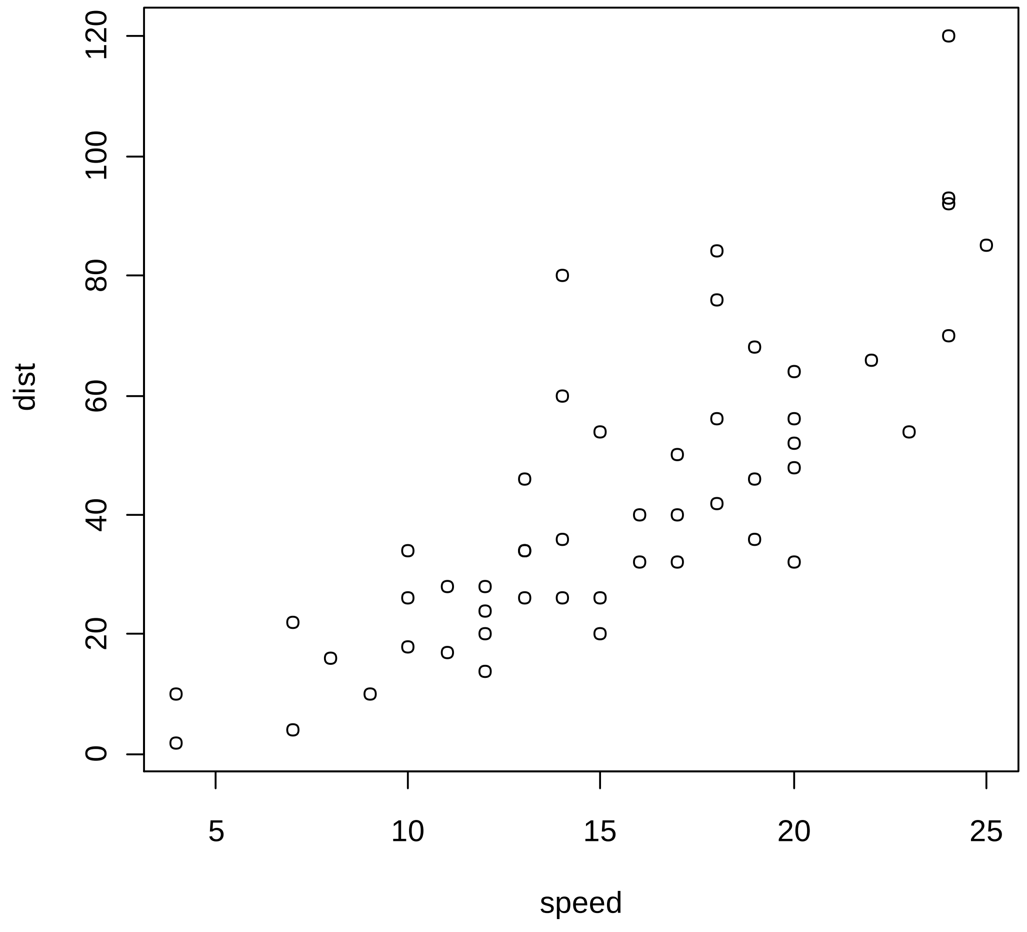
Plotting Scatter Plots

A scatter plot is a good tool to use when you have continuous bivariate data. Generally, the independent variable is plotted on the x-axis and the depended variable on the y. Scatters plots are useful tools for showing correlations between data. Plotting a scatter plot is extremely easy in R. Simply use the plot(x,y). For example using the data set "cars":

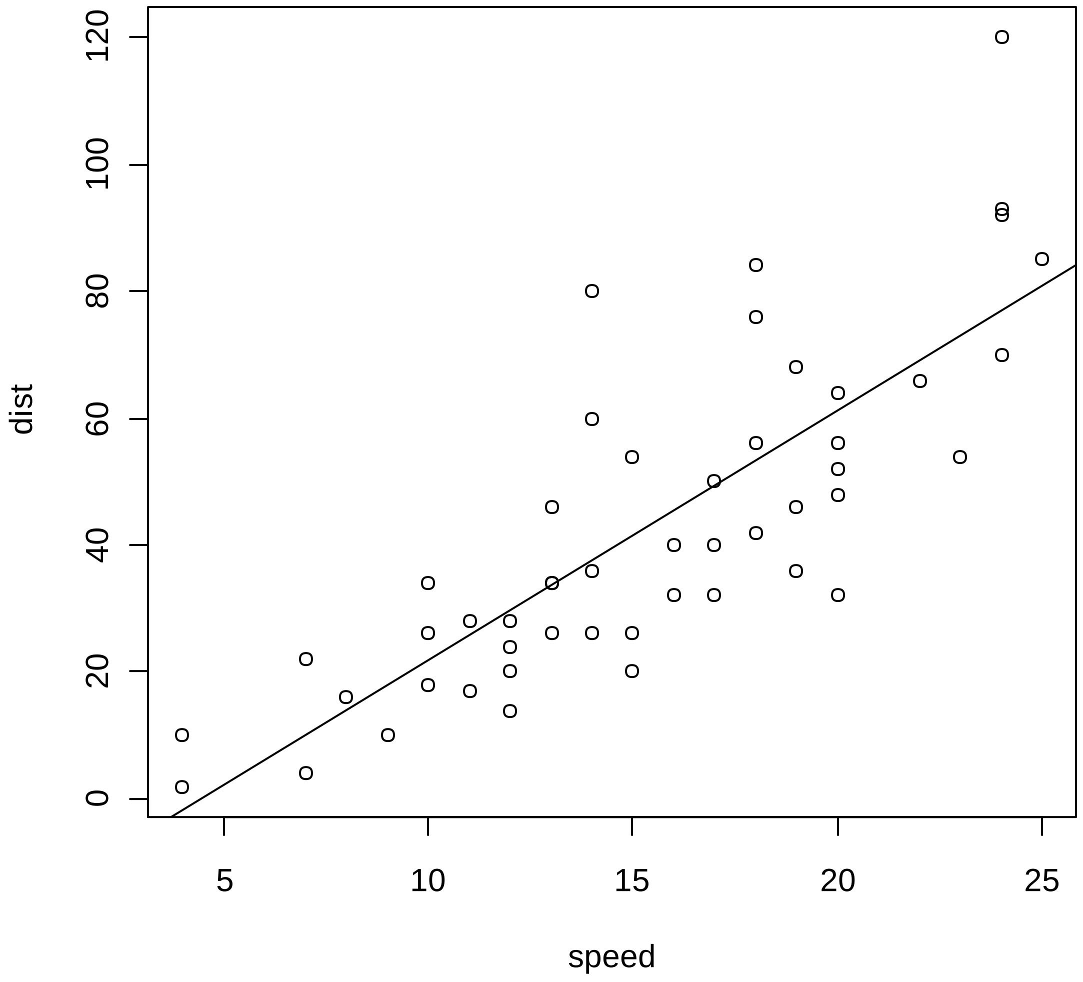
- <plot(speed,dist)

In the data set "cars" there are two variables speed and distance. Speed is the independent variable so it goes on the x axis and distance is the dependent variable so it goes on the y axis.



Adding a Regression Line to a Scatter Plot

- Adding a regression line to your scatter plot is very simple. A regression line can be used to quantify the strength of the relationship between y and x. The abline(lm(y~x)) function is used. For example
    - > abline(lm(dist~speed))
- The resulting graph looks like this:

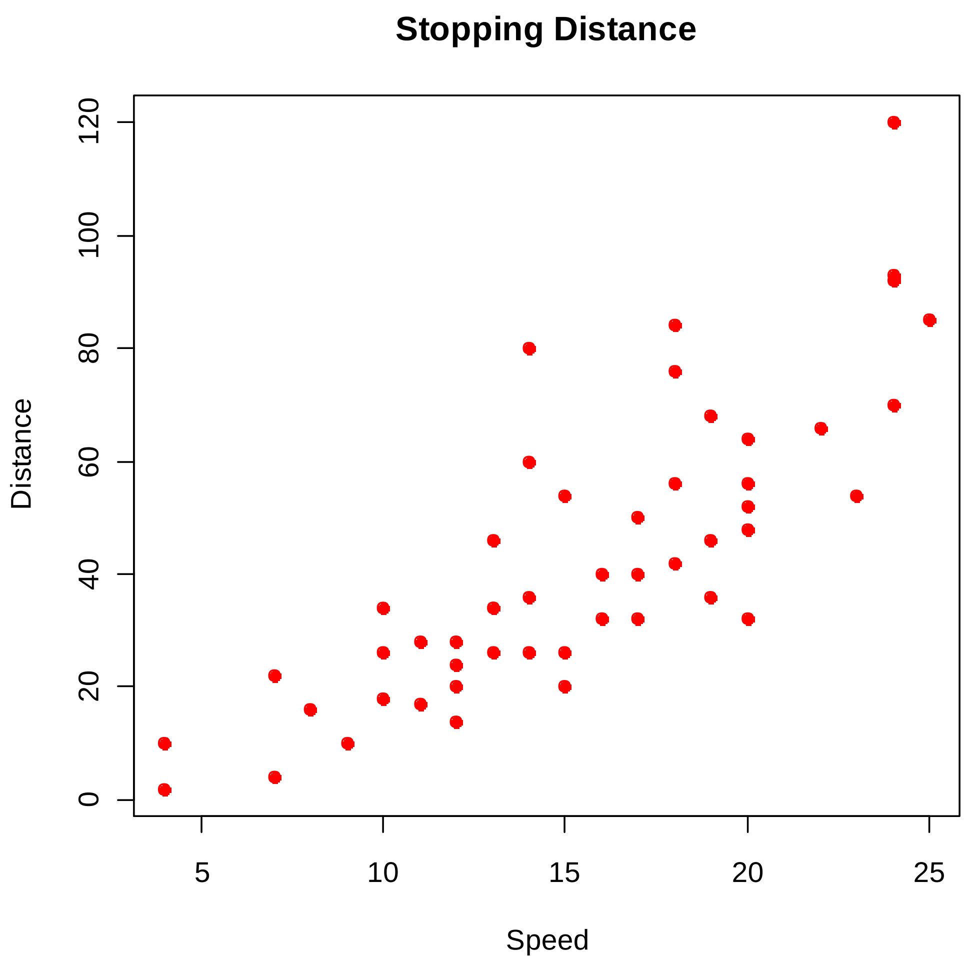


Making a Scatter Plot Look Good

- Labeling and naming plots is extremely important and very easy.
  - To add a title to the scatter plot the command is main=”x”
  - To label the axis simply use xlab=”x” and ylab=”x”
- Coloring the points is easy as well
  - To color points the command pch=21 is essential. Now set the background to a color of your choosing bg=”red” and the outline color, col=”red”
- For example with the data set cars:

```
> plot(speed,dist,main="Stopping Distance",xlab="Speed",ylab="Distance",pch=21,bg="red",col="red")
```

- This outputs the graph below and now it looks much better

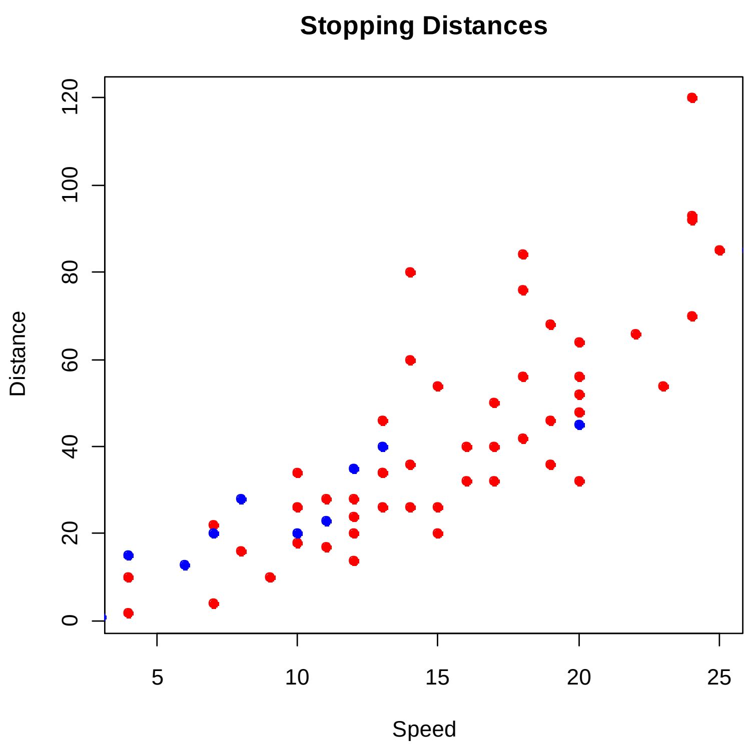


Adding Points to a Scatter Plot

Adding points to a scatter plot can be useful and it is extremely easy. To add points to a graph simply making use of the points() function.

1. Import the new data set into R.
    a. > read.table("cars2.txt",header=TRUE)
2. Attach the data set
    a. > cars2=read.table("cars2.txt",header=TRUE)
    b. > attach(cars2)
3. Add points to graph
    a. > points(Speed,Distance,col="blue", pch=21,bg="blue")

- Note that in the points() function you need to tell R where to get the points from. Speed and Distance are the variable names of the new data set cars2. If you are not sure what the variables are named, use the command the following command and it will output the variable names
    - >names(cars2)
    - [1] "Speed"   "Distance"
- Note how I colored the new data points blue so we can differentiate between cars and cars2



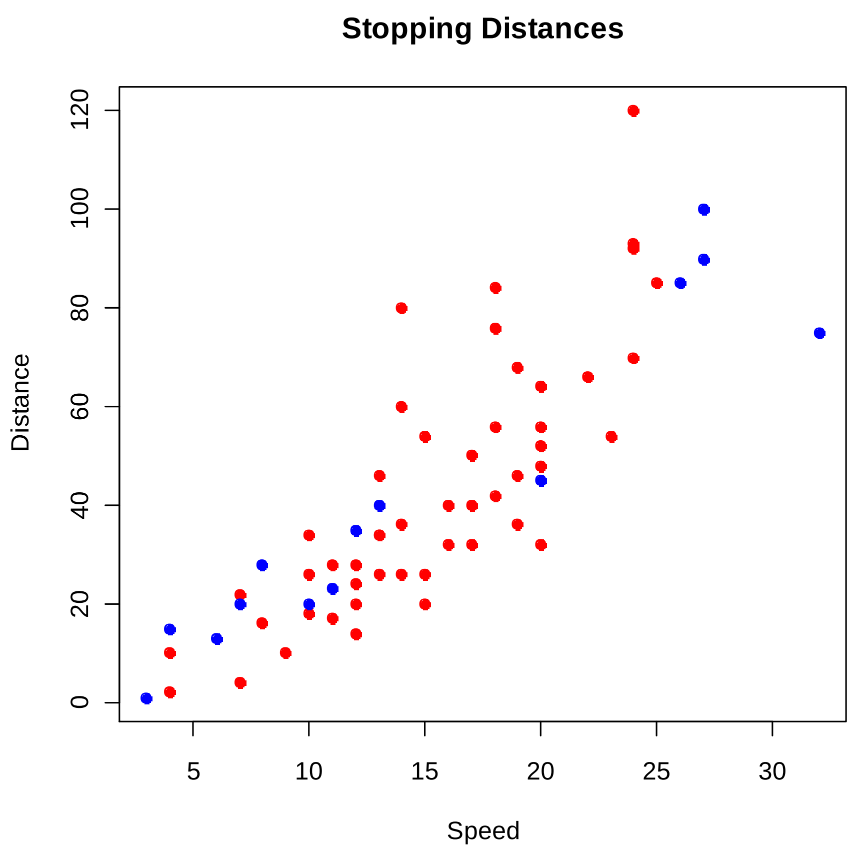
Rescaling Your Scatter Plot/Adding Two Data Sets to a Scatter Plot

We have successfully added data points to our scatter plot. However, all of the data points in cars2 are not showing on the graph. This is because the graph was scaled to fit cars, not cars2. It is simple to cure this problem. You can plot the points from both data sets on the same plot using type=”n” so that the axes are scaled such that all the points are encompassed on the graph (using the concatenation function). For example:

- >plot(c(speed,Speed),c(dist,Distance),xlab="Speed",ylab="Distance",main="Stopping Distances",type="n")

- The two variables speed(cars) and Speed(cars2) are the variables that we want on the x axis and the two variables dist(cars) and Distance(cars2) are the variables that we want on the y axis.
- This gives us a blank set of axis. Now we need to add the points to our axis by using the points() function. For example:

  - > points(speed,dist,col="red",pch=21,bg="red")
  - > points(Speed,Distance,col="blue",pch=21,bg="blue")
- These three commands give us the following graph, and this graph includes all data points

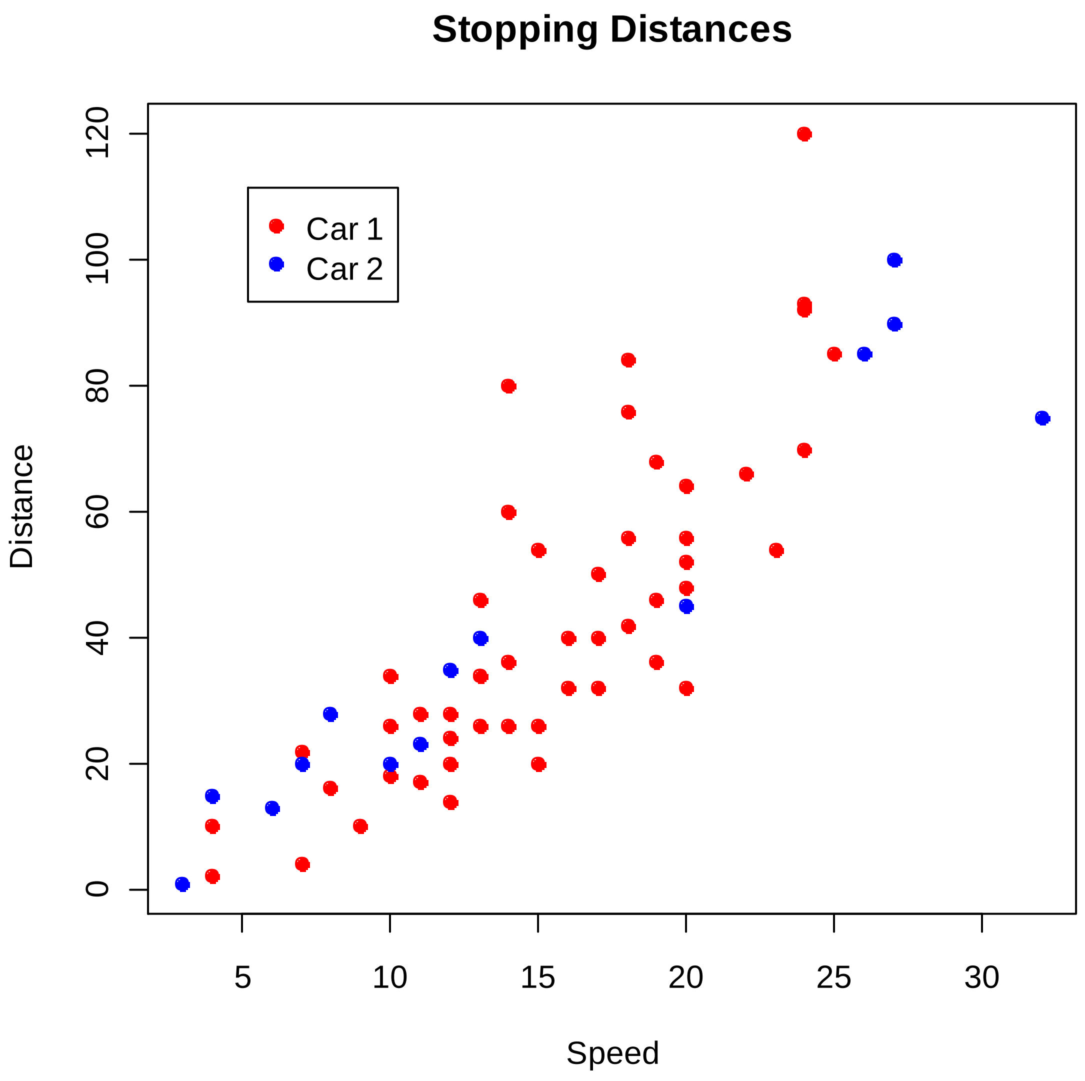


Adding a Legend

- The one thing that we are missing from our scatter plot is a legend. Adding a legend is extremely easy we just need to make sure that we use the concatenate function correctly. Each vector in the legend() function must have the same length. For example:

    - >legend(locator(1),c("Car 1","Car 2"),pch=c(21,21),pt.bg=c("red","blue"),col=c("red","blue"))

- Note how each function we used within legend() has the same vector length or number of vectors
- Also the locator(1) portion of the command allows us to place the top right corner of the legend to a place of your choosing. After you enter the command, simply click where you want the top right corner of the legend box. Here is an example

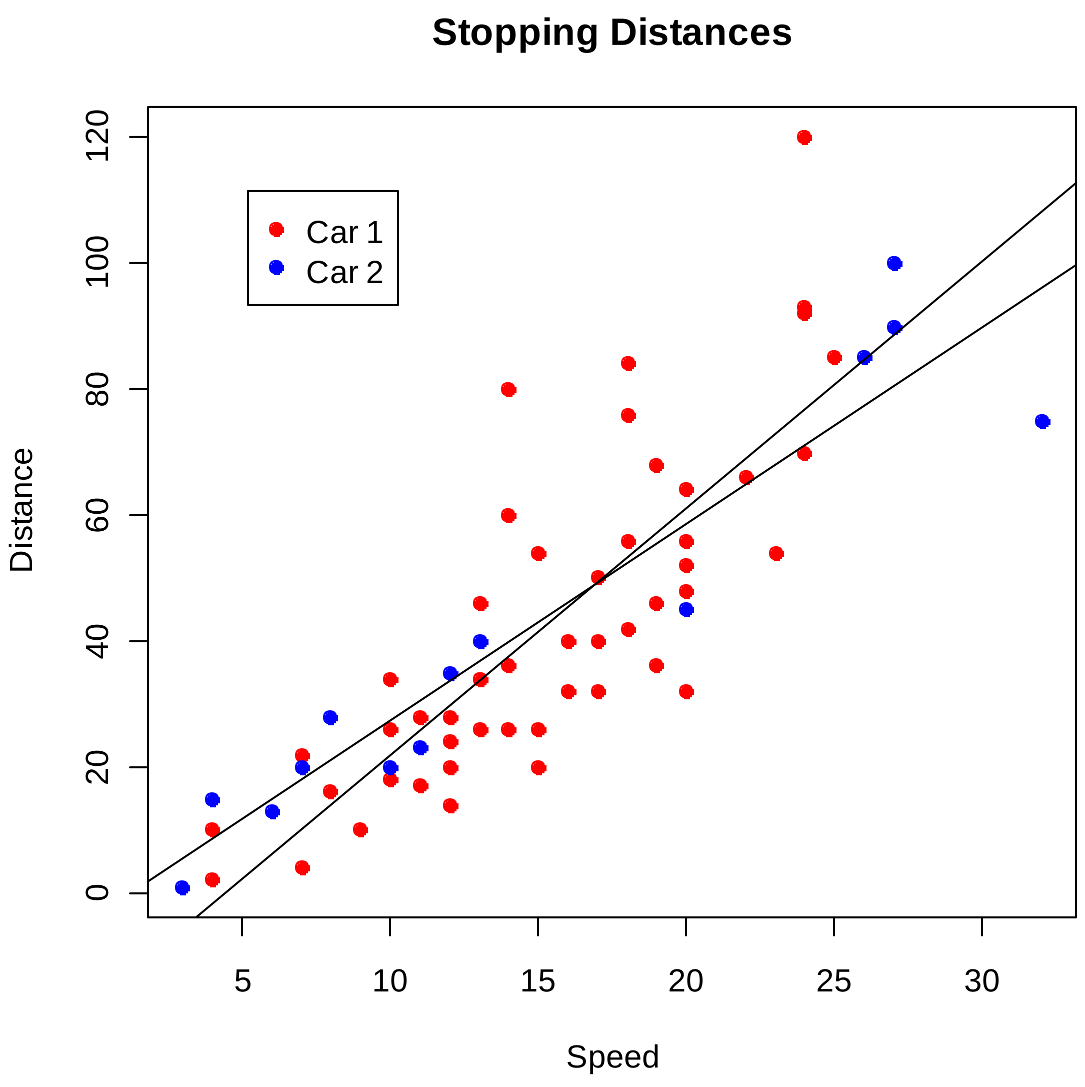


Adding regression lines to each data set

We can now add a regression line to each data set. We simply use the abline(lm()) function as shown above. For example:

- >abline(lm(Distance~Speed))
- >abline(lm(dist~speed))

- The below regression lines are now added to our graph as shown below



LOESS Regression

An extremely useful tool that goes hand in hand in hand with scatter plot analysis is local regression curve fitting or locally weighted scatter-plot smoothing (LOWESS or LOESS). This modern method of curve fitting is favored by the scientific community for its effectiveness and its virtually assumption free nature. In comparison to the older and more common least squares regression analysis, the LOESS curve fitting leaves no room for subjective input from the user making it far more robust for certain applications than the standard line regression method. LOESS does not require the specific input of a function, to which the computer will fit a model of the data.

How LOESS works

LOESS uses a complex algorithm by which local linear polynomial fit is iterated through the data. LOESS uses a smoothing parameter α, a number between 0 and 1, to determine how lenient the computer will be when fitting the curve. A large value of α will make the line more smooth and may help find a broad trend among the data, whereas a small value of α will make the line more sensitive to fluctuations in the data. If α is too large, you may be missing out on some valuable information in the patterns of the data. However an α value that is too small will make the line respond to random error in the data or noise. A good value for α is between .25-.5.

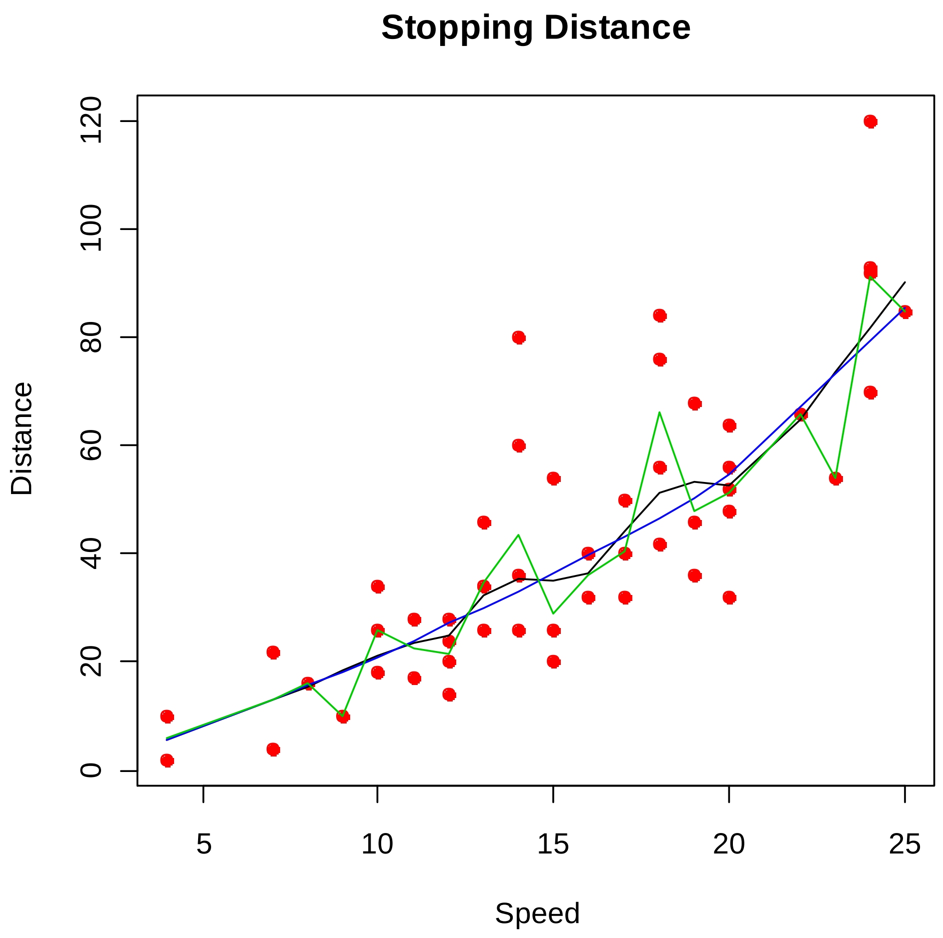
Another facet of the LOESS method is the weight function that it employs. It utilizes the tri-cube weight function which places more emphasis on points that make up a bulk of the information rather than outliers (this is precisely the opposite of variance and standard deviation). The tri-cube weight function looks like $w(x) = (1-|x|^3)^3 \, I\,[|x| < 1]$ Once again, what makes this test so valuable is the notion that it employs, that points that are closer to each other are more likely to be related in a simple way than points that are further apart.

The only disadvantages of LOESS are that it requires a large and densely sampled data set in order to work effectively and that it sometimes lacks the ability to provide a simple mathematical formula for expressing the data.

Let’s use R!

R has a very simple yet powerful function for LOWESS analysis. It follows from the standard draw line function in R. The only input that you need to add to the function template is the name of the bivariate data set, the desired alpha value, which will be called f in R, and the color of the line that you wish to add.

- attach(cars)
- plot(speed,dist,main="Stopping Distance", xlab="Speed", ylab="Distance", pch=21, bg=2, col=2)
- lines(lowess(cars, f=.3),col =1)
- lines(lowess(cars, f=.5),col =4)
- lines(lowess(cars, f=.1),col =7)



The black lines represent the LOWESS iteration with alpha value .3. The blue line is the LOWESS iteration with alpha set to .5, and the green is alpha equals .1. Clearly, the larger the alpha value the more smooth the curve is. With a large alpha value, (blue line) we can see that the general trend is positive and exponential. With the medium value, (black line) we can see that the data seems to be bimodal with two small peaks and 3 small local minimums. When the alpha value is very small (green line) we can really see how the data fluctuates. An interesting question to ask is: what can all of this information tell us? What can we infer from this? That depends on the particular task and the particular application. Here we see that the data appears to have a little bit of a sine curve look to it. However it is important to note that this observation may be based on random sampling. I wouldn't conclude anything unless I had more information and more points.