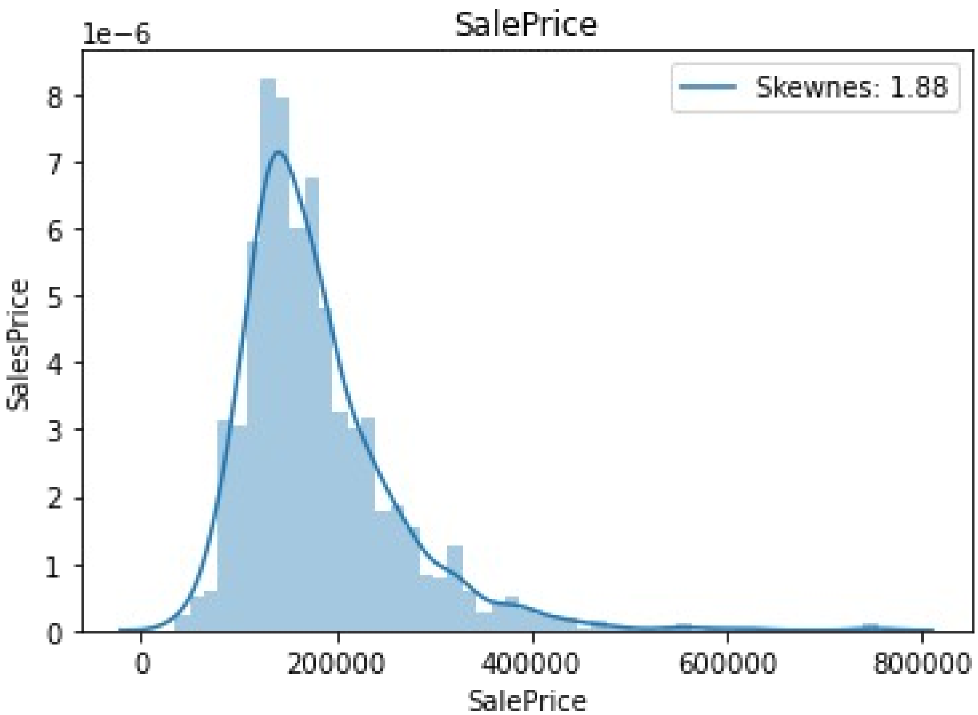

```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```




```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```




```
C:\Users\Prem\Anaconda3\envs\flight\lib\site-packages\seaborn\
distributions.py:2557: FutureWarning: `distplot` is a deprecated
function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar
flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```




```
#skewed = [feature for feature in continious if
data[feature].skew()<1]
#skewed

skewed = []
for feature in continious:
    if abs(df_train[feature].skew())<1:
        skewed.append(feature)
skewed
```

```
['BsmtUnfSF', '2ndFlrSF', 'GarageArea']
```

```
#for feature in continious:
    #if feature == "SalePrice":
        #pass
    #else:
       # df_train[feature] = np.log1p(df_train[feature])

df_train.shape
```

```
(2919, 74)
```

```
# correlation heatmap
plt.figure(figsize=(25,25))
ax = sns.heatmap(df_train[continious].corr(), cmap = "coolwarm",
annot=True, linewidth=2)

# to fix the bug "first and last row cut in half of heatmap plot"
```

```
#bottom, top = ax.get_ylim()
#ax.set_ylim(bottom + 0.5, top - 0.5)
```

```
# correlation heatmap of higly correlated features with SalePrice
hig_corr = df_train[continious].corr()
hig_corr_features = hig_corr.index[hig_corr["SalePrice"] >= 0.45]
hig_corr_features
```

```
abs(-5)
```

Handling categorical variables

```
#categorical = [feature for feature in df_train.columns if
df_train[feature].dtype == "O"]
#len(categorical)
```

```
categorical = []
for feature in df_train.columns:
    if df_train[feature].dtype == "O":
        categorical.append(feature)
len(categorical)
```

```
for feature in categorical:
    df_train.groupby(feature)['SalePrice'].median().plot.bar()
    plt.xlabel(feature)
    plt.ylabel('SalePrice')
    plt.title(feature)
    plt.show()
```

ORDINAL

```
from pandas.api.types import CategoricalDtype
```

```
df_train['BsmtCond'].unique()
```

```
df_train['BsmtCond'] =
df_train['BsmtCond'].astype(CategoricalDtype(categories=['NA', 'Po',
'Fa', 'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
```

```
df_train['BsmtCond'].unique()
```

```
df_train['BsmtExposure'] =
df_train['BsmtExposure'].astype(CategoricalDtype(categories=['NA',
'Mn', 'Av', 'Gd'], ordered = True)).cat.codes
df_train['BsmtFinType1'] =
df_train['BsmtFinType1'].astype(CategoricalDtype(categories=['NA',
'Unf', 'LwQ', 'Rec', 'BLQ','ALQ', 'GLQ'], ordered = True)).cat.codes
df_train['BsmtFinType2'] =
df_train['BsmtFinType2'].astype(CategoricalDtype(categories=['NA',
'Unf', 'LwQ', 'Rec', 'BLQ','ALQ', 'GLQ'], ordered = True)).cat.codes
df_train['BsmtQual'] =
df_train['BsmtQual'].astype(CategoricalDtype(categories=['NA', 'Po',
'Fa', 'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
```

```
df_train['ExterQual'] =
df_train['ExterQual'].astype(CategoricalDtype(categories=['Po', 'Fa',
'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
df_train['ExterCond'] =
df_train['ExterCond'].astype(CategoricalDtype(categories=['Po', 'Fa',
'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
df_train['Functional'] =
df_train['Functional'].astype(CategoricalDtype(categories=['Sal',
'Sev', 'Maj2', 'Maj1', 'Mod','Min2','Min1', 'Typ'], ordered =
True)).cat.codes
df_train['GarageCond'] =
df_train['GarageCond'].astype(CategoricalDtype(categories=['NA', 'Po',
'Fa', 'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
df_train['GarageQual'] =
df_train['GarageQual'].astype(CategoricalDtype(categories=['NA', 'Po',
'Fa', 'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
df_train['GarageFinish'] =
df_train['GarageFinish'].astype(CategoricalDtype(categories=['NA',
'Unf', 'RFn', 'Fin'], ordered = True)).cat.codes
df_train['HeatingQC'] =
df_train['HeatingQC'].astype(CategoricalDtype(categories=['Po', 'Fa',
'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
df_train['KitchenQual'] =
df_train['KitchenQual'].astype(CategoricalDtype(categories=['Po',
'Fa', 'TA', 'Gd', 'Ex'], ordered = True)).cat.codes
df_train['PavedDrive'] =
df_train['PavedDrive'].astype(CategoricalDtype(categories=['N', 'P',
'Y'], ordered = True)).cat.codes
df_train['Utilities'] =
df_train['Utilities'].astype(CategoricalDtype(categories=['ELO',
'NASeWa', 'NASeWr', 'AllPub'], ordered = True)).cat.codes

ordinal = ["BsmtCond" , "BsmtExposure" , "BsmtFinType1" ,
"BsmtFinType2" , "BsmtQual" , "ExterQual" , "ExterCond" ,
"Functional",
          "GarageCond" , "GarageQual" , "GarageFinish" , "HeatingQC" ,
"KitchenQual" , "PavedDrive" , "Utilities"]

len(ordinal)

df_train.shape
```

Nominal

- **One hot encoding**

```
#nominal = [feature for feature in categorical if feature not in
ordinal ]

nominal = []
for feature in categorical:
    if feature not in ordinal:
```

```
        nominal.append(feature)
nominal
```

```
#nominal = [feature for feature in categorical if feature not in
ordinal]
for feature in nominal:
    print(feature , len(df_train[feature].unique()))
```

```
new_nominal = ["Neighborhood" , "Exterior1st" , "Exterior2nd"]
#nominal1 = [feature for feature in nominal if feature not in
new_nominal]
```

```
nominal1 = []
for feature in nominal:
    if feature not in new_nominal:
        nominal1.append(feature)
nominal1
```

```
len(nominal)
```

```
len(nominal1)
```

```
nominal_variable = pd.get_dummies(columns = nominal1 , data =
df_train, drop_first=True)
nominal_variable.drop(new_nominal , axis = 1 , inplace = True)
```

```
nominal_variable.shape
```

- **One hot encoding with many variables**

```
df_train["Neighborhood"].value_counts()
```

```
def top_ten(feature):
    top_ten = []
    for x in feature.value_counts().sort_values(ascending =
False).head(10).index:
        top_ten.append(x)
    return top_ten
```

```
top_10_Neighborhood = top_ten(df_train["Neighborhood"])
top_10_Exterior1st =  top_ten(df_train["Exterior1st"])
```

```
df_train["Exterior1st"].unique()
```

```
df_train["Exterior2nd"].unique()
```

```
#top_10_Neighborhood = [x for x in
df_train.Neighborhood.value_counts().sort_values(ascending=False).head
(10).index]
#top_10_Exterior1st = [x for x in
df_train.Exterior1st.value_counts().sort_values(ascending=False).head(
10).index]
```

```
#top_10_Exterior2nd = [x for x in
df_train.Exterior2nd.value_counts().sort_values(ascending=False).head(
10).index]


for label in top_10_Neighborhood:
    df_train[label]= np.where(df_train["Neighborhood"]==label,1,0)

for label in top_10_Exterior1st:
    df_train[label]= np.where(df_train["Exterior1st"]==label,1,0)

#for label in top_10_Exterior2nd:a
    #df_train[label]= np.where(df_train["Exterior2nd"]==label,1,0)

#df_train[top_10_Exterior2nd].head()

df_train[top_10_Exterior1st].head()

df_train.head()

df_train.drop(["Neighborhood" , "Exterior1st" , "Exterior2nd"] , axis
= 1 , inplace = True)
df_train.drop(nominal1 , axis = 1 , inplace = True)

df_train.head()

train = pd.concat([nominal_variable , df_train] , axis = 1)

train.shape

train.head()

#preview the df
train = train.loc[:,~train.columns.duplicated()]
train.shape

train.head()

train.shape

train.isnull().sum().sum()

train.var()
```

Feature Selection

```
train_df= train[:1460]
test1 = train[1460:]

print(train_df.shape)
print(test1.shape)
#print(len(y_train))
```

```
X = train_df.drop("SalePrice" , axis = 1)
y = train_df["SalePrice"]
test = test1.drop("SalePrice" , axis = 1)

X.shape

len(y)

test1.head()

from sklearn.ensemble import ExtraTreesRegressor
model=ExtraTreesRegressor()
model.fit(X,y)

print(model.feature_importances_)

plt.figure(figsize = (20 , 20))
ranked_features=pd.Series(model.feature_importances_,index=X.columns)
ranked_features.nlargest(35).plot(kind='barh')
plt.show()

features = ranked_features.nlargest(23)

X = train_df[features.index]

X.shape

X.head()
```

Model Building

```
# split dataset into train and test
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size =
0.2, random_state= 5)
```

Robost scaller

```
test1 = test1[features.index]

# scaling dataset with robust scaler
from sklearn.preprocessing import RobustScaler
scaler = RobustScaler()
scaler.fit_transform(X_train)
X = scaler.transform(X_test)
test1 = scaler.transform(test1)
```

Linear Regression

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error # for calculating
mean_squared error
from sklearn.metrics import r2_score # for measering the goodness of
best fit line
```

```
reg = LinearRegression()
reg.fit(X_train , y_train)

y_pred = reg.predict(X_test)
rmse = np.sqrt(mean_squared_error(y_test , y_pred))

score=r2_score(y_test,y_pred)
print(f"value of R^2 is {score}")
print(f"rmse value is {rmse}")
```

Random Forest

```
# Random Forest Classifier
from sklearn.ensemble import RandomForestRegressor


rf = RandomForestRegressor()
rf.fit(X_train, y_train)

y_pred_rf = rf.predict(X_test)
score_rf=r2_score(y_test,y_pred_rf)
rmse = np.sqrt(mean_squared_error(y_test , y_pred_rf))


print(f"value of R^2 is {score_rf}")
print(f"rmse value is {rmse}")
```

Xgboost

```
import xgboost
xgb_model = xgboost.XGBRegressor()
xgb_model.fit(X_train,y_train)


y_pred_xg = xgb_model.predict(X_test)
score_xg=r2_score(y_test,y_pred_xg)
rmse = np.sqrt(mean_squared_error(y_test , y_pred_xg))


print(f"value of R^2 is {score_xg}")
print(f"rmse value is {rmse}")

from sklearn.model_selection import cross_val_score
cross_validation = cross_val_score(estimator = xgb_model, X =
X_train,y = y_train, cv = 10)
print("Cross validation accuracy of Xgboost model = ",
cross_validation)
print("\nCross validation mean accuracy of Xgboost model = ",
cross_validation.mean())

y_pred_hyper = rf.predict(test1)
y_pred_hyper
```

```
df = pd.read_csv("test.csv" , usecols = ["Id"])
```

```
df.head()
```

```
submit_test1 = pd.concat([df["Id"], pd.DataFrame(y_pred_hyper)],
axis=1)
submit_test1.columns=['Id', 'SalePrice']
```

```
submit_test1.head(20)
```

```
submit_test1 = submit_test1.astype({'Id': 'int', 'SalePrice':
'float'})
```

```
submit_test1.to_csv('sample_submission.csv', index=False)
```

```
df = pd.read_csv("sample_submission.csv")
df
```

Hyper parameter tuning

```
## Hyperparameter optimization using RandomizedSearchCV
from sklearn.model_selection import RandomizedSearchCV
```

```
#Randomized Search CV

# Number of trees in random forest
n_estimators = [int(x) for x in np.linspace(start = 100, stop = 1200,
num = 12)]
# Number of features to consider at every split

criterion = ["mse" , "mae"]

max_features = ['auto', 'sqrt', "log2"]
# Maximum number of levels in tree
max_depth = [int(x) for x in np.linspace(5, 30, num = 6)]
# Minimum number of samples required to split a node
min_samples_split = [2, 5, 10, 15, 100]
# Minimum number of samples required at each leaf node
min_samples_leaf = [1, 2, 5, 10]
```

```
# Create the random grid

random_grid = {'n_estimators': n_estimators,
               'max_features': max_features,
               'max_depth': max_depth,
               'min_samples_split': min_samples_split,
               'min_samples_leaf': min_samples_leaf}
```

```
rf_random = RandomizedSearchCV(estimator = rf, param_distributions =
random_grid,scoring='neg_mean_squared_error', n_iter = 10, cv = 5,
verbose=2, random_state=42, n_jobs = -1)
```

```
rf_random.fit(X_train,y_train)
```

```
rf_random.best_params_

prediction = rf_random.predict(X_test)
score_rf=r2_score(y_test,prediction)


print(f"value of R^2 is {score_rf}")
print('RMSE:', np.sqrt(mean_squared_error(y_test, prediction)))

y_pred_hyper = rf_random.predict(test1)
y_pred_hyper

df = pd.read_csv("test.csv" , usecols = ["Id"])
submit_test1 = pd.concat([df["Id"], pd.DataFrame(y_pred_hyper)],
axis=1)
submit_test1.columns=['Id', 'SalePrice']

submit_test1 = submit_test1.astype({'Id': 'int', 'SalePrice':
'float'})
submit_test1.to_csv('sample_submission.csv', index=False)

submit_test1
```