# Statistical Inference

**Minerva Schools at KGI**

**Jacob Rotich**

**CS50 Fall 2019**

## Introduction

In my data, I explore the prices of houses, based on different locations, sizes and even by the number of floors. I compare two variables set,how prices influence the number of floors of a house.I take samples of a number of houses and by using the confidence intervals, I estimate the prices of houses based on the number of floors they have. We also perform a difference in means to test for statistical and practical significance to determine. Overally, perhaps the comparison would be an insight to buyers, that before buying a house they may analyze this data sets and predict predetermined prices before purchasing a home. They may also avoid being overcharged by middlemen, by analyzing the selling value versus the predicted value of a house.

## Dataset and Methods

From the numerous options we were given from our assignment instructions, I chose one of the datasets about house pricing, (CSV of dataset can be found in the zipped folder, I have also imported part of the table in Appendix A, below.) This a sample collected by House Sales in King County.

The dataset was read into Python using the pandas package for analysis. Appendix A below represents only a part of the about 20,000 samples collected.

## Variables

In the data set, we may identify the independent variable as 'the number of floors in the house,' manipulating the floors of the house, we may get to know the prices of the house, which is the dependent variable. The independent and dependent variables tend to have an independent relation. The confounding variable in the dataset is the size and location of the house and may influence the relations of the independent variables. For example, whether old or new, a larger house tends to be more expensive than the newer houses. Expensive locations also tend to have their houses more expensive.

We can classify variables further to understand them.

The number of floors- is a discrete variable and quantitative. (Quantitative as it gives quantities in numbers and discreet as we can't use decimals or fractions in them: it gives data in terms of exact numbers, example 4 floors, 31 floors)

Prices are quantitative and Continuous- (We don't have exact prices for houses, a house price may, for example, be $84948.77 and its units is in dollars)

The size of the house is a continuous qualitative variable, which describes variables in terms of square feet.

The location of the house is a qualitative variable- Since it is a description of a place across the country without using numbers[1].

[1] **#Variables-** I try to identify all types of variables, the dependent, independent and the confounding variables. I classify them in terms of quantitative or qualitative data and identify the relations between the dependent and independent variable as independent.

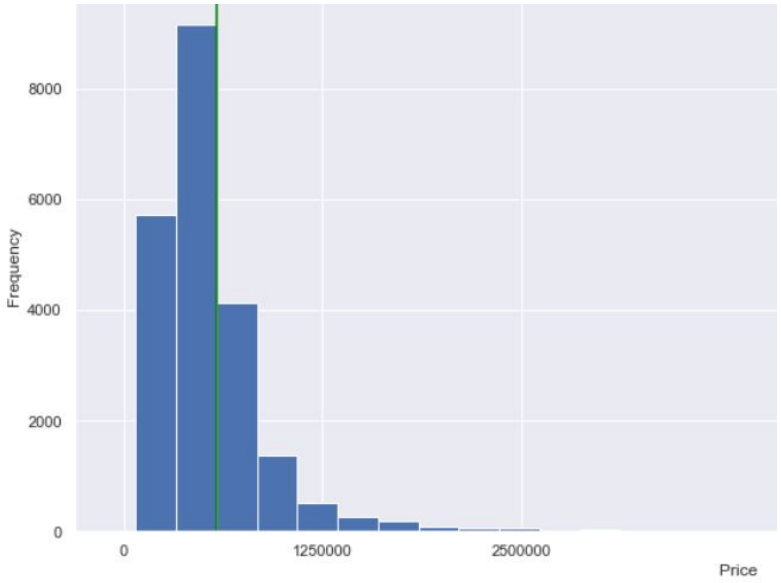
From the data set (Appendix A), I calculated different statistical measures such as mean and mode. I have outlined them in the table below. These are all sampled data sizes and are representative of the whole population. My results can be explained by a code screenshot, in appendix B[2]

| Statistical Measures/Variables | Price of the House(in dollars) | Floors |
|---|---|---|
| Mean (Total/no. of entries) | 540,102 | 1.494 ≅ 2 |
| Mode (Highest Frequency) | 350000 | 1.0 |
| Median (Middle value) | $450,000 | 1.5 ≅ 2 |
| Standard Deviation | 367120.81 | 0.5399 |
| Range (Max.- Min.) | 7625000 | 2.5 |
| | | |

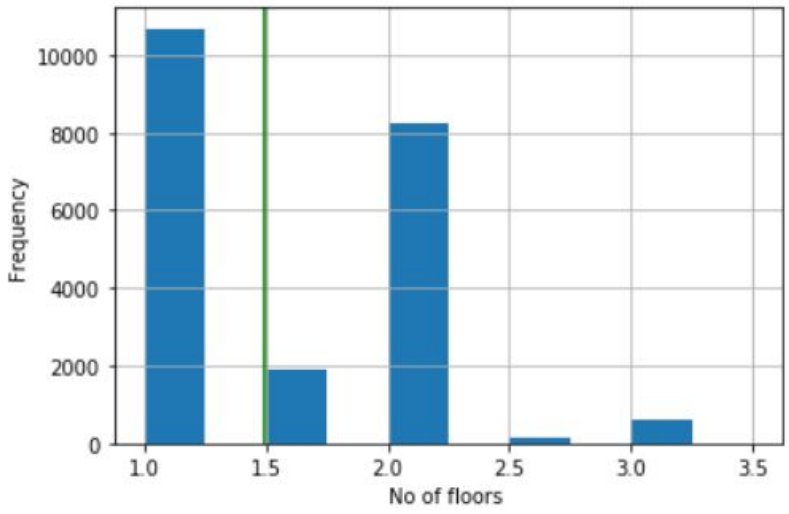
From the sample I had, I construct a histogram to represent the two variables, the prices of houses and the floors of the building. The histogram below show the price of the houses against

[2] **#Algorithms-** I choose an appropriate visualization, I write a code that gives rise to a Histogram, plus comment a code in an understandable way, with correct steps.

the frequency. The code Application may be found in Appendix C. The histogram that follows that then shows the floors of the houses against  Frequency.




(Figure 1. It is a graph of frequency against Prices of Houses. Maximum prices of houses were about $250000, the bar with the lowest size. The green line shows location of mean.)



(Figure 2. It is a graph of frequency against the Number of Floors. It is derived from a sample of 21612 houses, and the y-axis describes how frequent each number of floors appear. Green line indicates location of mean)

Before constructing histograms, I used the datasets for interpretations[3], I imagined getting a normal distribution, as by skimming through the data, I may not have noticed some outliers that would influence the histogram's shape.

Characteristics of the histograms: The first histogram is slightly skewed to the right, with the mean slightly higher than the median. The second histogram is quite bimodal with two peaks(Highest points at two particular places(At 1.0 and 2.0) From the sample data, houses with 1 floor, appear the most and $350000 is the typical house prices. A range of 7625000 for house prices makes the datasets not close to normal distribution and makes a median better choice of representation. The range also is responsible for dragging the mean quite away from the median, more to the right. In trying to evaluate whether a buyer should use the floor number to find the predetermined house price won't be a good idea because some houses with one floor were more expensive than houses with more floors due to the confounding variables we had[4].

To estimate how the pricing of houses, we compute a confidence interval for the mean prices of houses in which provides a good range of values. I use a statistical interval of 95%(0.95). From the histogram, I check if these assumptions are met, such as having independent variables, and

[3] **#Dataviz-** I plot a histogram and try to make it easily understandable by labelling the axes, and use clear units.
[4] **#Descriptive Stats-** I try to predict how would like before plotting it on histogram, use a histogram later to describe various aspects of the data like range and median and mean and try to find out the influences they may have on a buyer's particular choices. I also try to explain the shape of the histogram.

the size should be large enough. It also happens that they are higher than the rule of thumb(>30) because I use about 20,000 samples of existing data, and check skewness of graph. You can employ the Central Limit Theorem to conclude that the sampling distribution of the sample mean (with sample size 20000) is approximately normal[5], with a mean of 540102 dollars and a standard error of 367120. This allows us to accurately compute, using t-scores(Due to slight skewness) Looking at the histogram of house prices, I find it skewed to the right. Considering the large sample of 20,000 house prices, the probability of getting a price that is on the right side of the histogram is minimal as they only represent outliers(the modes are on the left sides). The few outliers represent the few samples of houses that are very expensive.

My Confidence Intervals is calculated from the python library, as seen in Appendix D. My lower bound for the price of houses is 535193.3812835508, and my upper bound is 544982.9022495081, which seems plausible. This is because my sample mean 540102, and is inside the confidence interval[6]. The minimal differences that exist between them are contributed by the high number of degrees of freedom I used, from (n-1), where n= 9023(Lowest number from the two data sets).

The S. D I got= Standard Error, which equals 367120(indicated in table earlier).

As I try to find out whether high buildings are more expensive than less tall buildings, I try to define what a tall building is.

---

[5] #**Distributions-** I stated the function of Central limit theory, and give choice of distributions and predictions on taking another sample data. I also calculate relevant calculations and give reason for choosing my distribution

[6] **#Confidence Intervals**- I follow correct ways of finding confidence intervals, as shown in my appendix below and justify reasons for using 2 tailed graph and in my conclusion I justify what a 95% confidence intervals is

A tall building>=2 floors

A short, tall building is the one with less than 2 floors.

My null hypothesis is that taller buildings are more expensive than the shorter buildings. (H0= (m1-m2=0, my sample mean prices of homes should be zero)

My null hypothesis is taller buildings are not more expensive than the shorter buildings.H1=(m1-m2 differences in means is not equal to 0)

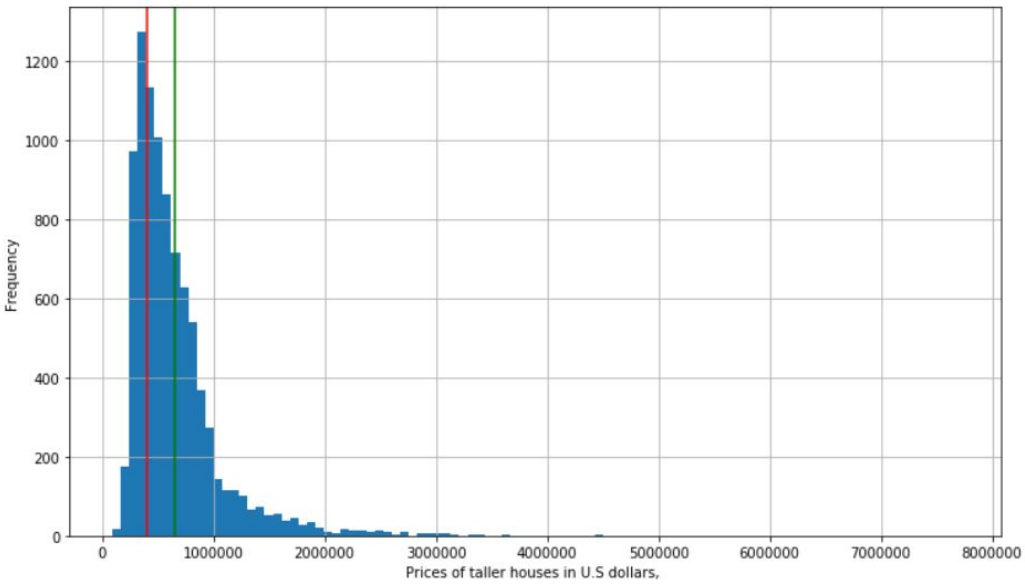
I would use the two tail testing because I would like to find the sizes on both sides. Since my confidence Intervals is 95%, I am using significance levels as 0.05. The consequences of making a type 1 error here would be wrong because from looking at the data, there have been cases of an increasing number of house prices with the number of floors. Therefore rejecting that taller buildings are not more expensive would quite not right.

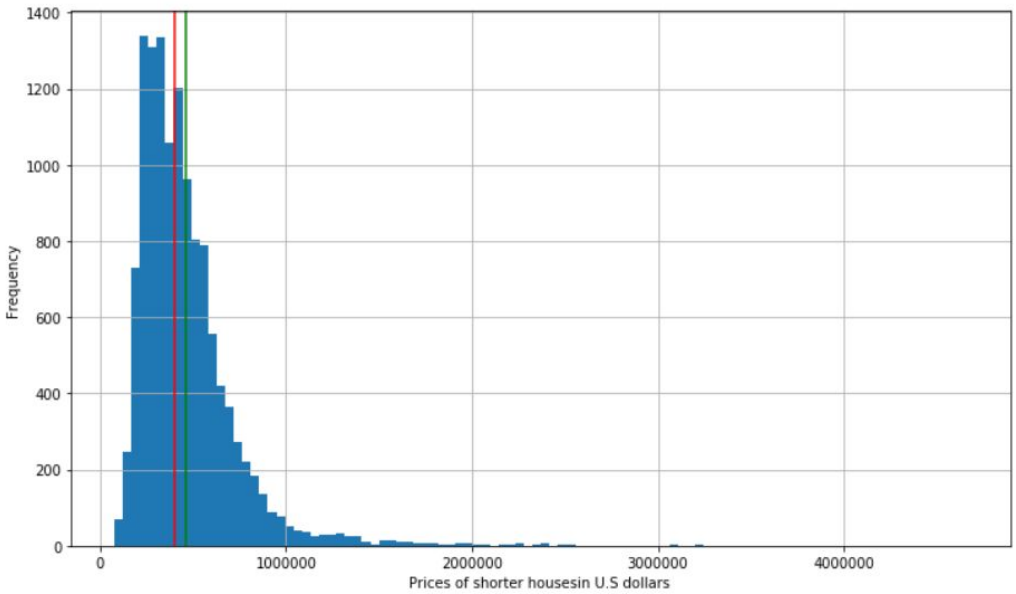
I construct a table to explain the second data sets better.

| | (Number of floors in the Houses given in (x)) | |
|---|---|---|
| | x<2 | x>=2 |
| Count | 12590 | 9023 |
| Mean | 538714.12 | 587045.02 |
| Standard deviation | 53715.06 | 367418.63 |
| Difference in mean | 192076.28 | 192076.28 |
| | | |

From the statistics, I collected I now compare the distribution of taller and shorter buildings, from the histograms below. The codes of explanations are in Appendix E and F

(Figure 3 below is a histogram of frequency against the prices of housing for taller buildings of 2 or more floors. It is derived from the sample of 21612 houses, and the y-axis describes how frequent each number of floors appear.The red line shows location of median while the green location of mean)



(Figure 4 above . It is a graph of frequency against the Number of prices for shorter houses.The red line shows location of median while the green location of mean)

I now use the t distribution and the difference of mean since I do not have the whole population's Standard Deviations and to avoid the bias that may accompany the z score.

**Statistical and Practical Significance**

To assess statistical significance, we calculate a p value. This can be computed using an online calculator online using the degrees of freedom of the lowest number as 9023 and 0.05 as the significance level. I first find the t value which equals= 1.9602, for a two tailed histogram. From another online calculator, I calculate the P value score  The P value= 0.04997. This slightly lower  than the alpha value of 0.05, so we I reject the null hypothesis.

To assess practical significance, we need a measure of effect size. Here, I use the Cohen's D because the data does not defer by a very great standard deviations. ( The upper bound is 367416.3936043793 and the Deviation of the second data: 353715.06155102566)
Using Cohen's D to measure the effect size equals, 394598.37747129716/ pooled Standard deviation.
The pooled Standard Deviation= ((S.Dev1**2) + (S.Dev 2**2))/2 where (S.Dev**2)= Variance
From the results I got, Python Functioning in appendix ..()
= (134996455967.65634 + 837978933593.75)/2
Pooled S.Dev= 972975389561
Cohen's D= 394598.37747129716 / 972975389561

=4*10^-7, This is lower than the 0.2. The very small difference depicts that it does not have practical significance.

**Conclusions**

From the set of reasonings,

I used a substantial sample of about 20000

I only used one sampling distribution

A higher sample size leads to a very lower standard error.

The P-value of the Statistical Significance is slightly less than the alpha level of 0.05.

Cohen's D value of the practical Testing is less than alpha of 0.02 hence not practical.

The null hypothesis would hold.

Therefore my null hypothesis that taller buildings are not more expensive than the shorter would be relevant. This can be inferred to a larger population as it is a more significant sample. My induction is strong but maybe very reliable to a larger population because I take though I take samples that are most likely true, I only used one sampling distribution, (I would require multiple sampling distribution to justify reliability)[7].

Word Count: 1547

[7] **#Induction-** I logically write my set of arguments in a way that can give answers/ an inductive answer at the end and state whether my reasoning holds. I also write my arguments logically

**References**

Harlfoxem. (2016, August 25). House Sales in King County, USA. Retrieved from https://www.kaggle.com/harlfoxem/housesalesprediction#kc_house_data.csv.

## Appendix

A full Jupyter notebook can be accessed here at the appendix section.

## Appendix A

In [5]:
```
import csv  #Imports the comma seperated values from a csv file.
import pandas as pd
import numpy as np #Invaluable to calculate descriptive stats such as mean, mode, and median
import matplotlib.pyplot as plt #creates a figure, plots some lines in a plotting area


data = pd.read_csv('kc_house_data.csv') #I represent the name of my data,with another name(data) pd.read.csv is a function th
data = data[1:] #Seperates the datum and the title bars(Title bars has date, price, bedrooms, bathrooms..)
data #shows the data in a table
```

Out[5]:

| date | price | bedrooms | bathrooms | sqft_living | sqft_lot | floors | waterfront | view | ... | grade | sqft_above | sqft_basement | yr_built | yr_renovated | zipco |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000000 | 538000.0 | 3 | 2.25 | 2570 | 7242 | 2.0 | 0 | 0 | ... | 7 | 2170 | 400 | 1951 | 1991 | 981 |
| 000000 | 180000.0 | 2 | 1.00 | 770 | 10000 | 1.0 | 0 | 0 | ... | 6 | 770 | 0 | 1933 | 0 | 980 |
| 000000 | 604000.0 | 4 | 3.00 | 1960 | 5000 | 1.0 | 0 | 0 | ... | 7 | 1050 | 910 | 1965 | 0 | 981 |
| 000000 | 510000.0 | 3 | 2.00 | 1680 | 8080 | 1.0 | 0 | 0 | ... | 8 | 1680 | 0 | 1987 | 0 | 980 |
| 000000 | 1225000.0 | 4 | 4.50 | 5420 | 101930 | 1.0 | 0 | 0 | ... | 11 | 3890 | 1530 | 2001 | 0 | 980 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 000000 | 360000.0 | 3 | 2.50 | 1530 | 1131 | 3.0 | 0 | 0 | ... | 8 | 1530 | 0 | 2009 | 0 | 981 |
| 000000 | 400000.0 | 4 | 2.50 | 2310 | 5813 | 2.0 | 0 | 0 | ... | 8 | 2310 | 0 | 2014 | 0 | 981 |
| 000000 | 402101.0 | 2 | 0.75 | 1020 | 1350 | 2.0 | 0 | 0 | ... | 7 | 1020 | 0 | 2009 | 0 | 981 |
| 000000 | 400000.0 | 3 | 2.50 | 1600 | 2388 | 2.0 | 0 | 0 | ... | 8 | 1600 | 0 | 2004 | 0 | 980 |
| 000000 | 325000.0 | 2 | 0.75 | 1020 | 1076 | 2.0 | 0 | 0 | ... | 7 | 1020 | 0 | 2008 | 0 | 981 |

## Appendix B

In [14]:
```
import scipy.stats #Invaluable to calculate mathematical functions like mode.
import numpy #Will be also helpful to calculate discriptive stats such as mean, median
import statistics #Calculate the descriptive stats.
#(Arithmetic Mean)
#The functions generally add the numbers in the lists and divide it by the number of entries.
mean1= numpy.mean(data['price']) #The inbuilt calculator solves (Total number of values/ number of entries)
mean2=numpy.mean(data['floors']) #The  function finds mean of the 'data' folder. It specifically finds mean of the ["floors"]
print ("Mean price:", mean1) # Prints the mean price of houses
print ("Mean number of floors:", mean2) # Prints the mean of floors
#(Median)
#Numpy This function generally sorts data, from the smallest to largest and finds the middle one. If the lists has an even nu
median1= numpy.median(data['price']) #The inbuilt function Finds the median(Middle value)
Median1=numpy.median(data['floors']) #specifying (data['floors']) mens I only get to calculate specifics of only this part ir
print ("Median price ", median1) # Prints the median
print ("Median number of floors ", median1)  ## Prints the median
#(Mode)
#with Scipy #This function the number that appears frequently and how many times it appears.
mode1= scipy.stats.mode(data['price']) # Scipy is better than stats function as it says the number in which mode appears.
mode2=scipy.stats.mode(data['floors'])
print ("Mode", mode1) #Prints mode1
print ("Mode", mode2) #Prints mode2
#Standard deviation
print('Standard Deviation:',numpy.std(data['price'])) #For each variable it adds the square of difference of the value and ac
#For variance we also expect to know how far a data can go above/below mean
print('Standard Deviation:',numpy.std(data['floors'])) #For each variable it adds the square of difference of the value and c
#For variance we also expect to know how far a data can go above/below mean
#Range
print("Range:",max(data['price']) - min(data['price'])) # The range give the maximum- minimum number of prices of houses.
print("Range:",max(data['floors']) - min(data['floors'])) #Finds range of floors
```

```
Mean price: 540088.1417665294
Mean number of floors: 1.4943089807060566
```

## Appendix C

```
plt.figure(figsize=(17,7),) #I set the X and Y limits of figure size
plt.hist(data['price'],  align='mid', bins=30) #To plot histogram with 30 bins, and aligned at middle

plt.xlabel("Price ") #Puting the labels of x and y
plt.ylabel("Frequency")
plt.title(" Frequency against Prices per house", fontsize=20)
plt.axvline(mean1,color='green', label='Mean') #Shows the location of the mean in the histogram
#plt.axvline(median1, color='red', label='Median') ##Shows the location of the median in the histogram
#plt.legend()
plt.xticks(np.linspace(0,2500000,3)) #To create the numeric sequences, ensure histogram starts at zero to 250000
plt.show()

plt.hist(data['floors']) #To plot histogram of how frequent number of floors appear.
plt.xlabel("No of floors ") #Puting the labels of x and y
plt.ylabel("Frequency")
plt.title(" Frequency against Number of Floors per house", fontsize=20) #Plot frequency
#mean2=('1.49')
#plt.axvline(mean2,color='green', label='Mean') #Shows the location of the mean in the histogram
#plt.axvline(median1, color='red', label='Median') ##Shows the location of the median in the histogram
#plt.legend()
plt.show()
```

## Appendix D

```
from scipy.stats import sem,t #Import standard error mean
from scipy import mean #Imports mean of a population
confidence=0.95 #Set the confidence Interval


n=len(data['price']) #Calculates the number of price in my data
print(n)

m=numpy.mean(data['price']) #Calculates mean
std_err= scipy.stats.sem(data['price']) #Inbuilt function to calculate the standard error means of data

#The n-1 finds the degrees of freedom
#t.ppf is used as my data  may require some shape parameters to complete its specification.
#My data requires to appear normally distributed as it is not
#this formula tries to make it normal:(1+confidence)/2, n-1)
#We then multiply the standard error and t.ppf((1+confidence)/2, n-1)  to get upper and lower limits
h=std_err * t.ppf((1+confidence)/2, n-1)
print (h)

start=m-h #This is the upper bound interval
print (start)
end=m+h #This is the lower bound of the confidence Interval
print (end)
```

## Appendix E

```
x=data["price"][data["floors"]<2]#This line of code identifies prices of houses with less than 2 floors
y=data["price"][data["floors"]>=2] #Identify number of houses with more than 2 floors, taller building

print(len(x))
print(len(y))
```

```
12590
9023
```

## Appendix F

```
#From the initial codes, I find the mean values of tall buildings and short buildings
mean1= numpy.mean(y) #The inbuilt calculator solves (Total number of values/ number of entries)
print ("mean of the first data: ", mean1)
mean2= numpy.mean(x) #The inbuilt calculator solves (Total number of values/ number of entries)
print ("mean of the second data: ",mean2) #Mean of short buildings
print ("difference in mean: ",(mean1-mean2)) #Difference in Means is computed
Dev1=numpy.std(x) #Calculates a standard deviation
Dev2=numpy.std(y)
print ("Deviation of the first data:" ,Dev1) #Prints deviation 1
print ("Deviation of the second data:" ,Dev2)

Variance= Dev1**2 #I also print variance to calculate practical significance
print ("Variance of the first data:" ,Variance)
Variance2=Dev2**2
print ("Variance of the second data:" ,Variance2)


plt.figure(figsize=(12,7))
plt.hist((y), bins=50) #To plot histogram with 50 bins
plt.xlabel("No of floors, ") #Puting the labels of x and y
plt.ylabel("Frequency")
plt.title(" Frequency against price of taller houses", fontsize=20)
plt.axvline(mean1,color='green', label='Mean') #Shows the location of the mean in the histogram
#plt.axvline(median2, color='red', label='Median') ##Shows the location of the median in the histogram


plt.show()
plt.figure(figsize=(12,7)) #Inputs desired size of the x and y axes
plt.hist((x) ,bins=50) #To plot histogram with 50 bins
plt.xlabel("No of floors ") #Puting the labels of x and y
plt.ylabel("Frequency")
plt.title(" Frequency against Prices of Shorter houses", fontsize=20)
plt.axvline(mean2,color='green', label='Mean') #Shows the location of the mean in the histogram
#plt.axvline(median2, color='red', label='Median') ##Shows the location of the median in the histogram
#plt.legend()
plt.show()
```