Houses Analysis Report:

1)Data Exploration:

First, here's an explanation for each of the columns to understand what they really represent:

- **Id**: A unique identifier for each house.
- **MSSubClass**: The building class of each house.
- **MSZoning**: The general zoning classification of each house.
- **LotFrontage**: The linear feet of street connected to each property.
- **LotArea**: The lot size of each property.
- **Street**: The type of road access to each property.
- **Alley**: Type of alley access to each property.
- **LotShape**: The general shape of each property.
- **LandContour**: The flatness of the property.
- **Utilities**: Type of utilities available to each property.
- **LotConfig**: The configuration of each property.
- **LandSlope**: The slope of the property.
- **Neighborhood**: Physical locations within Ames city limits.

- **Condition1**: Proximity to various conditions such as a busy street or railroad.
  Condition2: Proximity to various conditions (if more than one is present).
- **BldgType**: The type of dwelling within each house.
- **HouseStyle**: The style of dwelling within each house.
- **OverallQual**: Overall material and finish quality of the house.
- **OverallCond**: Overall condition rating of the house.
- **YearBuilt**: Original construction date of the house.
- **YearRemodAdd**: Remodel date (same as construction date if no remodeling or additions).
- **RoofStyle**: Type of roof on the house.
- **RoofMatl**: Roof material on the house.
- **Exterior1st**: Exterior covering on the house (most common).
- **Exterior2nd**: Exterior covering on the house (if more than one material).
- **MasVnrType**: Masonry veneer type on the house.
- **MasVnrArea**: Masonry veneer area in square feet.
- **ExterQual**: Exterior material quality rating for the house.
- **ExterCond**: Present condition of the material on the exterior for the house.

- **Foundation**: Type of foundation for the house.

- **BsmtQual**: Height of the basement for the house (if present).
- **BsmtCond**: General condition of the basement for the house (if present).
- **BsmtExposure**: Walkout or garden level basement walls for the house (if present).
- **BsmtFinType1**: Quality of basement finished area for the first type (if present).
- **BsmtFinSF1**: Type 1 finished square feet in basement for the house (if present).
- **BsmtFinType2**: Quality of second finished area (if present) in basement for the house (if present).
- **BsmtFinSF2**: Type 2 finished square feet in basement for the house (if present).
- **BsmtUnfSF**: Unfinished square feet of basement area for the house (if present).
- **TotalBsmtSF**: Total square feet of basement area for the house (if present).
- **Heating**: Type of heating in the house.
- **HeatingQC**: Heating quality and condition rating in the house.
- **CentralAir**: Central air conditioning in the house (yes or no).
- **Electrical**: Electrical system in the house.
- **1stFlrSF**: First Floor square feet in the house.
- **2ndFlrSF**: Second floor square feet in the house.
- **LowQualFinSF**: Low quality finished square feet in all floors except basement and attic for the house.
- **GrLivArea**: Above grade (ground) living area square feet in the house.

- **BsmtFullBath**: Basement full bathrooms in the house (if present).
- **BsmtHalfBath**: Basement half bathrooms in the house (if present).
- **FullBath**: Full bathrooms above grade in the house.
- **HalfBath** : Half baths above grade in the house
- **BedroomAbvGr** : Number of bedrooms above basement level
- **KitchenAbvGr** : Number of kitchens above basement level
- **KitchenQual** : Kitchen quality rating
- **TotRmsAbvGrd** : Total rooms above grade (does not include bathrooms)
- **Functional** : Home functionality rating
- **Fireplaces** : Number of fireplaces in a home
- **FireplaceQu** : Fireplace quality rating
- **GarageType** : Garage location
- **GarageYrBlt** : Year garage was built
- **GarageFinish** : Interior finish of garage
- **GarageCars** : Size of garage in car capacity
- **GarageArea** : Size of garage in square feet
- **GarageQual** : Garage quality rating
- **GarageCond** : Garage condition rating

It’s also essential to know the exact type of each feature, so here’s a list of their types:

- **Id**: integer
- **MSSubClass**: integer
- **MSZoning**: string
- **LotFrontage**: float
- **LotArea**: integer
- **Street**: string
- **Alley**: string
- **LotShape**: string
- **LandContour**: string
- **Utilities**: string
- **LotConfig**: string
- **LandSlope**: string
- **Neighborhood**: string
- **Condition1**: string
- **Condition2**: string
- **BldgType**: string
- **HouseStyle**: string

- **OverallQual**: integer
- **OverallCond**: integer
- **YearBuilt**: integer
- **YearRemodAdd**: integer
- **RoofStyle**: string
- **RoofMatl**: string
- **Exterior1st**: string
- **Exterior2nd**: string
- **MasVnrType**: string
- **MasVnrArea**: float
- **ExterQual**: string
- **ExterCond**: string
- **Foundation**: string
- **BsmtQual**: string

- **BsmtCond**: string
- **BsmtExposure**: string
- **BsmtFinType1**: string
- **BsmtFinSF1**: float

- **BsmtFinType2**: string
- **BsmtFinSF2**: float
- **BsmtUnfSF**: float
- **TotalBsmtSF**: float
- **Heating**: string
- **HeatingQC**: string
- **CentralAir**: string (yes or no)
- **Electrical**: string
- **1stFlrSF** : integer
- **2ndFlrSF** : integer
- **LowQualFinSF** : integer
- **GrLivArea** : integer
- **BsmtFullBath** : integer
- **BsmtHalfBath** : integer
- **FullBath** : integer
- **HalfBath** : integer
- **BedroomAbvGr** : integer
- **KitchenAbvGr** : integer
- **KitchenQual** : string
- **TotRmsAbvGrd** : integer
- **Functional** : string
- **Fireplaces** : integer
- **FireplaceQu** : string
- **GarageType** : string
- **GarageYrBlt** : float
- **GarageFinish** : string
- **GarageCars** : integer
- **GarageArea** : float
- **GarageQual** :string
- **GarageCond**
- **PavedDrive**: string
- **WoodDeckSF**: integer
- **OpenPorchSF**: integer
- **EnclosedPorch**: integer
- **3SsnPorch**: integer

- **ScreenPorch**: integer PoolArea: integer
- **PoolQC**: string
- **Fence**: string
- **MiscFeature**: string

- **MiscVal**: integer
- **MoSold**: integer
- **YrSold**: integer
- **SaleType**: string
- **SaleCondition**: string
- **SalePrice**: integer

2)Data processing

Date Loading

First of all, when working with a dataset we should get familiar with it and get to know its special features. If any of those features may cause problems in the future, we should be prepared ahead for handling them. In order to do that, we get the list of our data frame's columns, its shape (which includes its number of rows and columns), its head (which gives the first few rows of the whole table to have a clearer vision of the table), its tail (shows us the few last rows of the table) and so on.

Df.head() : Gives the first few rows of our table.

Df.tail : Gives the last few rows of our table.

Df.shape : gives the table's size, which is the number of its rows and columns.

Df.columns : gives us the list of the columns' names(name of features).

Df.dtypes: gives us the list of the type of each column of the table.

Df.describe(): Provides a summary of the central tendency, dispersion and shape of our dataset. In other words, gives us some detailed statistical information of each column, excluding NaN values:

- **count**: The number of non-empty values.
- **mean**: The average (mean) value.

- **std**: The standard deviation.
- **min**: The minimum value.
- **25%**: The 25th percentile.
- **50%**: The 50th percentile.
- **75%**: The 75th percentile.
- **max**: The maximum value.

Null Values

Now that our dataset is not a mystery to us anymore, the first thing to be worried about is null values. In some datasets there may exist some columns or features that their percentage of null values is very high. In this case even if we try to fill in the null values with different methods or strategies, it would be a waste of time. The main reason for that is because we don't really have enough information or data to work with and fill in the null values. For example, the "MiscFeature" in our data frame has the percentage of over 90 for null values, in this case we can just drop the column since it doesn't have much worth for us.Just like what we did for "MiscFeature", we just drop other columns that have so many null values.

Columns we dropped because of high percentage of null values:

MiscFeature: 96.3 %

Fence: 80.75%

PoolOC: 99.52%

Alley: 93.77%

FirePlaceQu: 47.26%

Normality

The next thing I preferred to do was checking the distribution of features, and see if they're normal or not. There are many tests available to check the normality of a feature's distribution, I used two of them: Anderson-Darling and Shapiro-Wilk.

Anderson-Darling test:

The Anderson-Darling test is a statistical test used to determine whether a given sample of data is drawn from a given probability distribution . The test is most commonly used to test whether a sample of data comes from a normal distribution, but it can also be used to test other distributions as well .The test statistic is then compared to critical values from a table, which depend on the significance level and sample size . If the test statistic is greater than the critical value, then we reject the null hypothesis that the sample comes from the hypothesized distribution .

Shapiro-Wilk test:

The Shapiro-Wilk test is a statistical test used to determine whether a given sample of data is drawn from a normal distribution . The test is based on the difference between the observed distribution and the expected normal distribution. The test statistic is calculated as the sum of the squared differences between the observed and expected values, weighted by the expected variance .

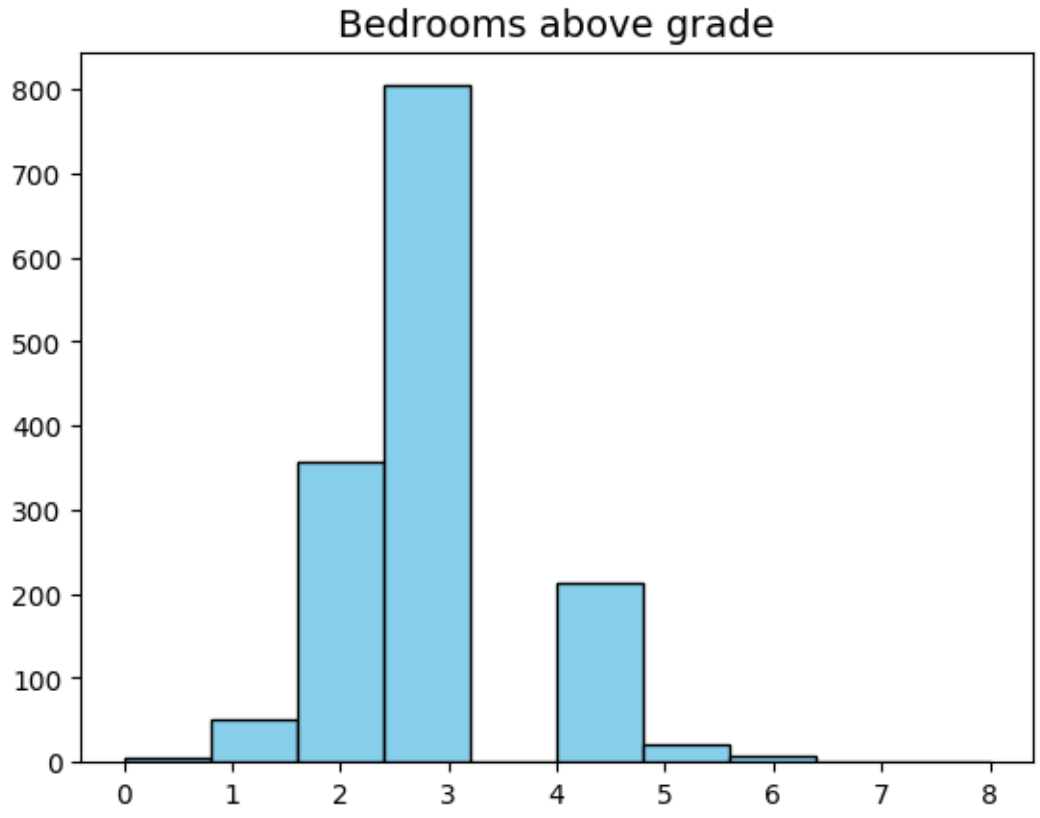
The Shapiro-Wilk test is commonly used to test whether a sample of data comes from a normal distribution. The null hypothesis of the test is that the sample comes from a normal distribution. If the p-value of the test is less than the significance level (usually 0.05), then we reject the null hypothesis and conclude that the sample does not come from a normal distribution .

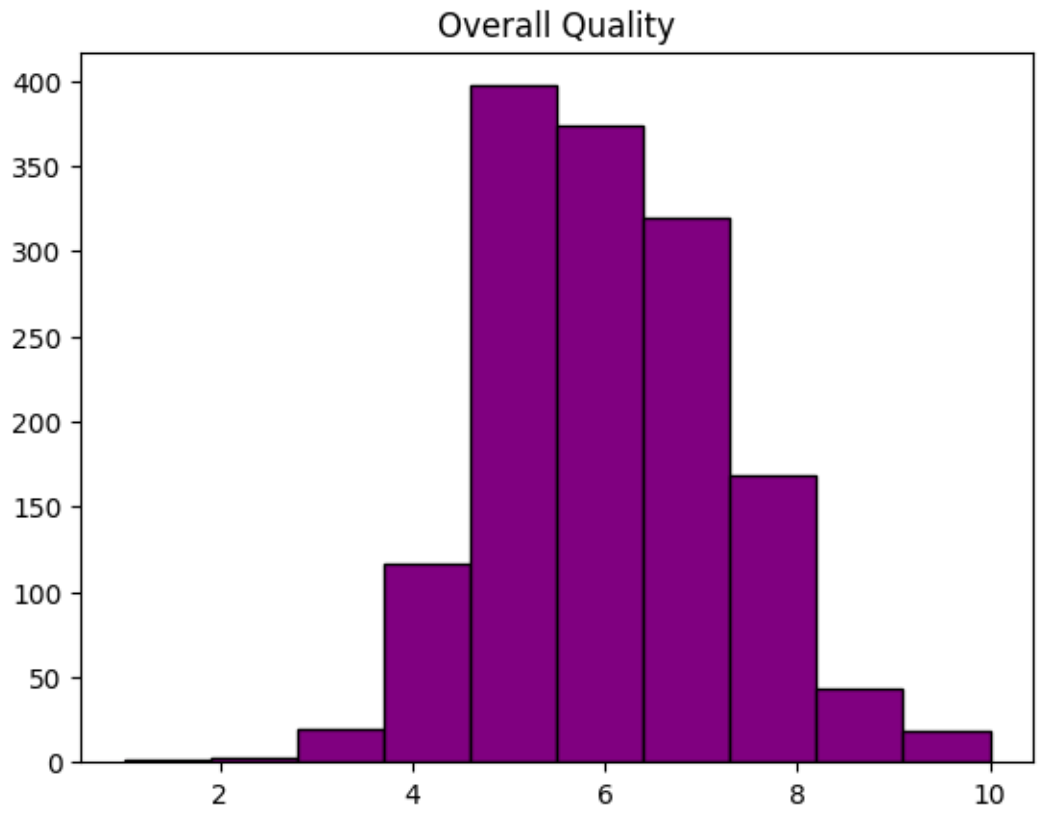
The Shapiro-Wilk test is more powerful than other tests of normality, such as the Kolmogorov-Smirnov test, for small to moderate sample sizes . However, it can be less powerful than other tests for large sample sizes .

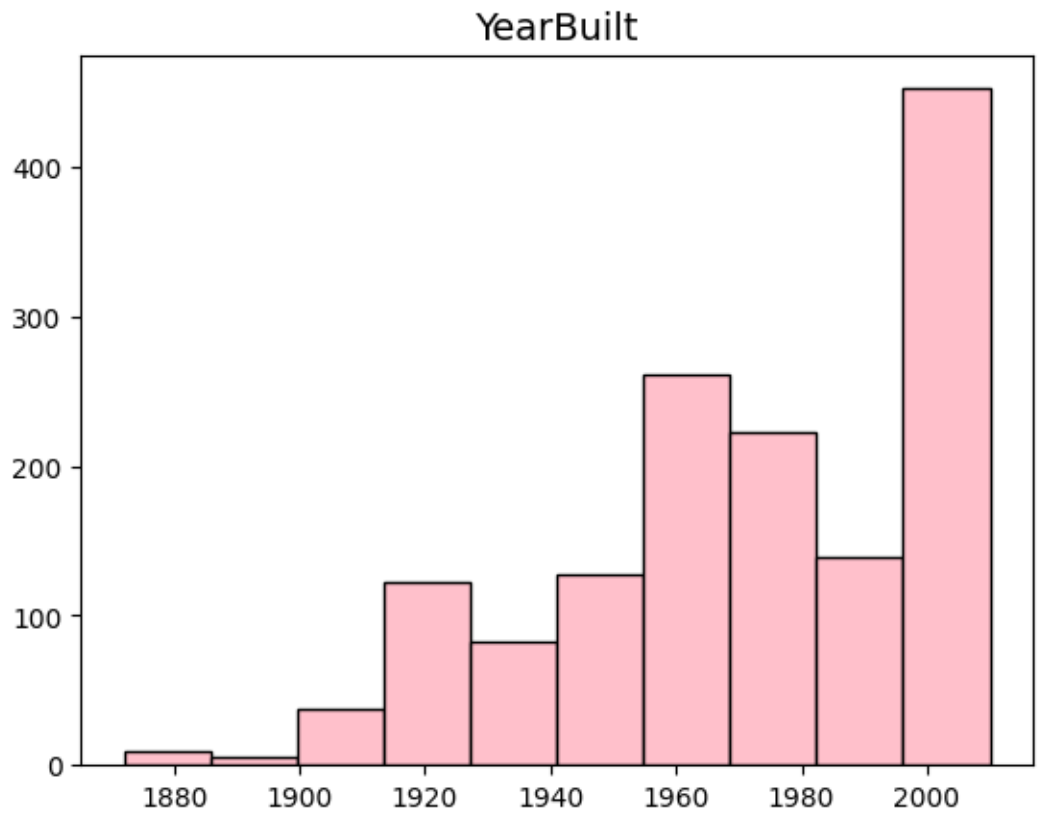
Visualization-check:

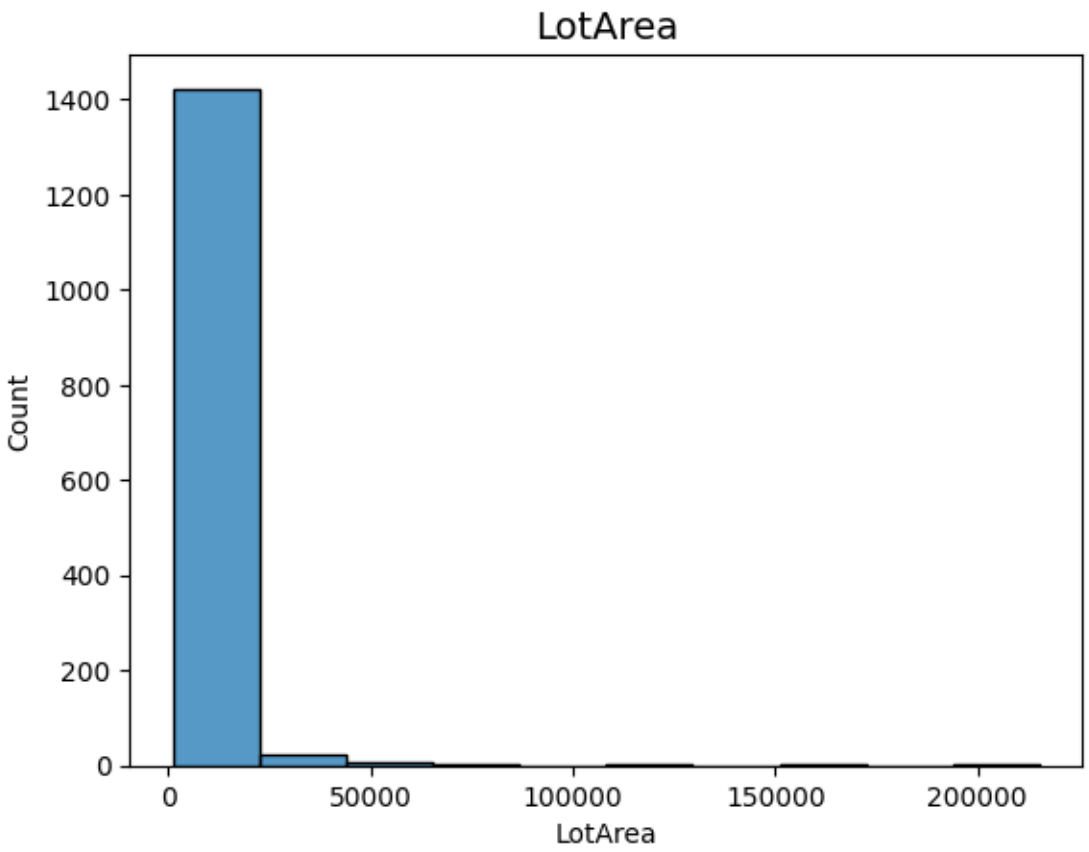
Another way for checking the distribution of a feature is to check it visually by plotting that feature (Like plotting a histogram for it). This method is definitely less accurate that the tests mentioned above but I always prefer to have a clear image of what the data looks like.

For Anderson-Darling normality test I examined only some of the features like: Overall Quality, Bedroom above grade and Year built.And for Shapiro-Wilk test, I chose Lot area and the plotted it using seaborn's histogram. Expectedly, none of the features had a normal distribution.




Overall Quality
400
350
300
250
200
150
100
50
0
2
4
6
8
10


YearBuilt
400
300
200
100
0
1880
1900
1920
1940
1960
1980
2000



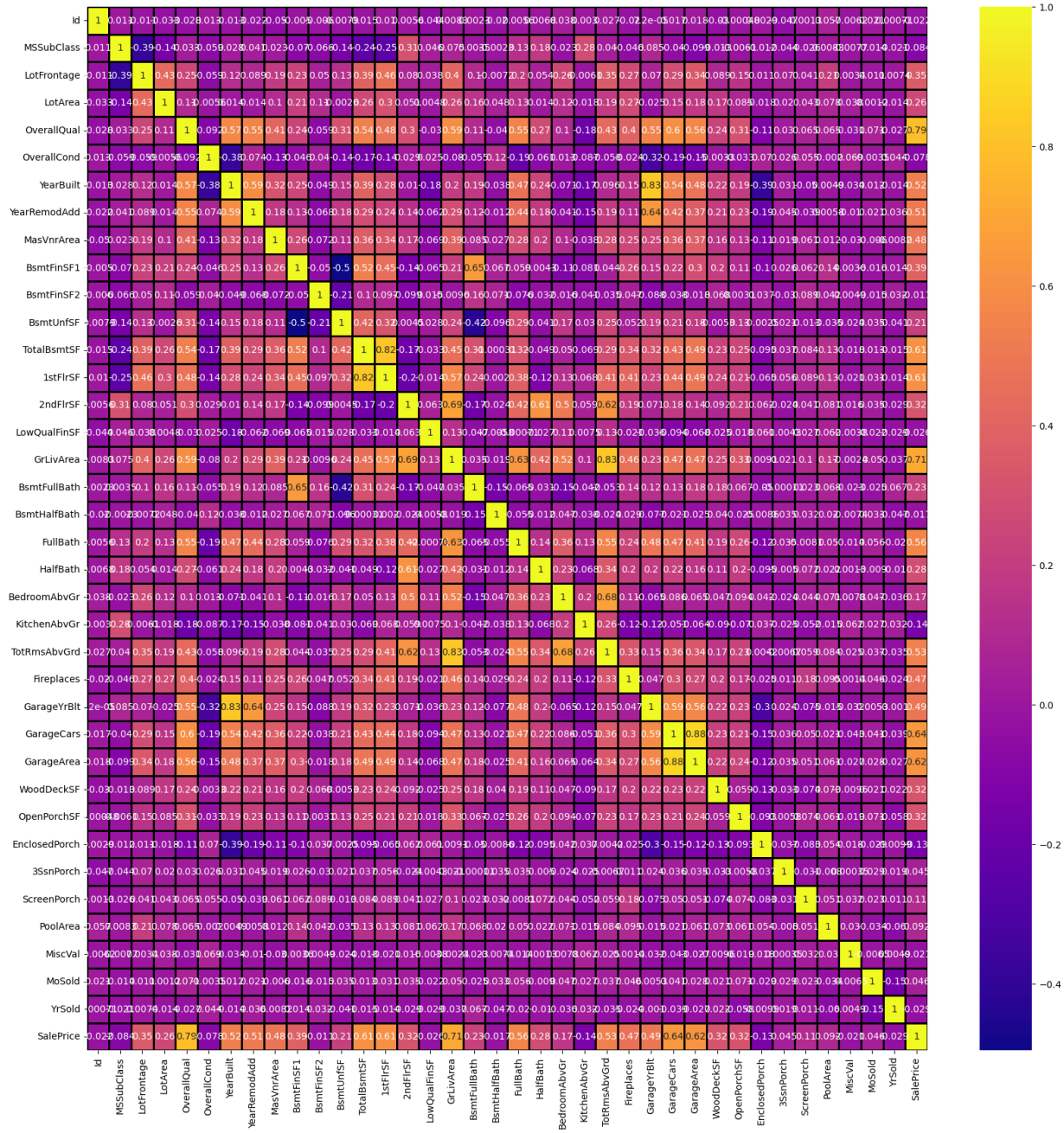
Correlation

In this section we use some correlation test to ensure if some of the features are correlated or not.

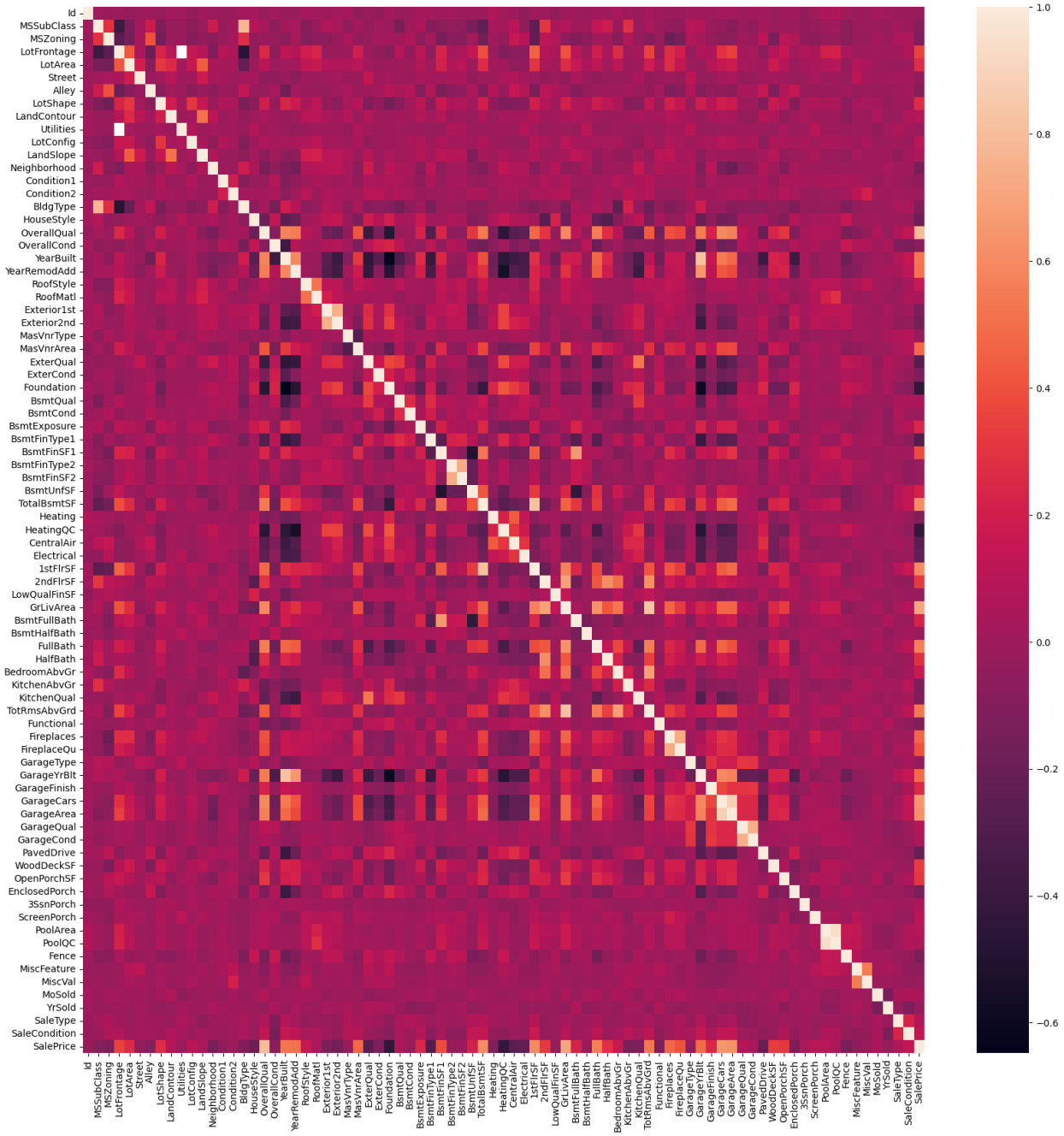
Correlation heatmap:

A correlation heatmap is a graphical representation of the correlation between multiple variables in a dataset. It is a color-coded matrix that shows how closely related different variables are. Each variable is represented by a row and a column, and the cells show the correlation between them. The values in the cells are color-coded to indicate the strength of the correlation, with darker colors indicating stronger correlations.

I Plotted a heatmap specifically for numerical columns :



And then another for all of the table's columns:



Correlation tests:

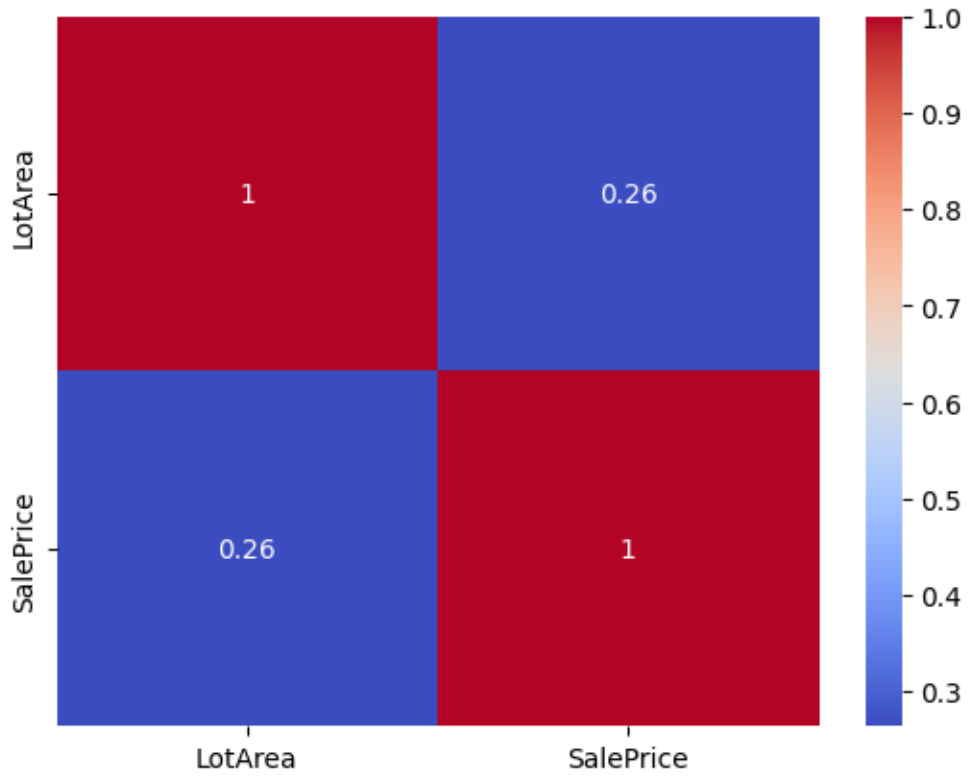
Correlation tests are statistical tests used to determine whether two variables are related to each other, and if so, how strongly they are related . Correlation tests can be used to test whether two variables have a positive or negative relationship, or whether they are unrelated.

Pearson test:

The Pearson correlation test is a statistical test used to determine the strength and direction of the linear relationship between two continuous variables . The test measures the degree to which the two variables are related to each other, and it produces a correlation coefficient that ranges from -1 to 1.

So I applied this test on MSSubClass and Sale Price and it turned out that there's not a very strong correlation between them. The same thing was seen for Lot Area and Sale Price as well.

Here's the heatmap fpr Sale Price and Lot Area:



Anova test:

The ANOVA test (Analysis of Variance) is a statistical test used to determine whether there are any statistically significant differences between the means of two or more groups . The test compares the variance between the groups to the variance within the groups to determine whether the differences between the groups are significant or not.

I used one way anova test here to figure out how much different Neighborhood and Sale Price are. It turned out that their difference is significant and as a result of that it couldn't have happened just by chance and therefore they must be correlated in some way.

T-test

A t-test is a statistical test used to compare the means of two groups of data and determine whether they are significantly different from each other. It is often used in hypothesis testing to determine whether a process or treatment actually has an effect on the population of interest, or whether two groups are different from one another.

In this section, I applied t-test on Sale Price and 2ndFlrSF, and the result lead me to the conclusion that there's also a relation between these two columns.

The End.