> iris\_data<-iris

> table(iris\_data$Species)

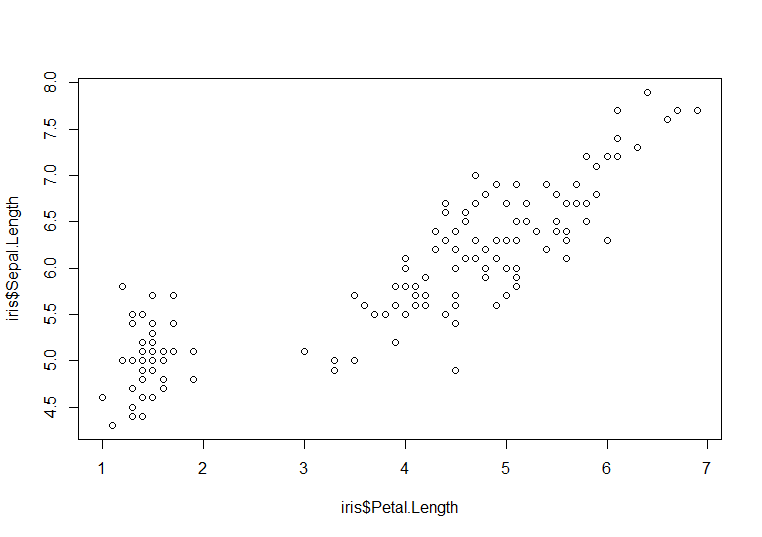
setosa versicolor virginica

50 50 50

#Scatter Plot is for Bivariate data i.e how does 1 variable change w.r.t another variable.

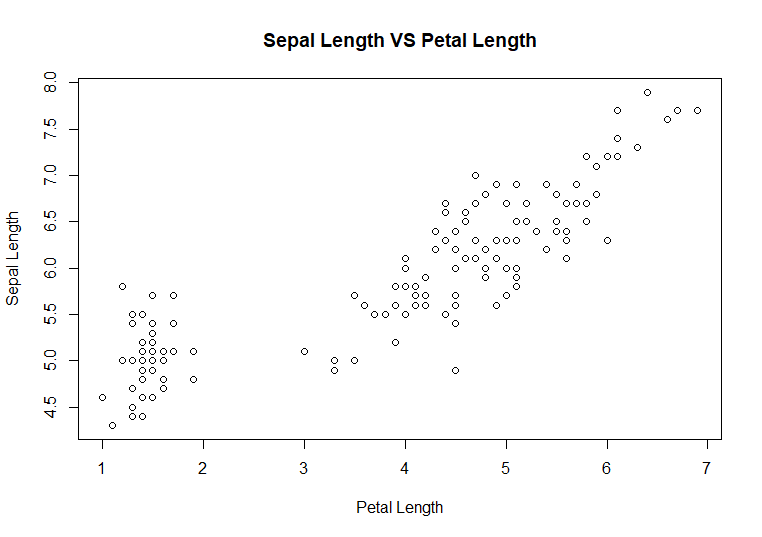
#Here we are trying to make out how does sepal length change wrt petal length# (y axis ~ X-axis).

> plot(iris$Sepal.Length~iris$Petal.Length)



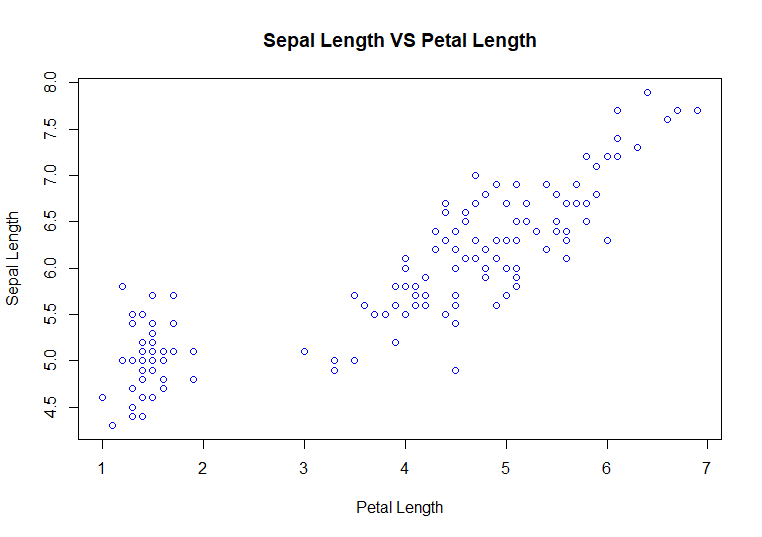
**Observation:** As the petal length increases, Sepal length Also increases. i.e there is a linear relationship between these 2 variables.

plot(iris$Sepal.Length~iris$Petal.Length,main="Sepal Length VS Petal Length",xlab="Petal Length",ylab="Sepal Length")



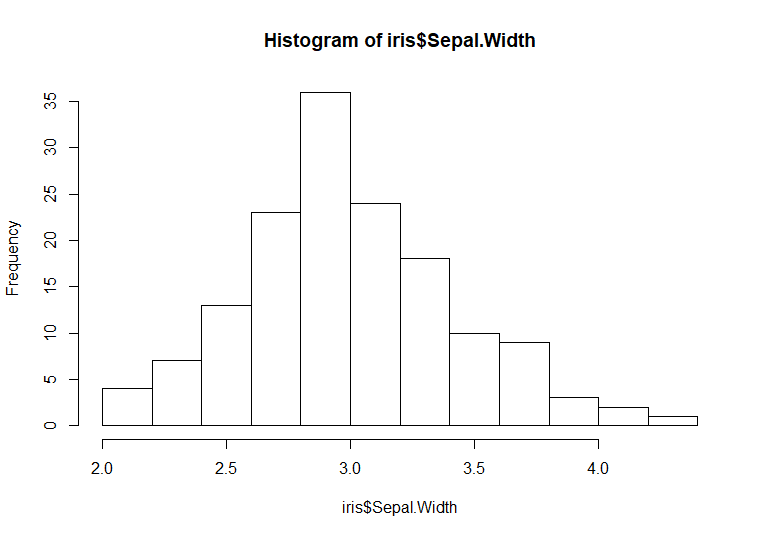
#Lets Add Color to the Plot. col= Color and pch= points Choice

> plot(iris$Sepal.Length~iris$Petal.Length,main="Sepal Length VS Petal Length",xlab="Petal Length",ylab="Sepal Length",col="blue",pch=1)



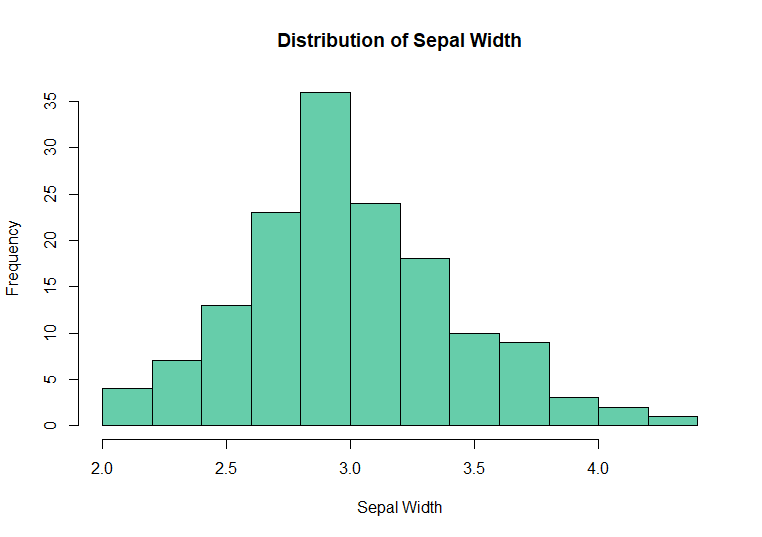
**Univariate Analysis Using Histogram**

>hist(iris$Sepal.Width)



Minimum Value of Sepal Width is 2cm to 5 cm. Most of the data is around 3cm.

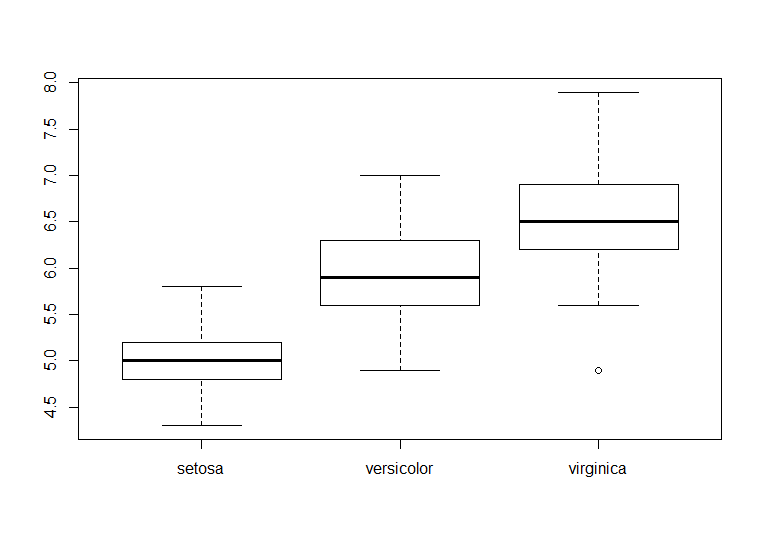
> hist(iris$Sepal.Width,main="Distribution of Sepal Width",xlab = "Sepal Width",col="aquamarine3")



**Box-Plot:**

**We can use box plot to determine how does a continuous variable change w.r.t. a categorical Variable. (SepalLength is a continuous variable and Species is a Categorical Variable).**

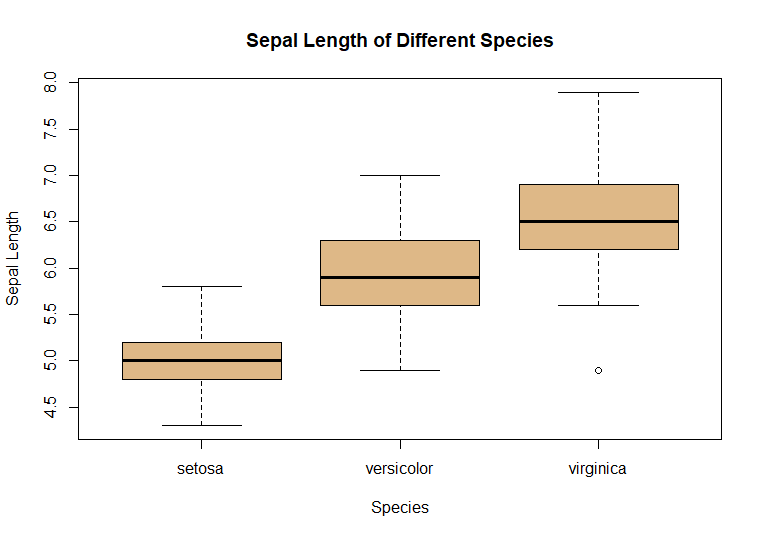
> boxplot(iris$Sepal.Length~iris$Species)

****

**Setosa has the lowest sepal length**

**Virginica has highest sepal length**

> boxplot(iris$Sepal.Length~iris$Species,col="burlywood",main="Sepal Length of Different Species",xlab="Species",ylab="Sepal Length")

****

**These are the graphs made with base graphics. There are 2 major issues with this:**

1. **A lot of print quality. i.e if we want to publish these plots for a journal that would not be possible.**
2. **These are just images, if you want to add leaves over these images again that is not possible with the help of piece graphics.**

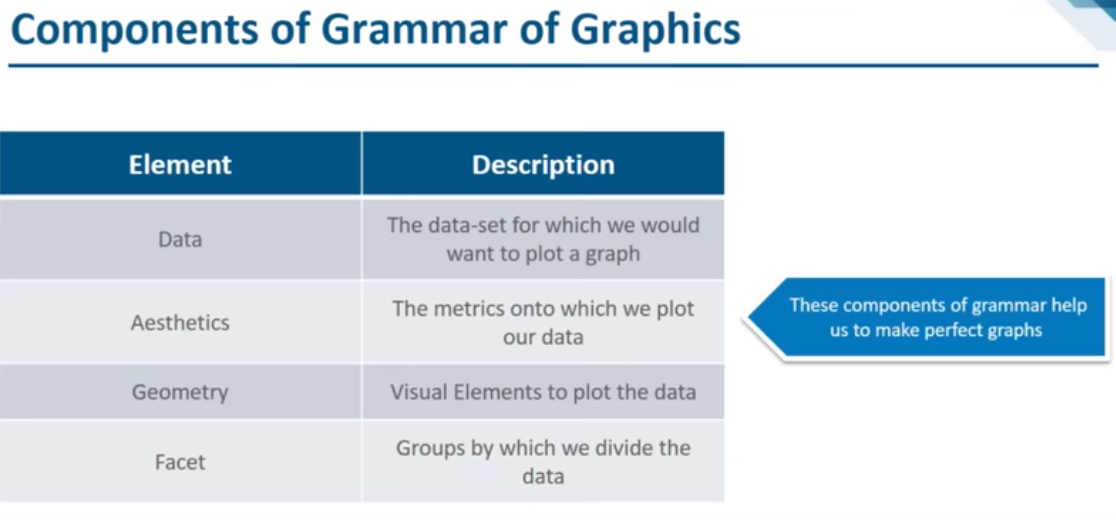
**This is where we need the help of ggplot package.**

**Ggplot 🡪 Grammar of Graphics Plot**

**2 Sentences:**

1. **I am jhon**
2. **Am john i**

**I am jhon is correct because it follows grammatical rules. Now every form of communication needs to have grammar & visualization is a form of communication, it needs to have foundation of grammar.**

****

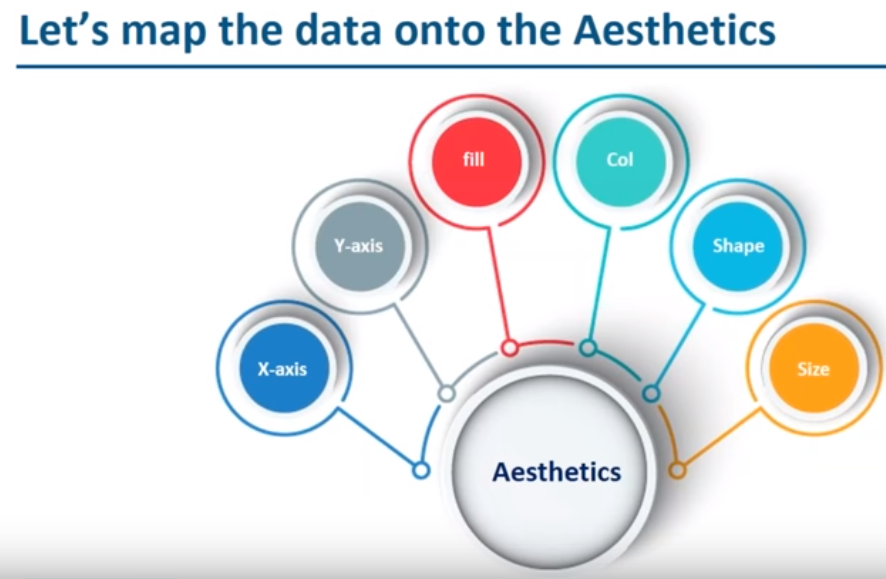
**Data:** is the data set we select.

**Aesthetics:** are the scales on to which we plot our data.

**Geometry:** Box Plot, Scatter Plot, Histogram.

**Facets:** We will be working with huge dataset and when we're working with huge data sets there would be lot of categorical variables and Lot of numerical variables and plotting all of them onto a single graph might lead to chaos that is why we need to facet or data or in simple terms we need to divide the data into groups and when we do this this gives us better visualization.

**We will select the data and we'll mark the data columns on to the aesthetics and these are the aesthetics which we have x-axis y-axis fill color shape in size so for the Easter decks part will be taking the iris dataset and we'll start off by mapping the petal length and sepal length onto the x and y aesthetic then we'll also take the species column and map this onto the color esthetic next instead of the color aesthetic will be mapping the species column on to the shape aesthetic.**

****

**Install the Package**

> install.packages("ggplot2")

Bring the package into R.

> library(ggplot2)

**1. Select the data**

> ggplot(data = iris)

**You will get a blank Plot**

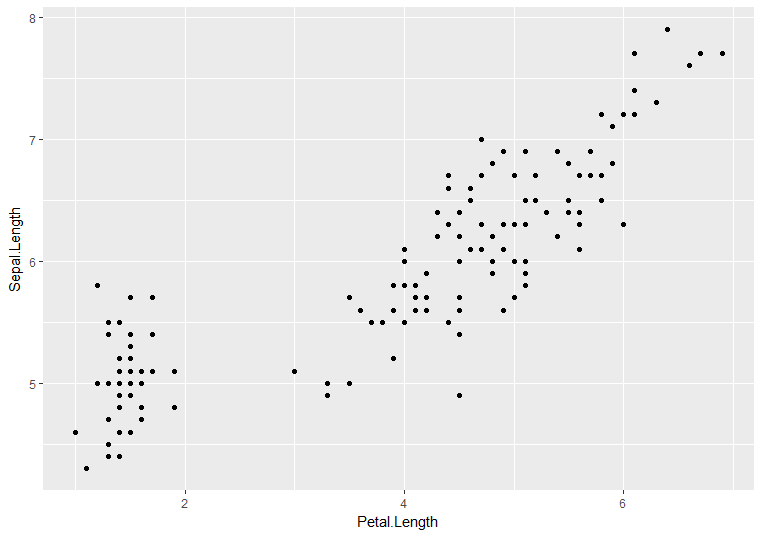
**2. Map the data columns into aesthetics.**

> ggplot(data=iris,aes(y=Sepal.Length,x=Petal.Length))

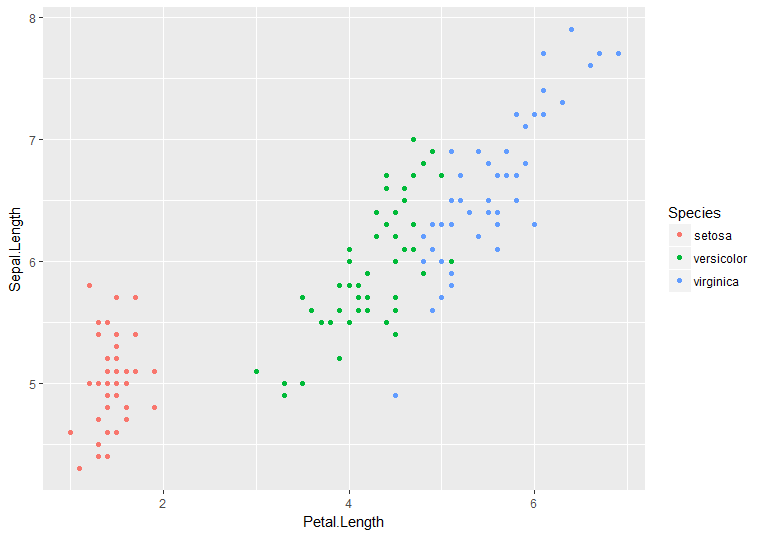
**Map the sepal length onto y aesthetic and petal length on to x aesthetic**

**3. Geometry: Geometry is geom\_point here. This will give us a scatter plot.**

> ggplot(data=iris,aes(y=Sepal.Length,x=Petal.Length))+geom\_point()

****

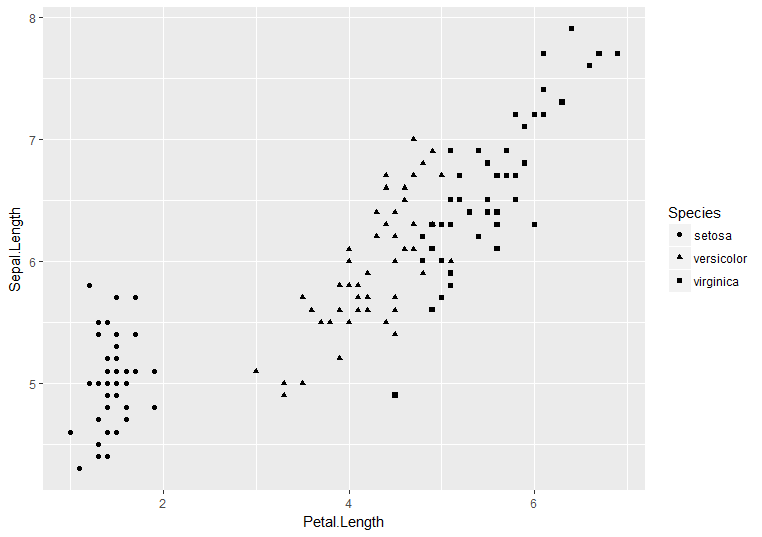
> ggplot(data=iris,aes(y=Sepal.Length,x=Petal.Length,col=Species))+geom\_point()

****

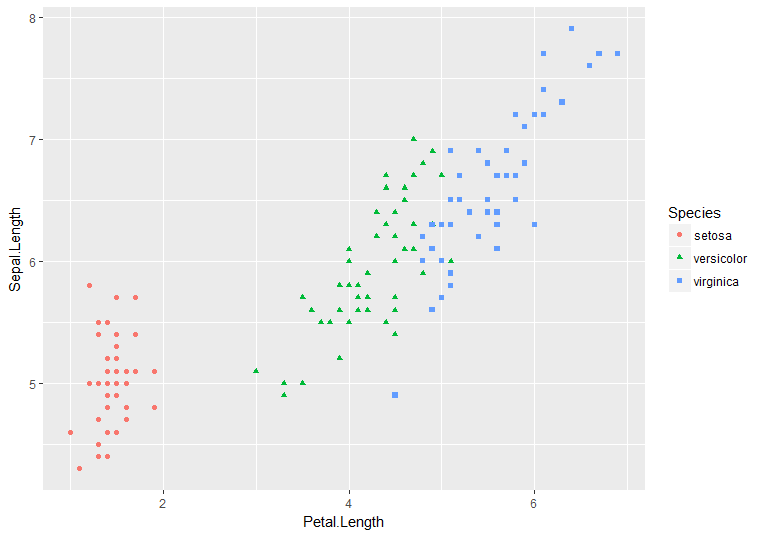
**Observation: Setosa: Lower Sepal and Petal Length**

**Virginica: Higher Petal and Sepal Length**

> ggplot(data=iris,aes(y=Sepal.Length,x=Petal.Length,shape=Species))+geom\_point()

****

> ggplot(data=iris,aes(y=Sepal.Length,x=Petal.Length,col=Species,shape=Species))+geom\_point()

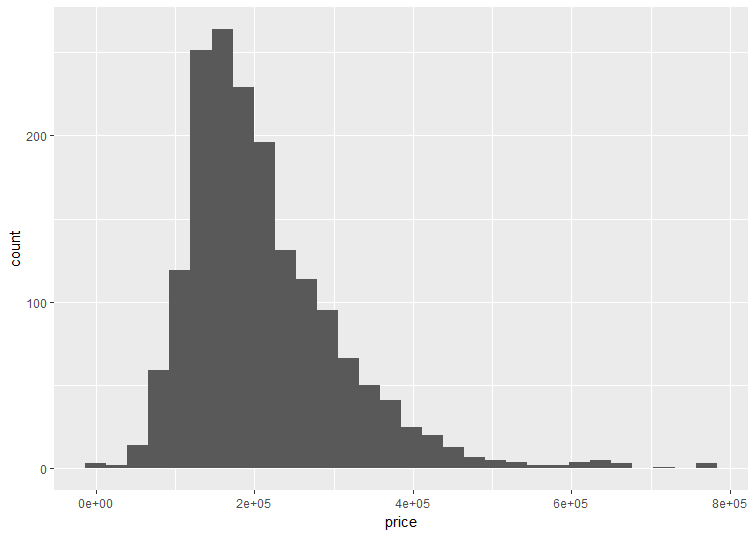
****

**#Remove the unwanted data from the dataset**

**house=house[c(-1,-2)]**

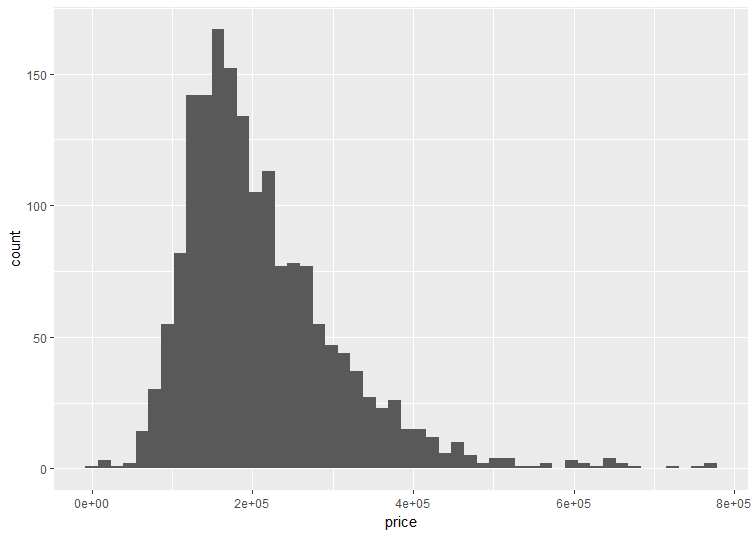
**Map “price” to x aesthetic.**

**ggplot(data=house,aes(x=price))+geom\_histogram()**

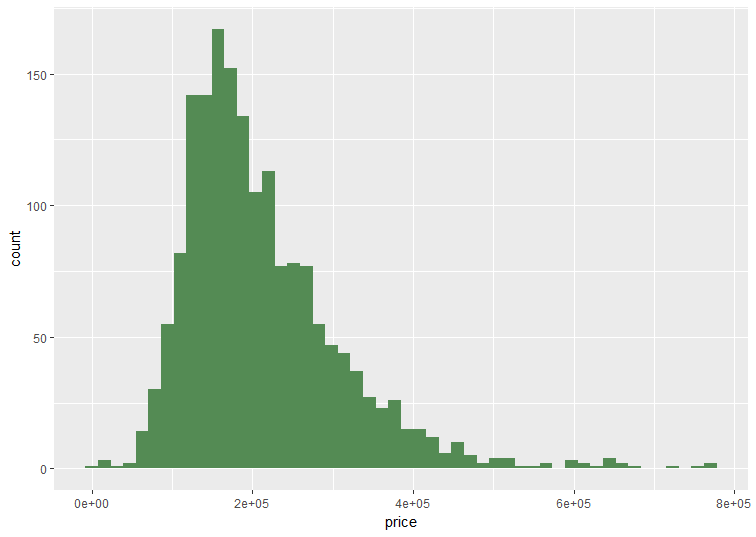
****

* Most of the houses, their price range is around 2lakhs to 4lakhs and max price of the house is around 8lakhs.
* Note: 30 is the default value for number of bins.

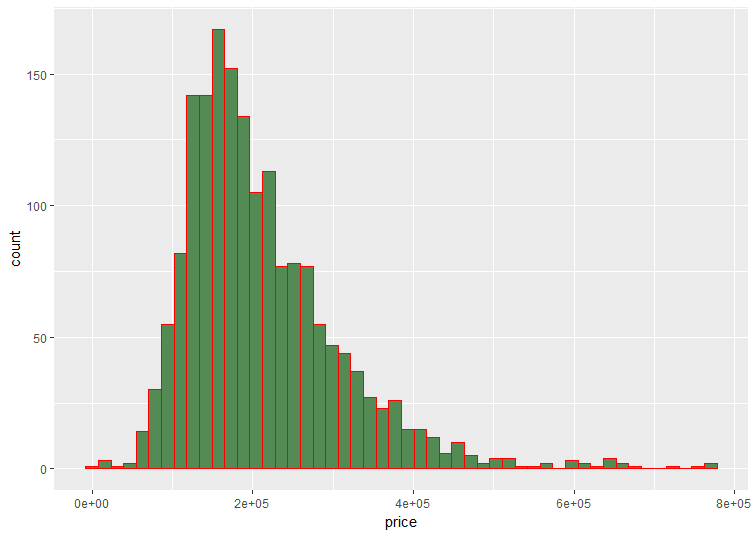
**ggplot(data=house,aes(x=price))+geom\_histogram(bins=50)**

****

**ggplot(data=house,aes(x=price))+geom\_histogram(bins=50,fill="palegreen4")**

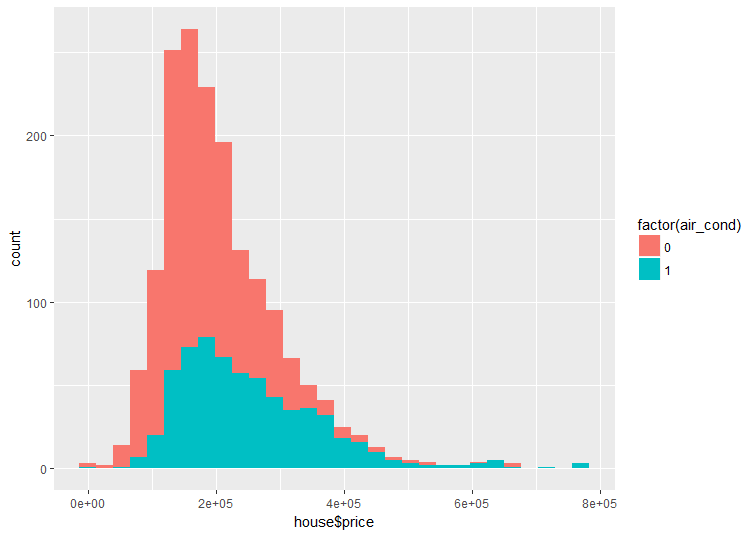
****

**ggplot(data=house,aes(x=price))+geom\_histogram(bins=50,fill="palegreen4",col="red")**

****

**These graphs actually don’t look really pretty. So lets use fill as aesthetic and not as the attribute.**

**ggplot(house,aes(x=house$price,fill=factor(house$air\_cond)))+geom\_histogram()**

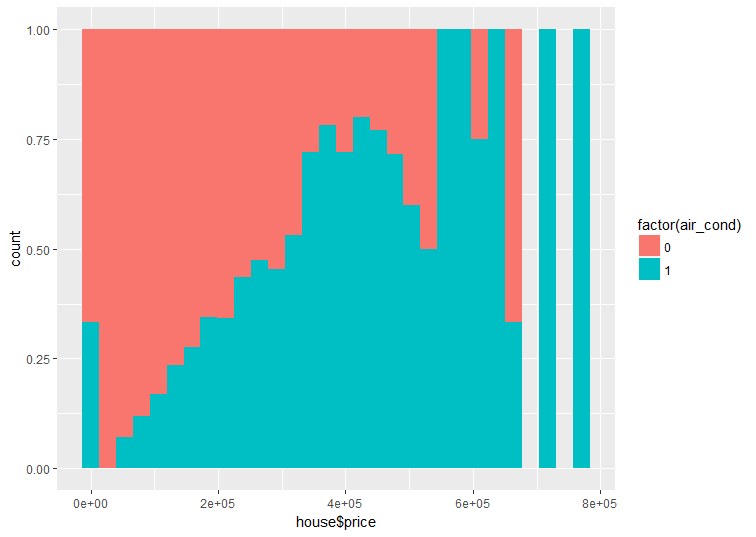
****

If the house has **centralized air conditioning** then the color of the bin would be **blue** and if the house does not have centralized air conditioning then the color of the bin will be Orange.

So what we can conclude from this is most of the houses do not have air conditioning so if you see price range for blue, around 2 lakhs to 3.75 lakhs and there is a greater proportion of houses which do not have centralized air conditioning, so as the price increases from 3.75 lakhs to 8 lakhs, the probability of the house having air-conditioning increases.

Let's change the position

ggplot(house,aes(x=house$price,fill=factor(air\_cond)))+geom\_histogram(position = "fill")



This give us the count of the bins, this will give us the proportion, so I am using **position=fill.**

When I say position=fill, this graph will give me the proportion instead of count.

What we can see is as the price increases from $50000 to $400000 there is a greater possibility of the house having air conditioning.

The houses which are in the price range of 4 lakhs, there is a greater likelihood for them to have air conditioning, as the price of the house increases from 4 lakhs to 8 lakhs the probability of the house having air-conditioning increases even more.

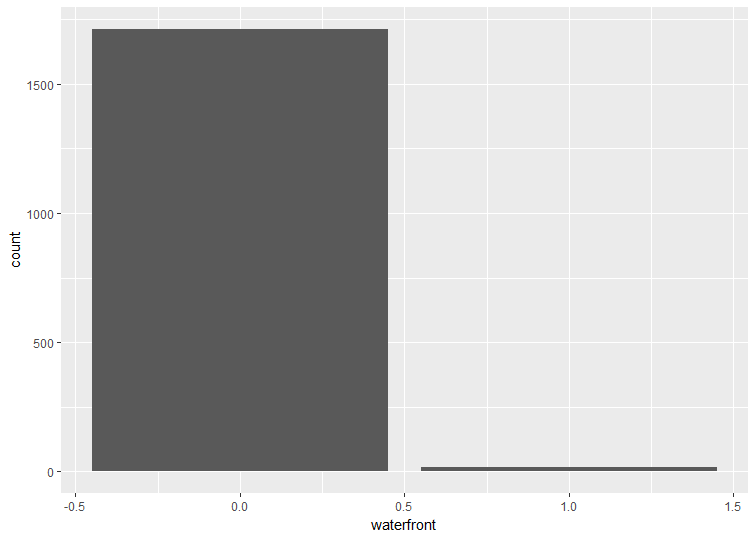
From the two bins you can see that, if the price of the house is closer to 8 lakhs then the house will definitely have centralized air conditioning.

**Bar plots**

Select data, map data column on to the aesthetic i.e we have mapped waterfront which is a categorical variable and we've mapped this onto the X aesthetic.

We have selected geometry which is a geom\_bar() that'll give us a bar plot.

> ggplot(data=house,aes(x=waterfront))+geom\_bar()

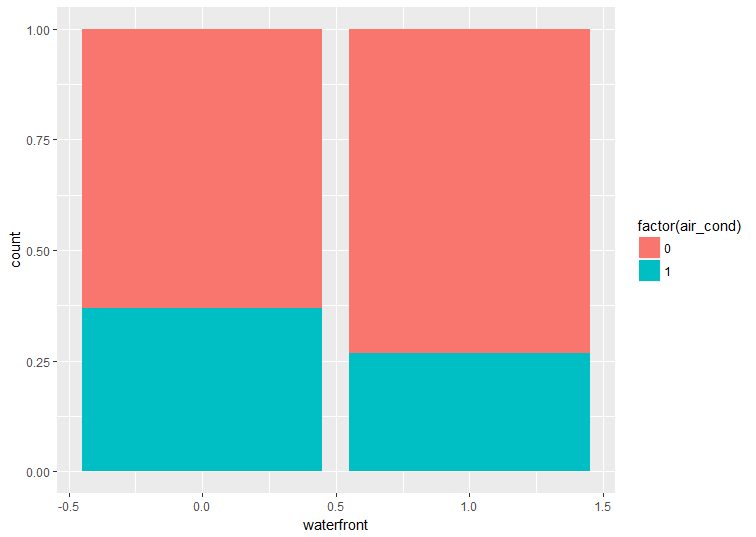


We see that this gives us the distribution of whether the house has waterfront or not.

Let's look at the data set, this tells us that there are around 1728 entries or in other words there are 1728 houses and out of the 1728 houses we can see that around 1600 houses will not have a waterfront, so this bar tells us that the houses do not have a waterfront and bar which tells us that the houses have a waterfront.

So there are very few houses which have a waterfront.

* ggplot(data=house,aes(x=waterfront,fill=factor(air\_cond)))+geom\_bar(position = "fill")

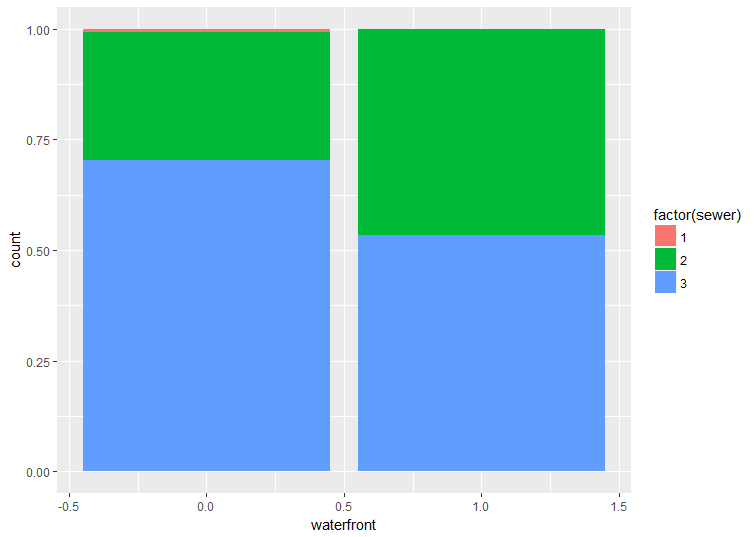


We have mapped the air conditioning column onto the fill aesthetic so the color is determined by whether the house has air conditioning or not and we've given the position to be fill, this gives us the proportion so these two.

If the house does not have a waterfront then around 30% of the houses will have centralized air conditioning and if the house has a waterfront then around 25% of the houses will have centralized air conditioning.

Another data column to the fill esthetic, now we are mapping the sewer data column or the type of sewer system onto the fill esthetic

>ggplot(data=house,aes(x=waterfront,fill=factor(sewer)))+geom\_bar(position = "fill")



If you look at this, we see that if the house does not have a waterfront there is a tiny bit of possibility that the house might not have a sewer system as well and if the house has a waterfront then the house will definitely have a sewer system whether it is private or public.

Another inference is if the house has a waterfront then there is greater possibility for it to have a private sewer system, so if the house does not have a waterfront then there is a greater probability that it will have a public sewer system.

**TIME SERIES ANALYSIS**

1. **Why do we need time series analysis**?

There are quite a few algorithms which can predict values, then why do we need one more algorithm called the **time series algorithm** to predict value.

The other algorithms such as the linear regression or logistic regression there were two variables, one was a dependent variable and one was the independent variable, you used to predict the dependent variable based on the values of the independent variable.

For example a particular value of x you used to see what will be the value of y. In time series analysis you will be given only one variable and you have to predict that variable.

Now imagine if you just have one variable, how will you predict it? and the only thing which will be given in the data set is in accordance with the time.

For example, if I owned a car company and I say- last month I sold around 23 cars, this month I sold 24 cars, what is the sales that I will achieve in the next month?

In this case the only variable that I have is the sales part. I don't know what is the other variable that I have to depend on, the only thing that is there is time and I have to predict for the next month.

We use time series analysis when you have only one variable and you have the time as a second variable that you will be predicting the value for.

1. **what is time series analysis?**

Time series is a series of data points indexed in a time order. It's a series in which

you have indexed the points according to a time order.

You use a time series algorithm to create a model, once you have created that model you use that model to predict future values.

1. **When do we not use time series analysis?**

**We don't use time series analysis whenever the values are constant.**

For example, my companies sold 23 cars in the past month, 23 cars in this month, what is the sales that is going to be in the next month. In this case my number of cars is same for the previous month and this month as well so there is no need of any analysis.

**You cannot use time series analysis if your value can be represented using a function.**

For example the **sin(x) function**, if you have the x value you can get that value by putting it in the function, now you can apply time series analysis on this as well but there is no point because you could have got that value by putting it in the function itself

1. **Cases in which you cannot use time series analysis**

You cannot do time series analysis if the data is not stationary. If your data is not stationary or your data series is not stationary you cannot apply time series analysis on that.

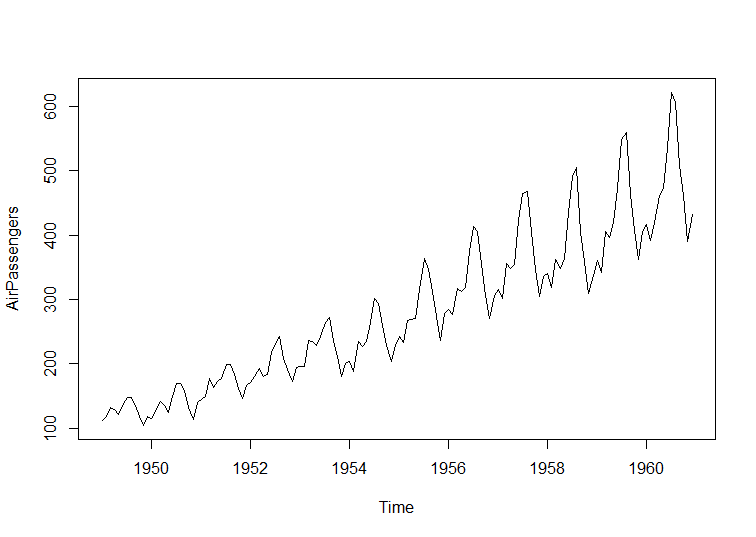
What is stationary: For the series to be stationary there are 3 conditions:

1. The mean should be constant according to the time.
2. The variance should be equal at different time intervals from the mean.
3. Covariance should be also equal.

If these three conditions are met then only we can say series is stationary and then we can apply the time series analysis.

We will be doing our time series analysis on **air passengers**.

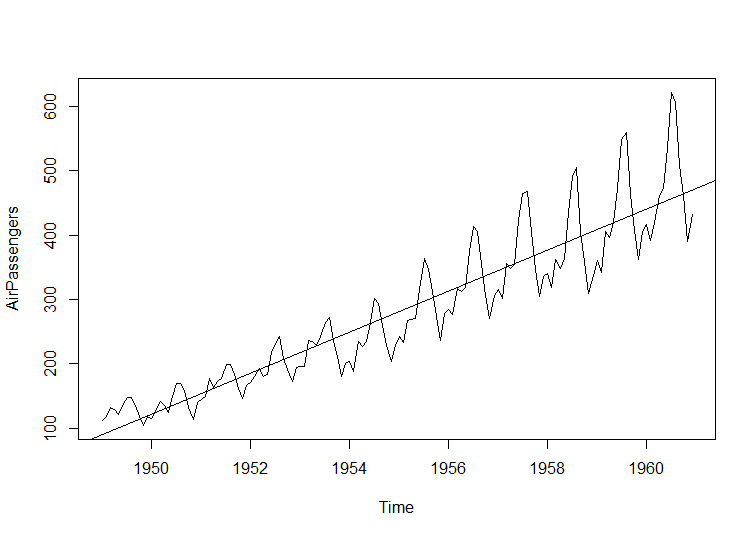
> plot(AirPassengers)



We have data from 1949 to 1964 each and every month and we have to predict for the next 10 years what will be the sales.

**mean** is basically the average of the points as you'll be calculating the average of these two points and you'll pass a line through it.

* abline(lm(AirPassengers~time(AirPassengers)))



1. The mean is changing according to the time frame, For example the mean increased from 1950 to 1952, So this means value should not be increasing but should be constant.
2. Variance is like I said the distance of each point should be same from the means But that is not the case.

We have to transform the data so that our variance becomes equal.

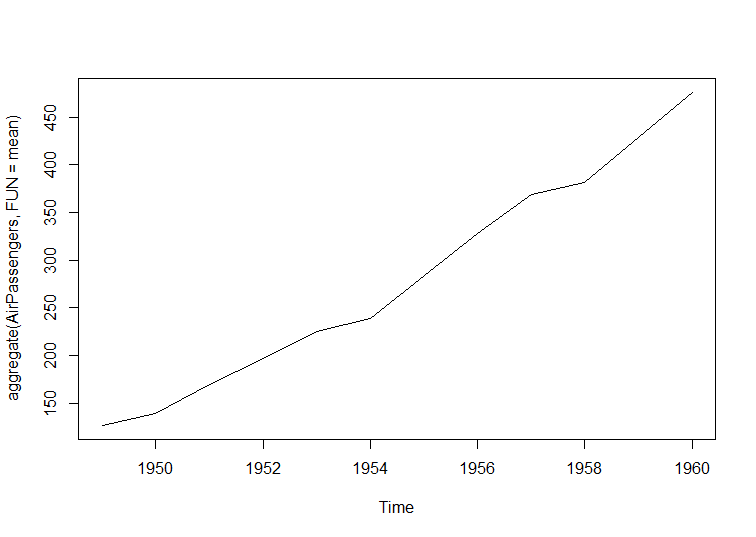
If my variance, mean and covariance all become equal, then data series becomes stationary and hence can apply time series analysis on that data.

**Components of time series:**

There are basically three components:

1. **General trend:** A general trend is basically how are your values going, are they increasing or decreasing.

* plot(aggregate(AirPassengers,FUN = mean))

****

Values are increasing every year and that is how the trend is going. Trend can be either upwards or it can point downwards and it's a component of time series.

1. **Seasonal: A peak or dip seen in a time interval.**

A seasonal component can be understood with an example. You have an airline company and would have noticed that at every festival there is going to be a peak in the number of commuters that are going to take airline, for example during Christmas a lot of people will be going home and hence there will be a spike in number of people who'll be traveling through airline and hence you would have also noticed that whenever you try to book your tickets during peak times i.e during Diwali or during Christmas that the fares are quite high and they are high because the airlines have already speculated that at this particular time number of passengers would be more and it doesn't matter how much cost they are asking you and hence that is the time that the airline makes the maximum profit.

1. **Irregular fluctuations: The uncontrolled situations which arise due to which the value changes.**

Irregular component is basically something like say in November you have fog and hence a lot of flights were canceled, and the number of commuters hence goes down. But it is not the same case next year fog is not going to be there. It is not something that is repeating every year, something that there are uncontrolled circumstances is called the irregular fluctuations that you have in your time series analysis.

**Coding**

1. **Import the data set**
2. **What class the dataset belong to**

class(AirPassengers)

1. **To check the start and end of the time series**

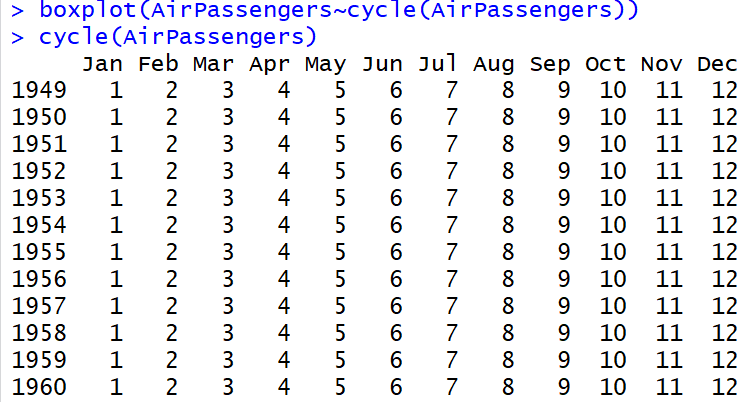
**start(**AirPassengers)

**end(**AirPassengers)

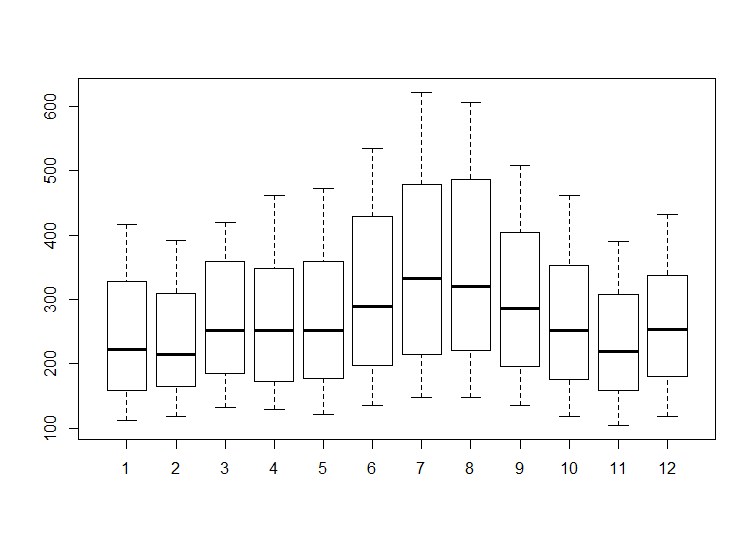
1. **Frequency will tell the intervals**

**frequency**(AirPassengers)





**Note: Cycle will print the cycle across years.**



In Boxplot all you can analyze, which month maximum passengers were travelling, for my data of 11 years I can see that every year the number of people who were going more or people who are traveling more is in the month of August,7 means August, so in the month of August and in the month of July, with boxplot we can check seasonality part.

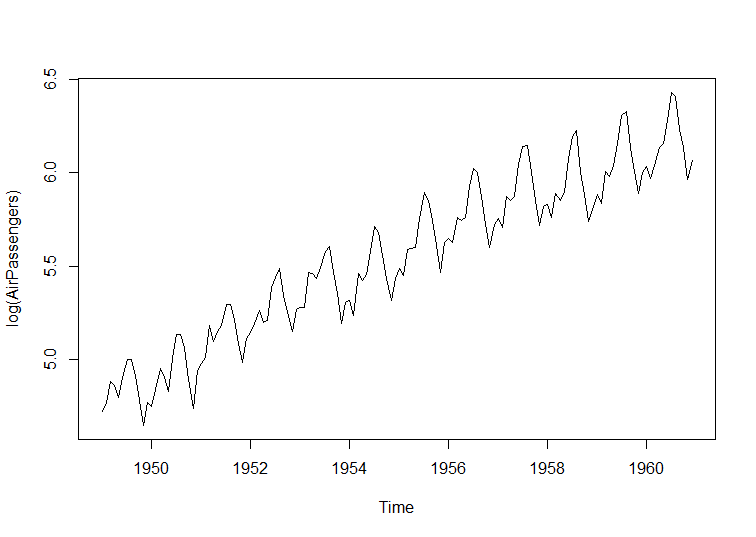
Can check the trend of data.

* **Plot(AirPassenger)**

Convert this particular series to stationary. mean is also increasing the variance is also not equal, so lets make data stationary.

For dealing with variance, we'll be applying the log function. Apply the log function it'll make the variance equal.

> plot(log(AirPassengers))



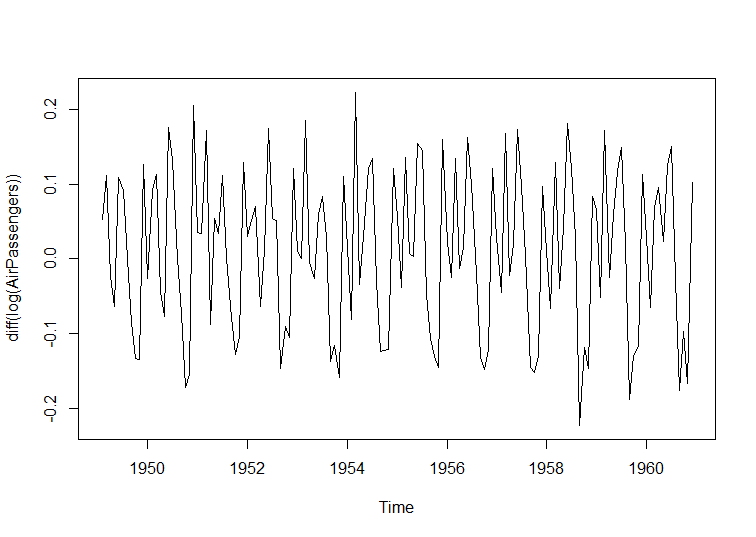
If I pass a mean line the variance is going to be equal.

If i pass a mean line each point is going to have the same distance at each interval of time, this can be done using the log function.

Hurdle that I have right now is that the mean is still changing according to time. I want the mean to stay constant according to time so that my series becomes stationary.

The way to do that is differentiating it.

> plot(diff(log(AirPassengers)))



The mean is now constant according to time. At the same time variance is also going to be the same.

Series is now stationary and I can apply the model on it.

The time series analysis model you can apply to do the time series analysis you have the **ARIMA** model you have the **whole printers** model etc.

**ARIMA** is basically an acronym AR stands for AutoRegressive, **ma** stands for **moving average** and **I** stands for **integration**.

Auto Regression is when you are seeing the past values to predict your own value that is called Auto regression.

Moving average, there is a formula for moving average using which you take different intervals and you calculate the average.

**I** is integration.

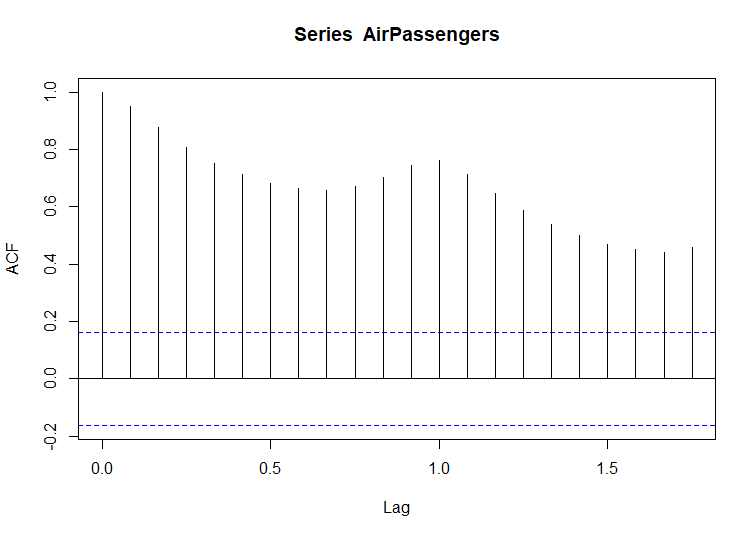
With these three we have values We have some values associated with **AR** called **p** with **I** have the value called **D** and with **MA** I have the value called **Q**.

Whenever I am applying the ARIMA model I have to go ahead and include these three values as well we have to get from the graph called the autocorrelation function graph.

Using graph we will determine the value of **P,** value of **Q** and the value of **D**.

The autocorrelation function when we are not applying anything to our series, when you have not transformed a series to stationary.

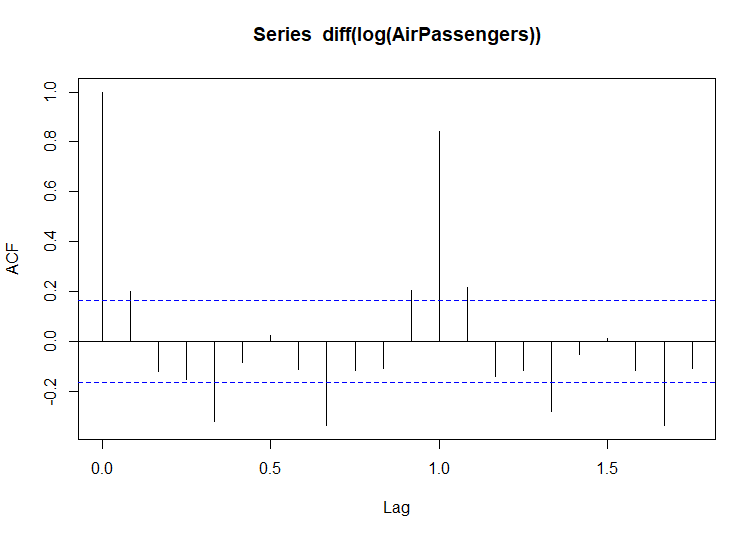
> acf(AirPassengers)



In this graph you can see the values are exceeding the blue line. Our aim is to keep it under the blue line and invert.

If i apply the differentiation and the logarithmic function to data series and call the acf() we will get a graph. With ACF I can calculate the value of **Q**

> acf(diff(log(AirPassengers)))

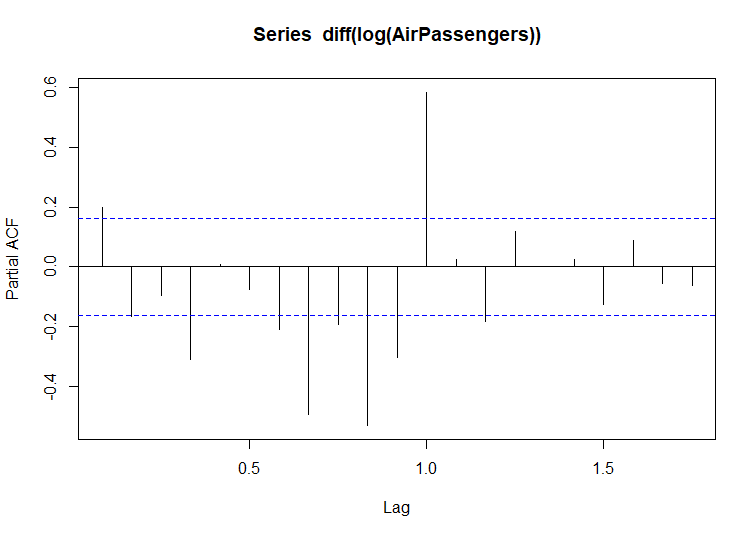


Q is the coefficient for MA. I see the first line which gets inverted. For example 0th line, 1st line and 2nd line is the line which gets inverted, just before that the line which is there we have to take that value.

We will be taking the 1st line as value of Q. **S**o the value of Q becomes 1.

Same method will be applied in the partial autocorrelation function graph.

> pacf(diff(log(AirPassengers)))



From the graph I will get the p-value, after 0th line, 1st line is inverted hence my value of P becomes 0.

So value of P is 0 & value of Q is 1 and value of D is basically the differentiation that you do in your function, in time series algorithm I did differentiation once and got the mean constant to my time line.

You can do differentiation multiple number of times, as I have done differentiation only once and hence value of D = 1.

If you do differentiation twice value of B will be 2, you do that until and unless your mean becomes equal or your mean becomes constant.

P=0, D=1 and Q=1. I'll be using that in ARIMA model.

When you pass function, you will be specifying the PDQ values, these parameters change according to your data set.

**Create the model**

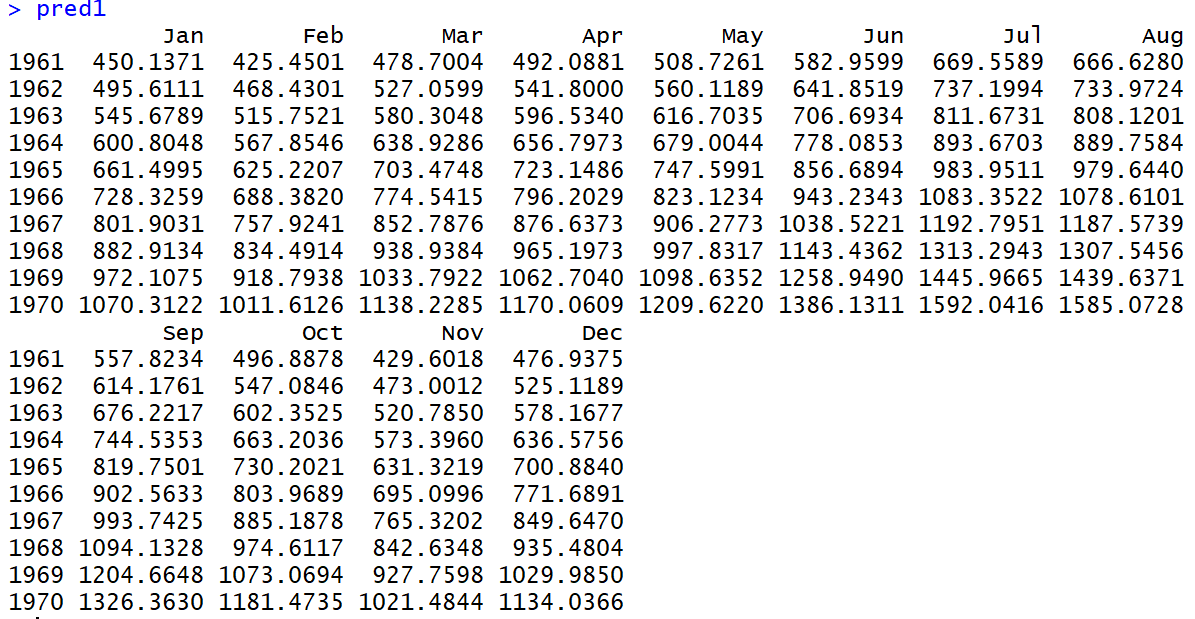
fit<-arima(log(AirPassengers),c(0,1,1),seasonal = list(order=c(0,1,1),period=12))

Now I will be predicting values for the next 10 years.

> pred<-predict(fit,n.ahead=10\*12)

Once I have predicted that the values are in logarithmic form. To convert them into the decimal form you have to use the e value. e value is 2.718, to see these values

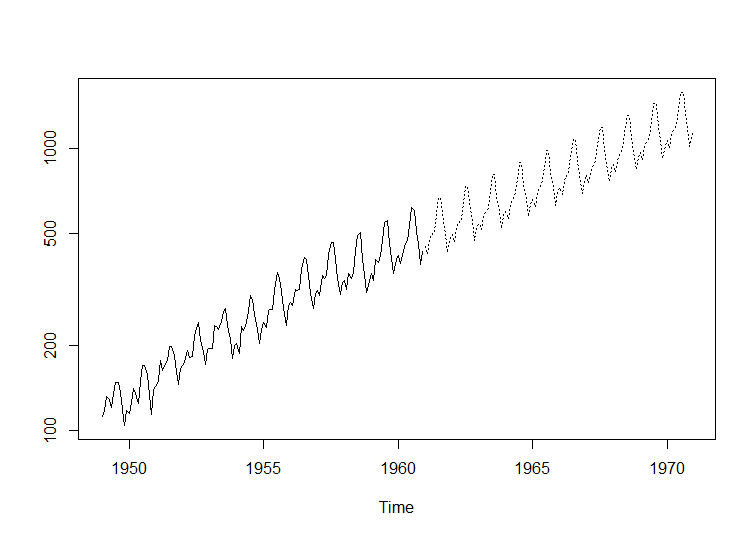
> pred1<-2.718^pred$pred



I had data set till 1960 and analysis has now given the values till 1970, it says in the year 1970 in September, we will have these 1326 passengers who will be traveling by air. This is what my model predicted.

**Plot the graph:**

> ts.plot(AirPassengers,2.718^pred$pred,log="y",lty=c(1,3))

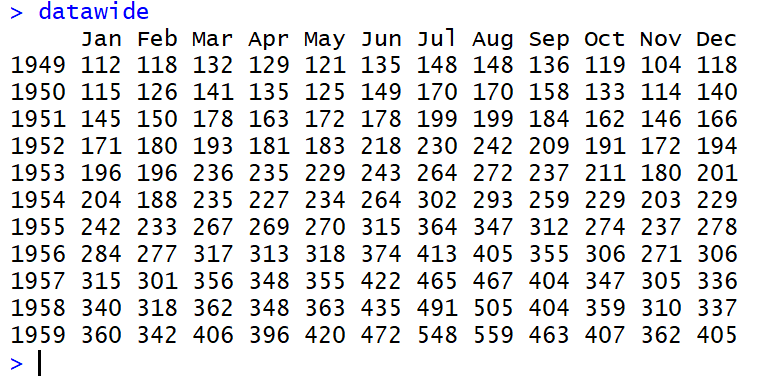
****

These dotted lines is what we have predicted for the coming years.

This model have taken care of this seasonality factor, irregularity factor and trend, all these three factors combined you have predicted the values for the future.

**Test this model**: We are going to take a data set till 1959 and then we are going to predict the value of 1960 and then validate that value of 1960 from the already existing value that we have in our dataset.

> datawide<- ts(AirPassengers,frequency = 12,start=c(1949,1),end=c(1959,12))

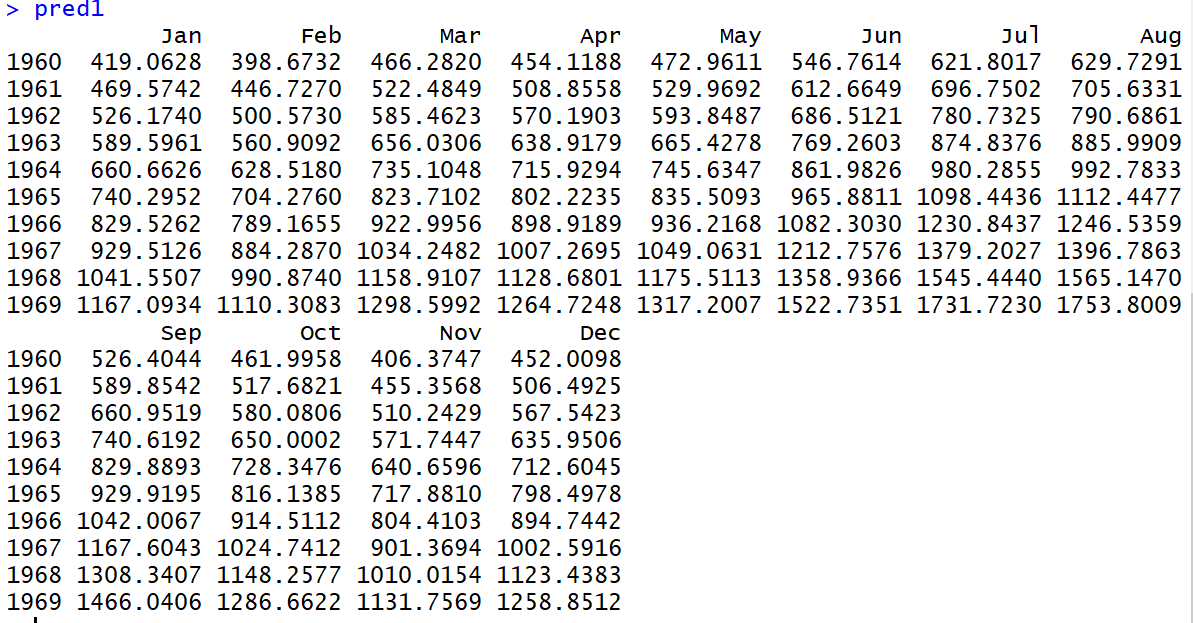


Now create the model out of this particular data set:

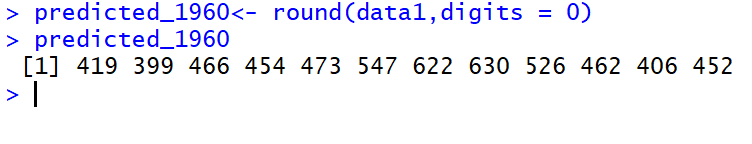
> fit<-arima(log(datawide),c(0,1,1),seasonal = list(order=c(0,1,1),period=12))

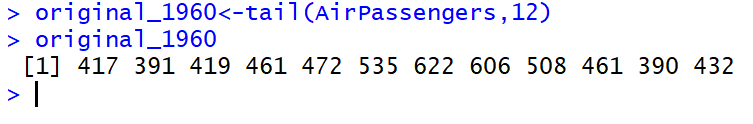
> pred<-predict(fit,n.ahead = 10\*12)

> pred1<-2.718^pred$pred



> data1=head(pred1,12)





If I compare original value in 1960 January 417 people flew and according to my model 419 people, but it's pretty close.

**Real Time Application of Time Series**

Say I have a call center company or an e-commerce company and my customers calls me on a daily basis and I have to adjust the shift of the people. Say there are around 100 people who can handle the maximum traffic. Say for example at 4:00 p.m. I get around 20 customer calls but at 8:00 p.m. I get around 60 or 70 customer calls, I can predict that for tomorrow and can arrange the shift of people according to that. I have to adjust the shift so as to match the needs of the customers and I can predict how many customers will be calling me by time series analysis.

