Data Visualization for arbitrary data sets

Data Visualization for arbitrary data sets

Executive Summary:

This document describes methods to visualize arbitrary data sets. This uses the 'R Programming Language' - a language used predominantly in statistics. In this project, we are using built in graphics packages with our custom R functions. We are able to analyze one dimensional data, two dimensional data and correlations between multi dimensional data. This is very useful in doing exploratory data analysis and to understand the relationship between data better. In plotting 2 dimensional data, we are able to deal with non numeric data as well.

Limitation and improvements:

If the data is not formed properly, the plots won't be drawn. One of the improvements is to make the data work well.

Case-1 SP500 Stock market returns by year visualization

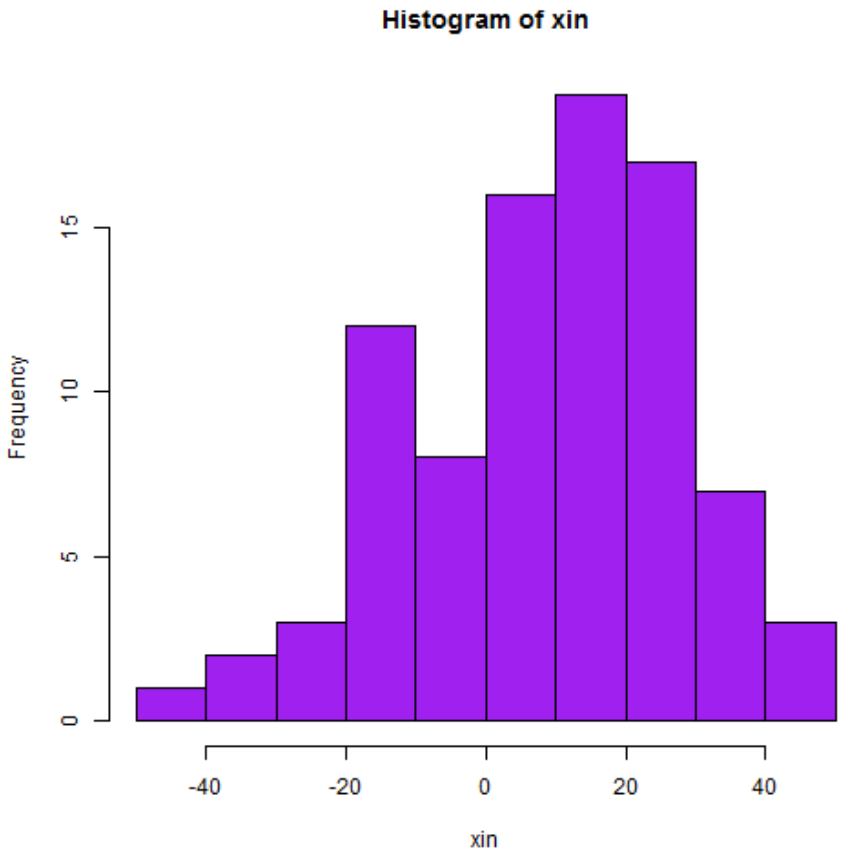
The first plot shows the histogram of the returns. It takes only one set of variables. The second graph shows the distribution of the returns. The third graph shows the returns by year and fits a polynomial statistcal function to fit the data. The fourth graph joins the data points with a line.

```
install.packages(c("ggplot2","gcookbook"))
```

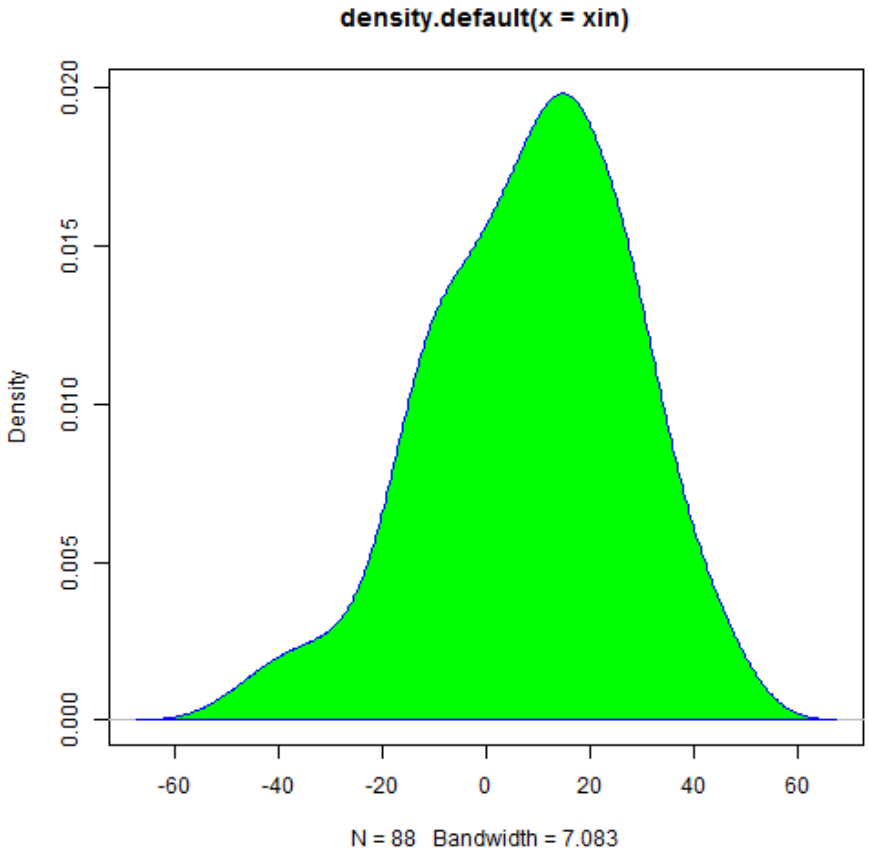
```
## Warning: packages 'ggplot2', 'gcookbook' are in use and will not be
## installed
```

```
library('ggplot2')
library('gcookbook')
source("Plot2D.R")
stkdata <- read.csv("SP500RetByYearCleaned.csv",sep=",")
```

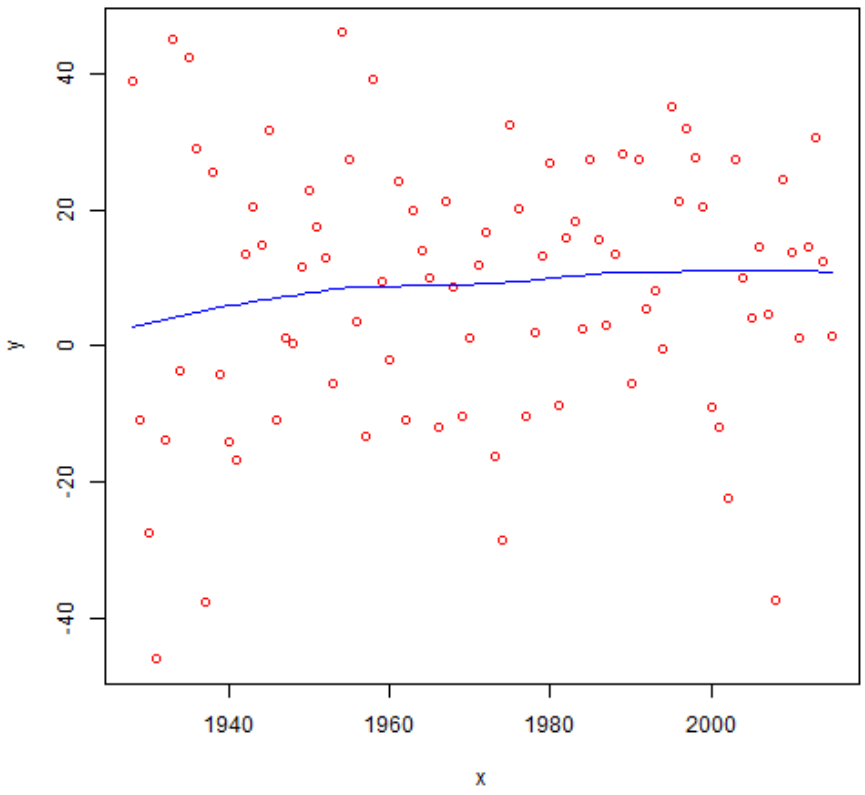
```
source("PlotHist.R")
plothist(stkdata$return,"purple")
```



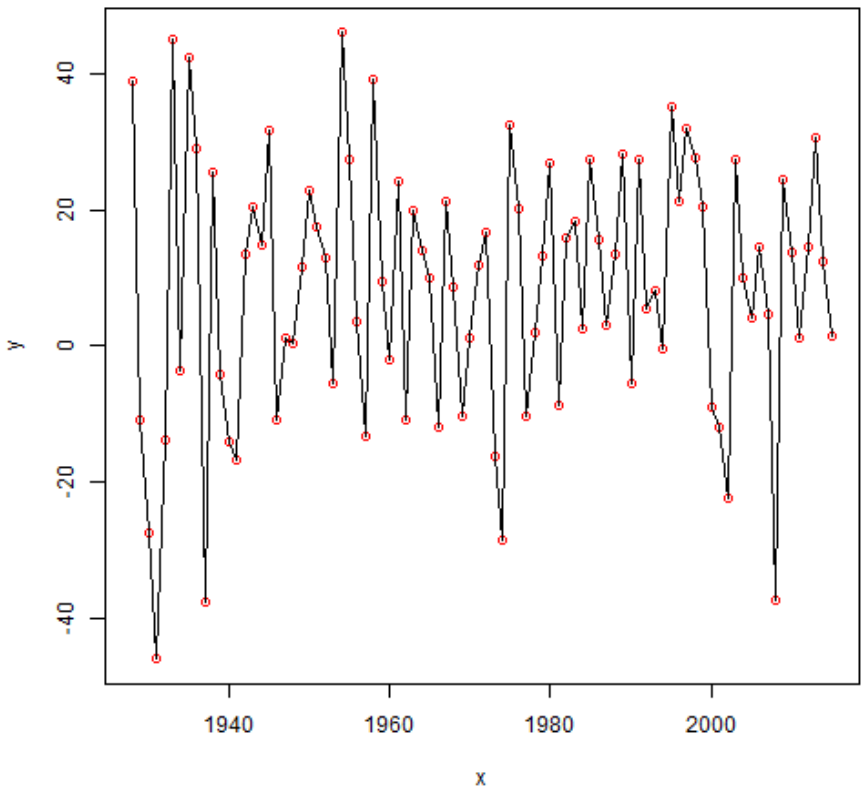

```
source("PlotDensity.R")
plotdensity(stkdata$return,"green")
```




```
plot2d(stkdata$year,stkdata$return)
```




```
plot2d(stkdata$year,stkdata$return,TRUE,FALSE)
```



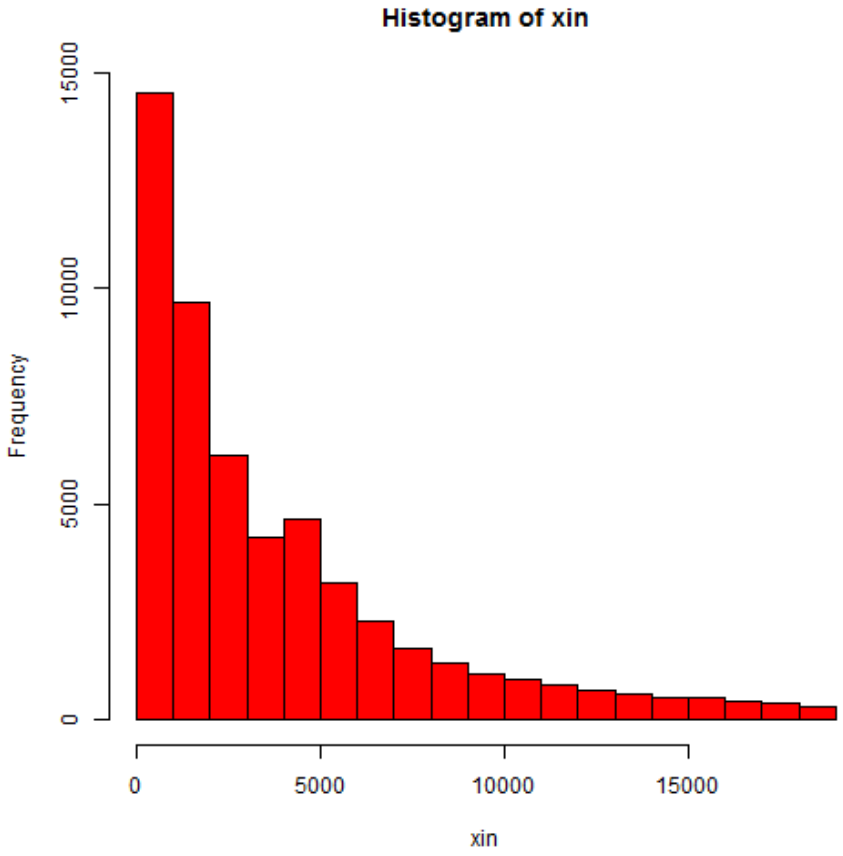
Diamond data set.

After exploring the data set, we look at the relationship betwen carat and price. Then we look at the histogram of carats and prices to understand its distribution. We repeat the same functions using the density function. The graphs below show the visualization for diamond data set by one set of variables.

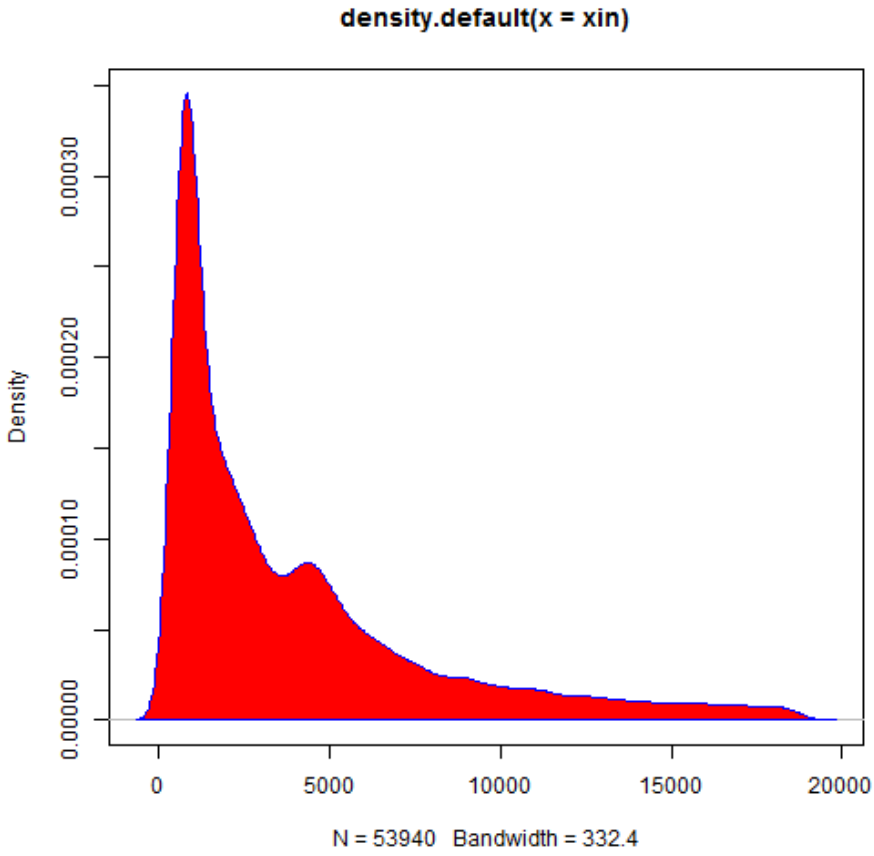
```
head(diamonds,n=1)
```

```
##   carat   cut color clarity depth table price    x    y    z
## 1  0.23 Ideal     E     SI2  61.5    55   326 3.95 3.98 2.43
```

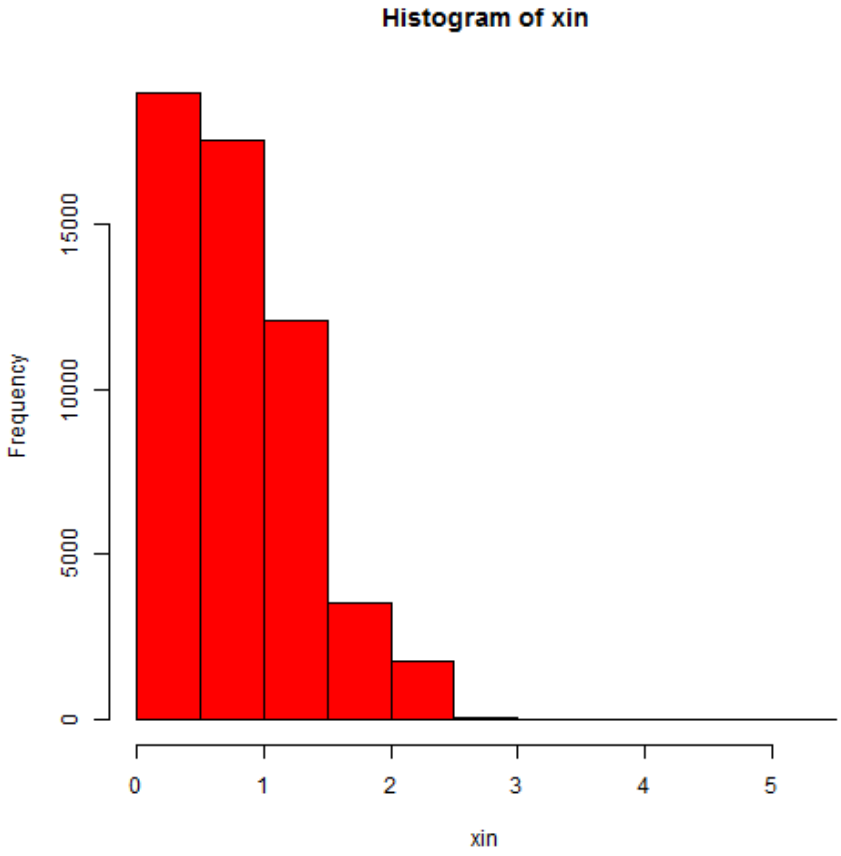
```
plothist(diamonds$price)
```



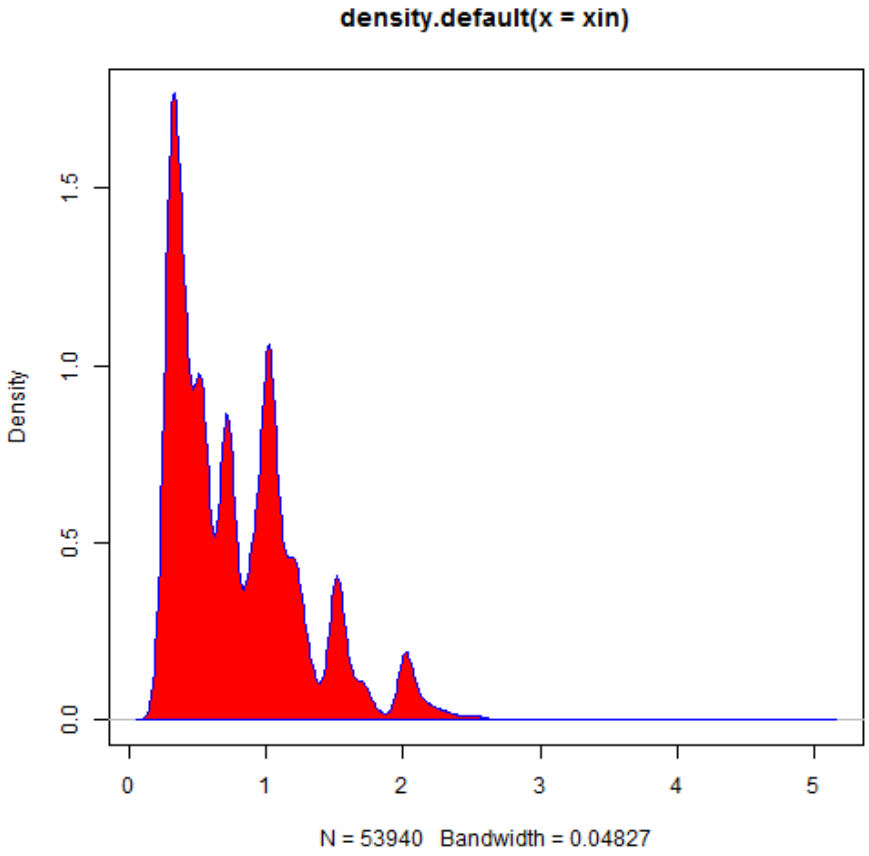

```
plotdensity(diamonds$price)
```




```
plothist(diamonds$carat)
```

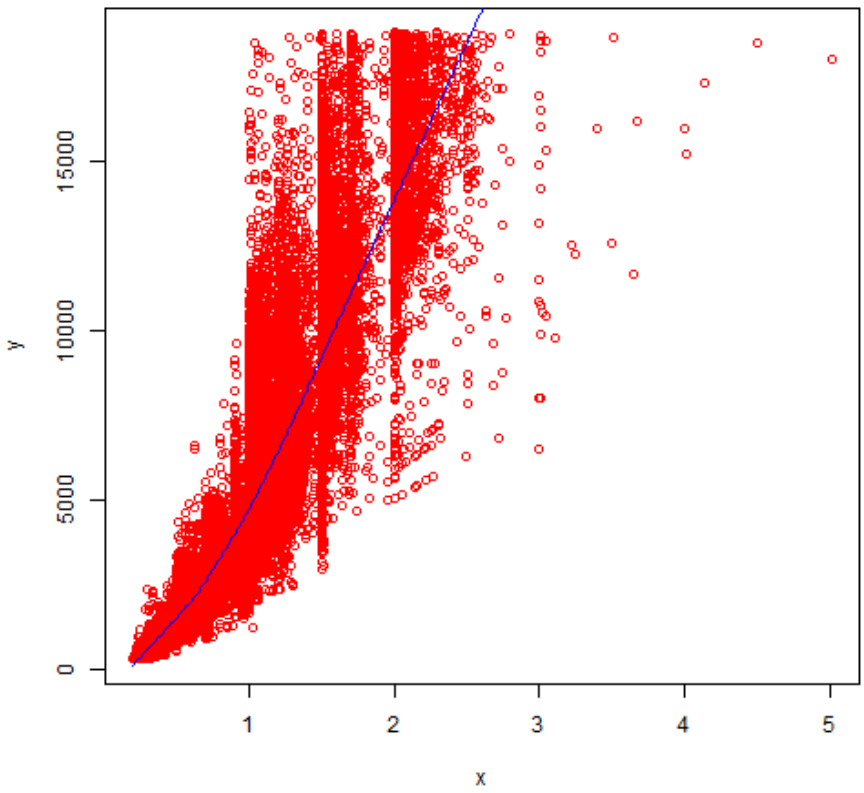



```
plotdensity(diamonds$carat)
```

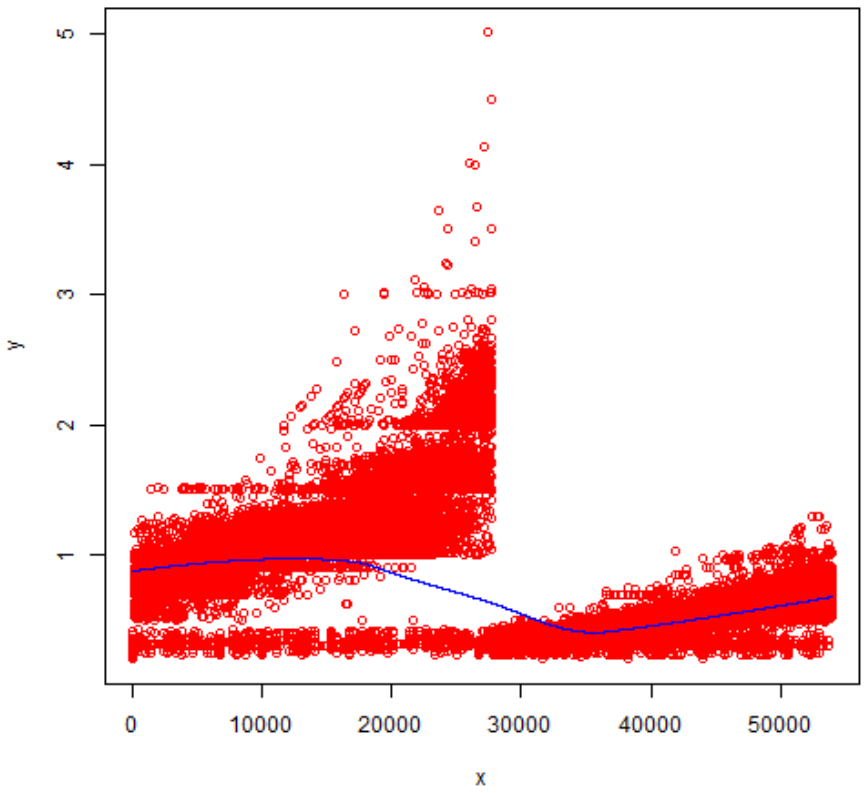


These two plots show the relationship between carat and price followed by cut and carat. The cut type is not a numeric variable.

```
plot2d(diamonds$carat,diamonds$price)
```




```
plot2d(diamonds$cut,diamonds$carat)
```



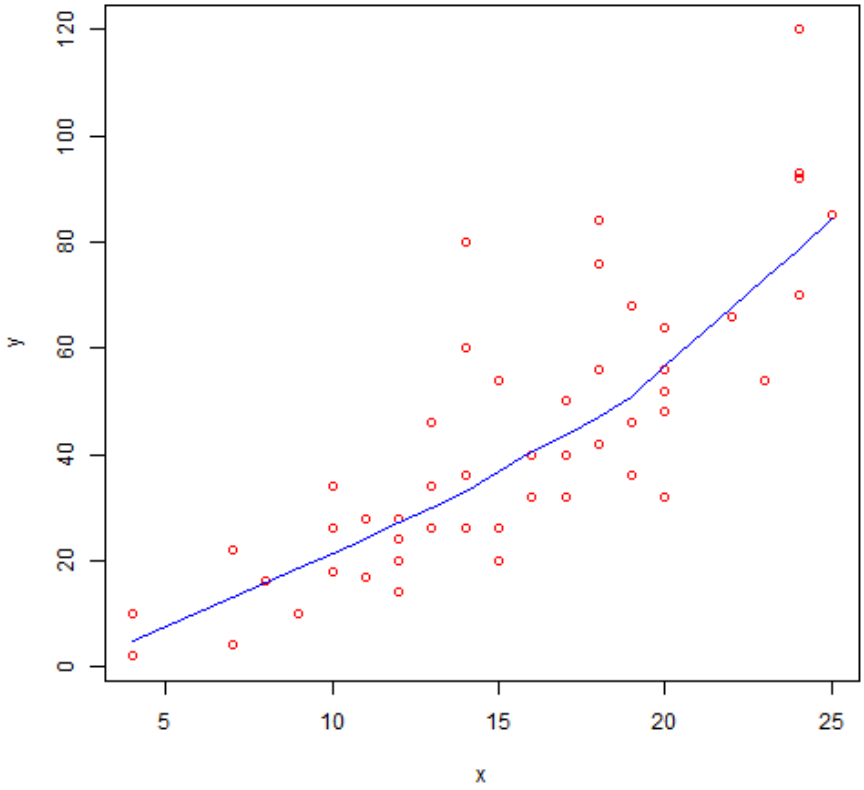
Distance needed for a car to stop at different speeds

We use different plots to explore the data set and then find the relationship between speed and distance to stop.

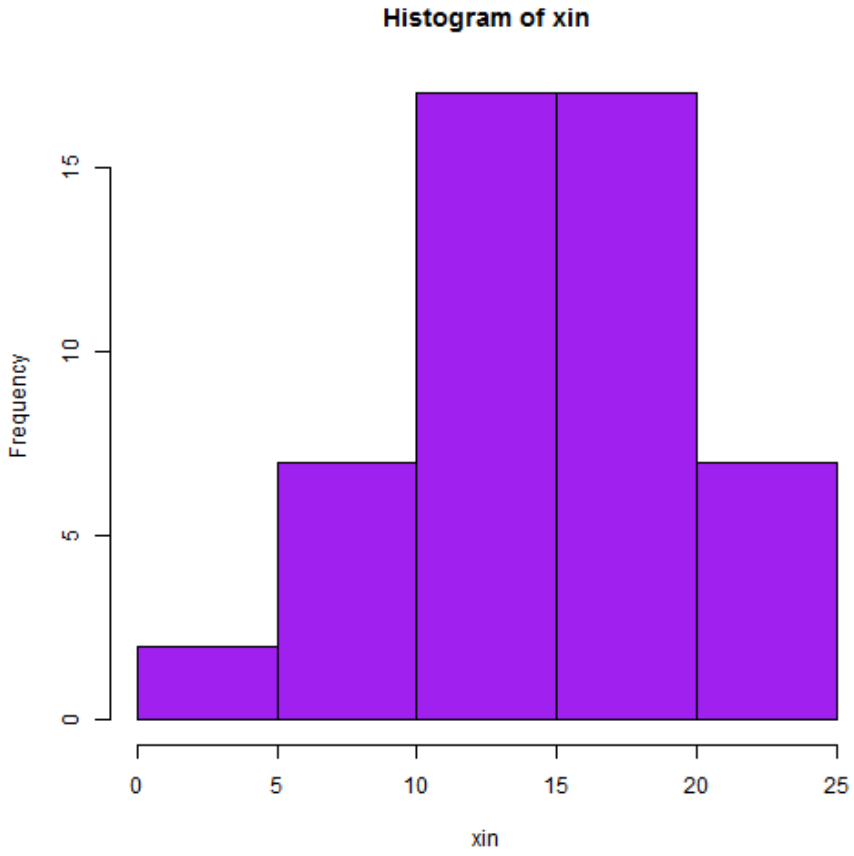
```
head(cars,n=1)
```

```
##   speed dist
## 1     4    2
```

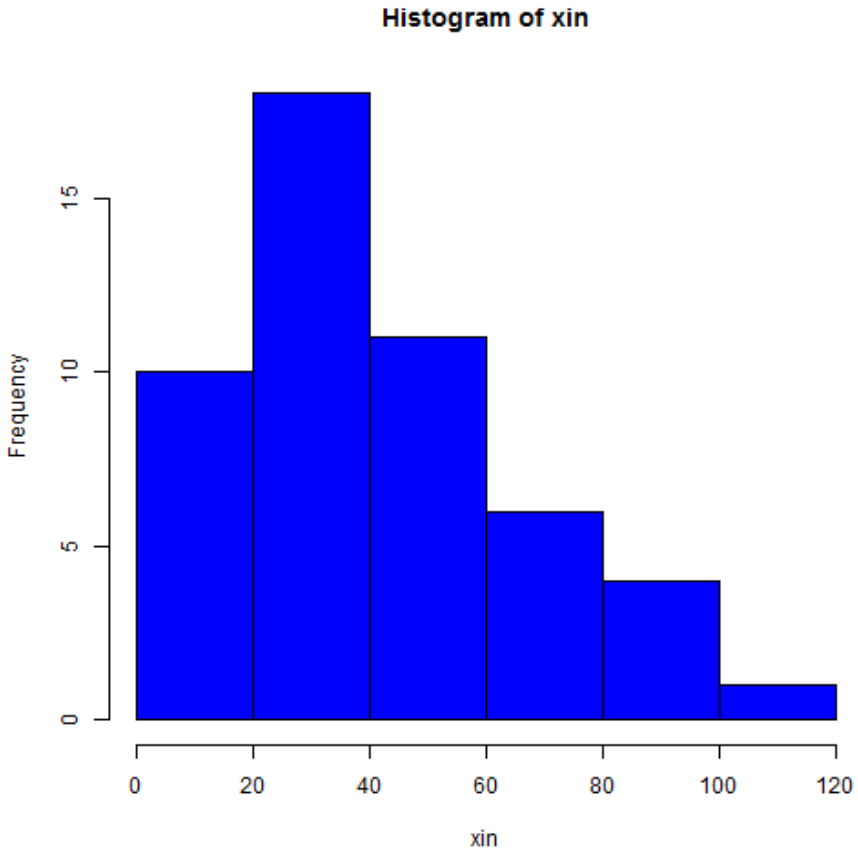
```
plot2d(cars$speed,cars$dist)
```



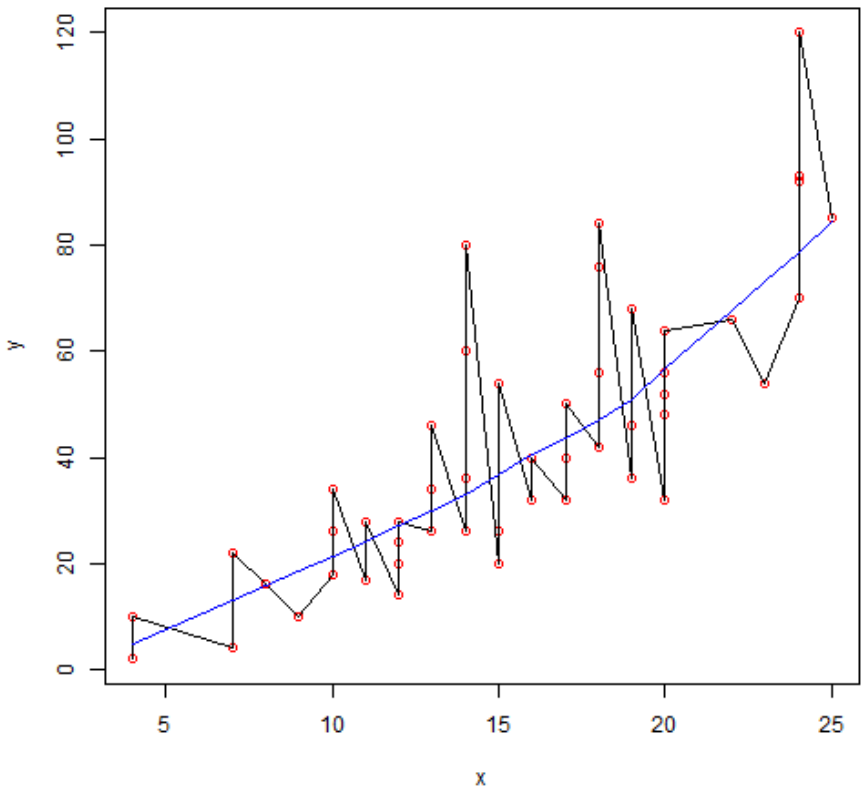

```
plothist(cars$speed,"purple")
```



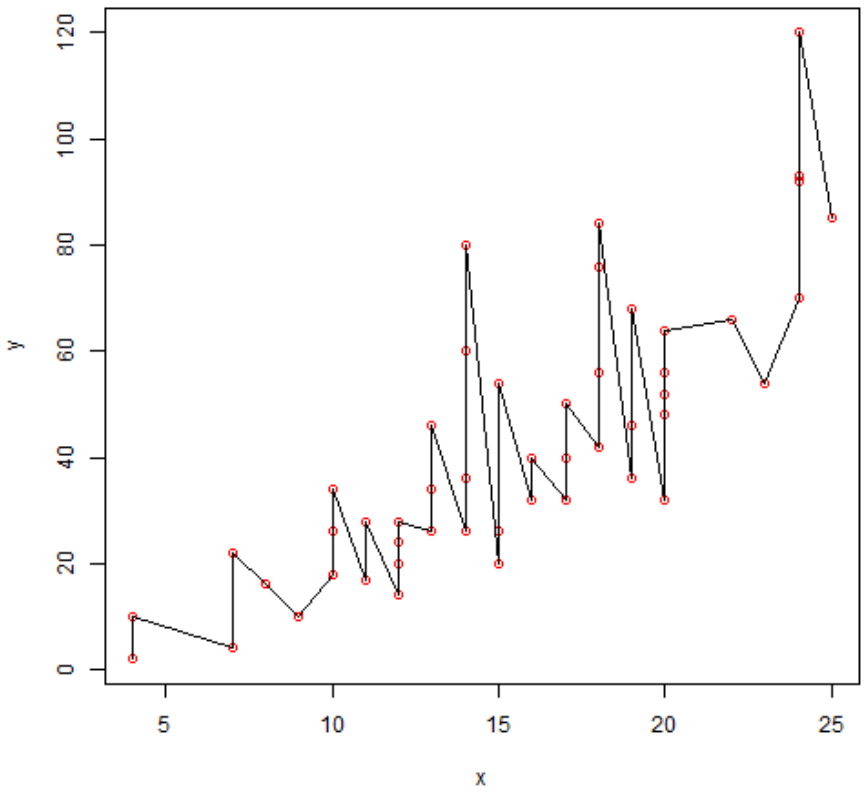

```
plothist(cars$dist,"blue")
```




```
plot2d(cars$speed,cars$dist,TRUE,TRUE)
```




```
plot2d(cars$speed,cars$dist,TRUE,FALSE)
```



Parent Child height using Galton dataset

The Galton data set computes the height of children given parent's heights. The data shows that children of taller parents tend to be shorter and the children of taller parents tend to be shorter.

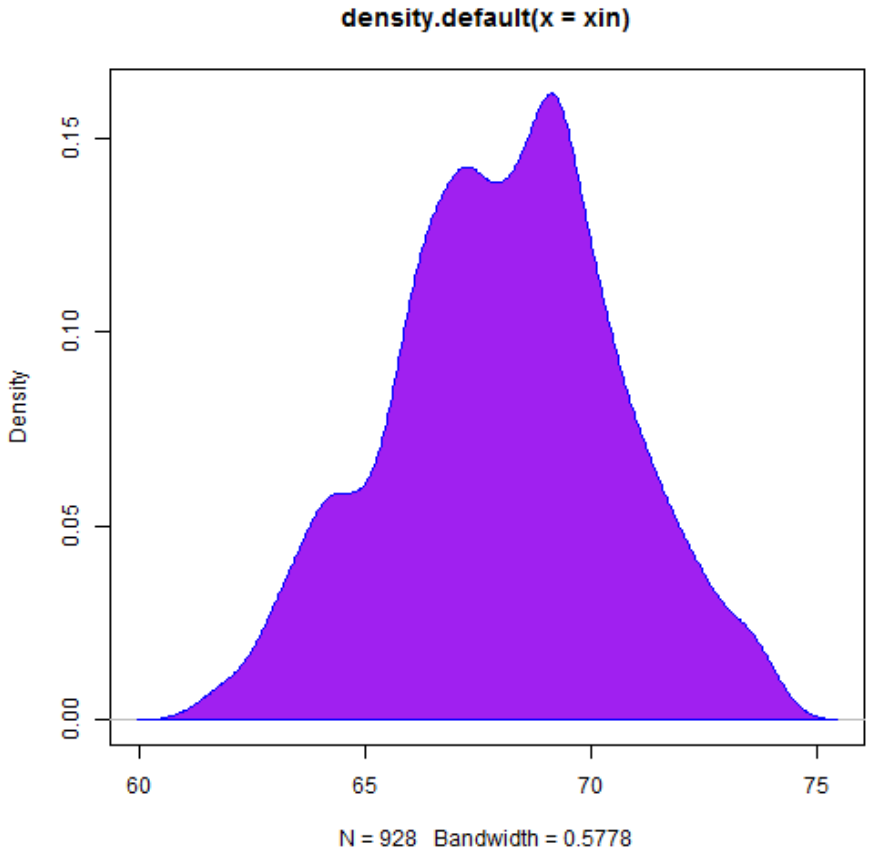
```
install.packages("UsingR")
```

```
## Warning: package 'UsingR' is in use and will not be installed
```

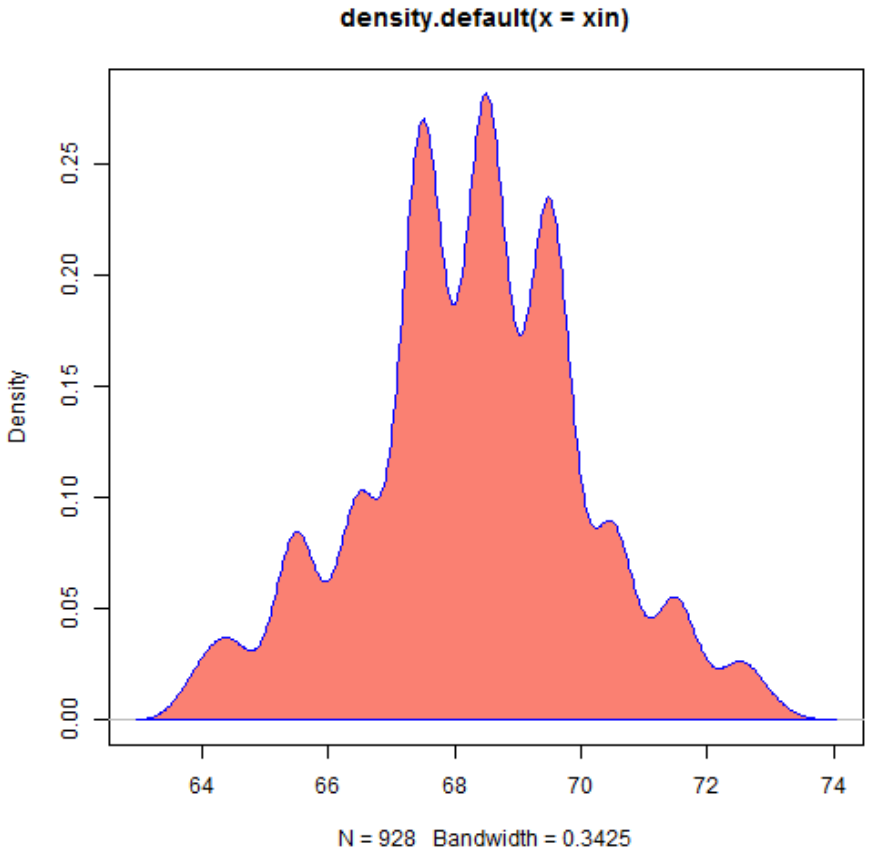
```
library(UsingR)
data(galton)
head(galton,n=1)
```

```
##   child parent
## 1  61.7   70.5
```

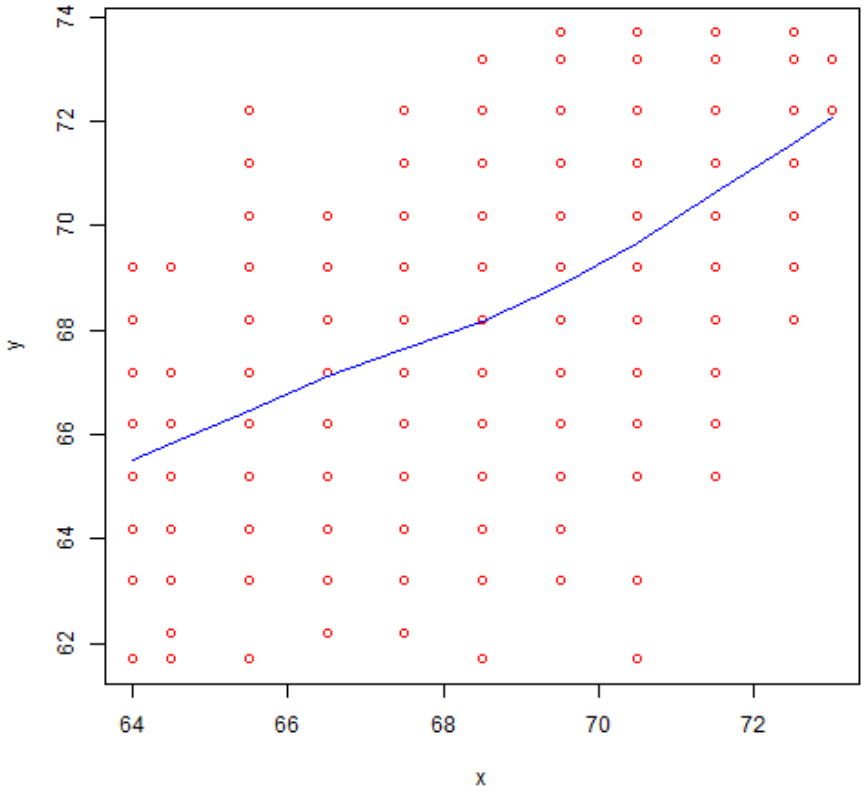
```
plotdensity(galton$child,"purple")
```




```
plotdensity(galton$parent,"salmon")
```




```
plot2d(galton$parent,galton$child)
```



US Population by age group Visualization

In this data set, we explore the US population data set and then plot it using the plot2d function to explore the relationship between data.

```
head(uspopage,n=10)
```

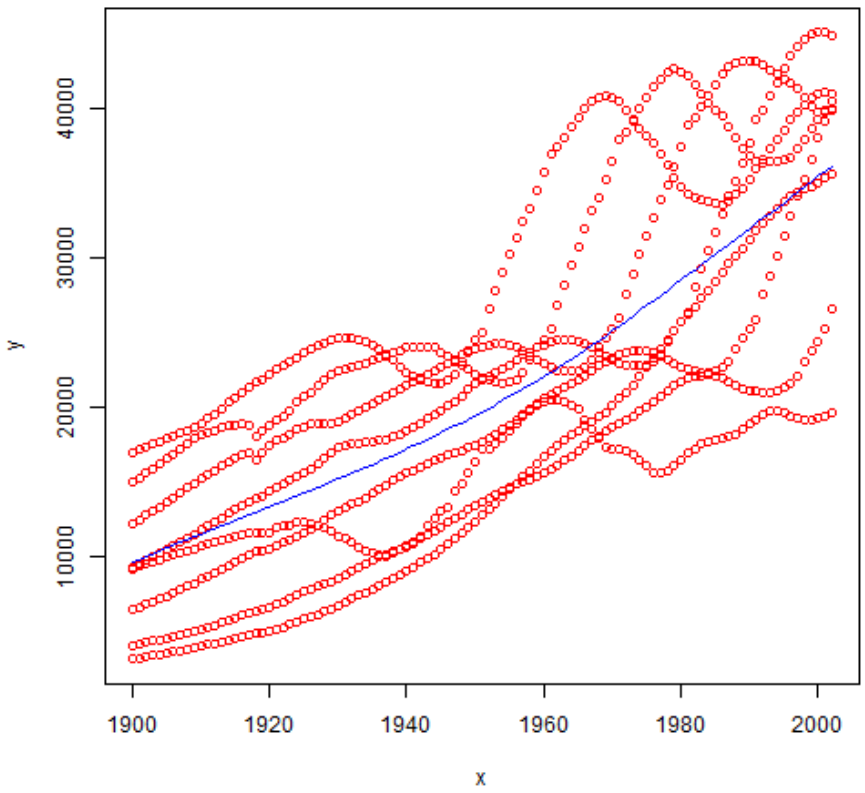
```
##    Year AgeGroup Thousands
## 1  1900       <5       9181
## 2  1900     5-14      16966
## 3  1900    15-24      14951
## 4  1900    25-34      12161
## 5  1900    35-44       9273
## 6  1900    45-54       6437
```

```
## 7  1900    55-64      4026
## 8  1900      >64      3099
## 9  1901       <5      9336
## 10 1901     5-14     17158
```

```
tail(uspopage,n=10)
```

```
##     Year AgeGroup Thousands
## 815 2001    55-64     25315
## 816 2001      >64     35352
## 817 2002       <5     19609
## 818 2002     5-14     41037
## 819 2002    15-24     40590
## 820 2002    25-34     39928
## 821 2002    35-44     44917
## 822 2002    45-54     40084
## 823 2002    55-64     26602
## 824 2002      >64     35602
```

```
plot2d(uspopage$Year,uspopage$Thousands)
```



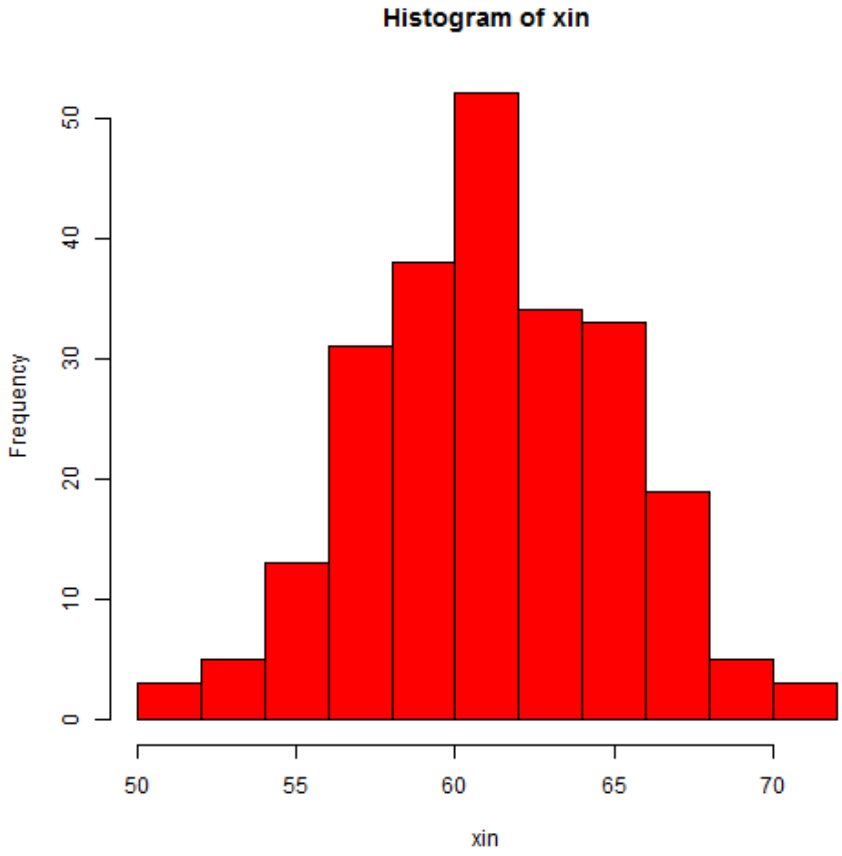
Height and weight of school aged children

This dataset explores the height and weight of school aged children. We visualize the data using histogram and plot2d functions.

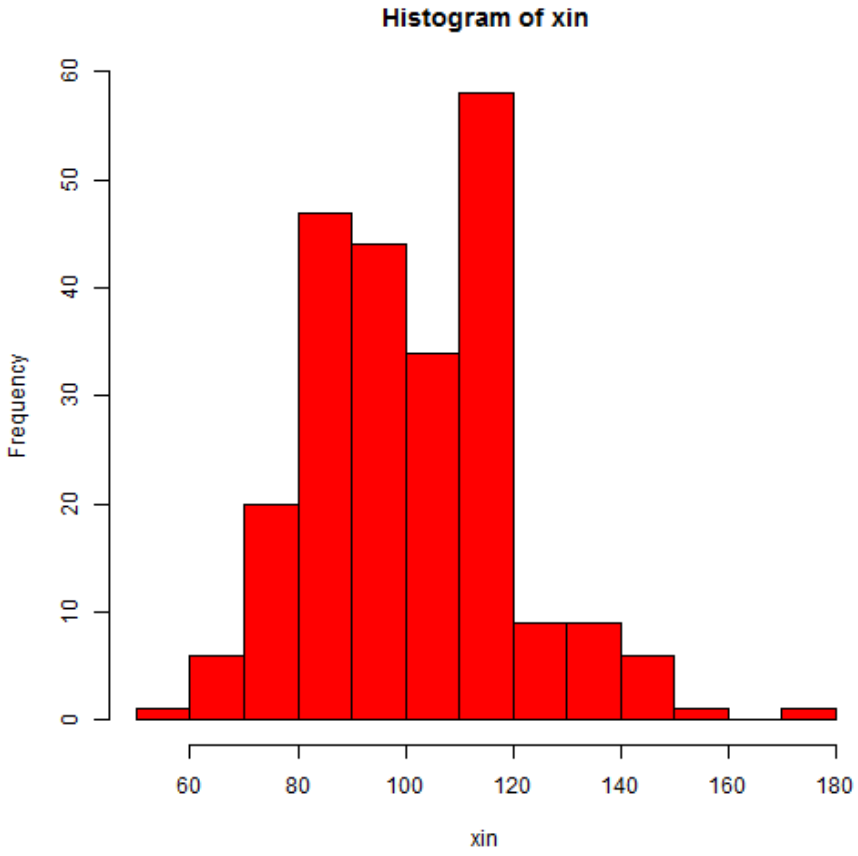
```
head(heightweight,n=1)
```

```
##   sex ageYear ageMonth heightIn weightLb
## 1   f   11.92      143     56.3       85
```

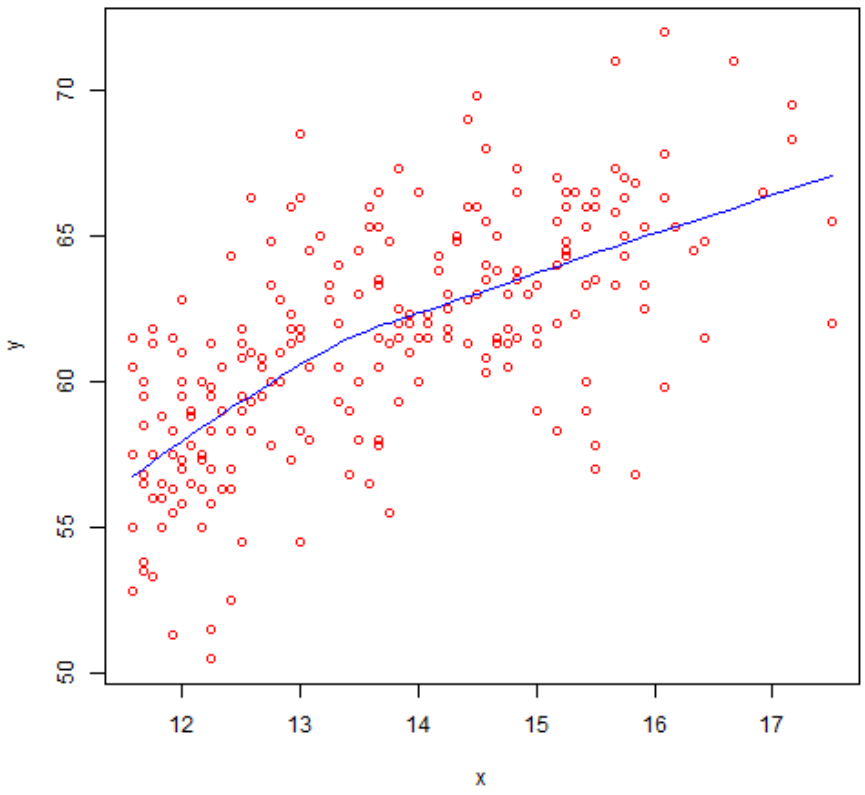
```
plothist(heightweight$heightIn)
```




```
plothist(heightweight$weightLb)
```



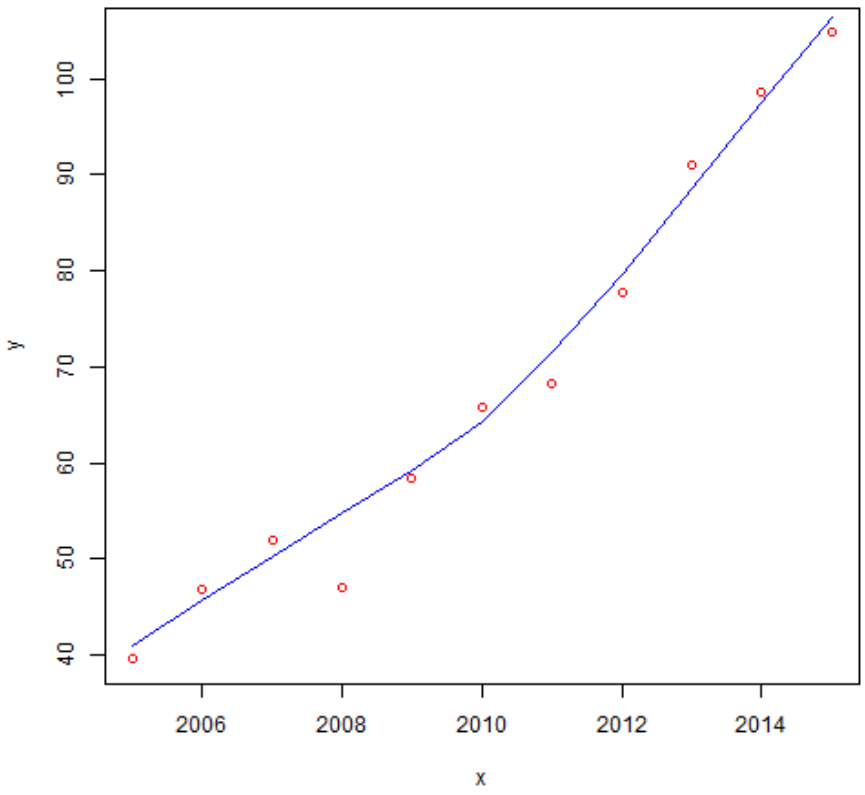

```
plot2d(heightweight$ageYear,heightweight$heightIn)
```



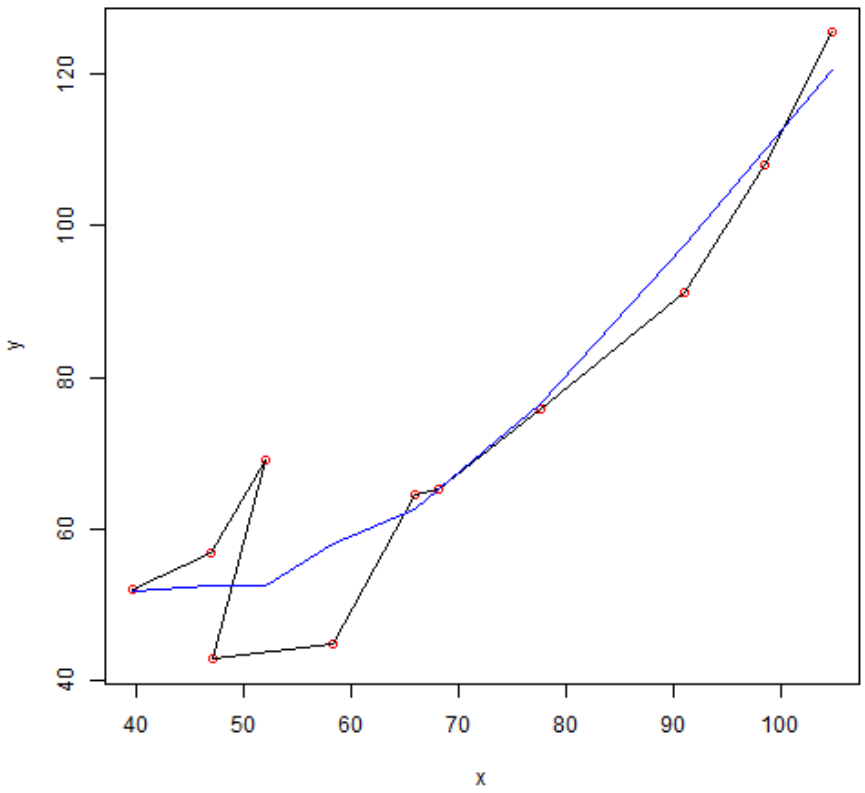
Berkshire Hathaway Book Value Growth, Low Price and High Price

In this example, we look at the relationship between Berkshire Hathaway's book value growth by year, book value and low price, book value and high price. Finally, we look at the correlation between various data points using pairplot.

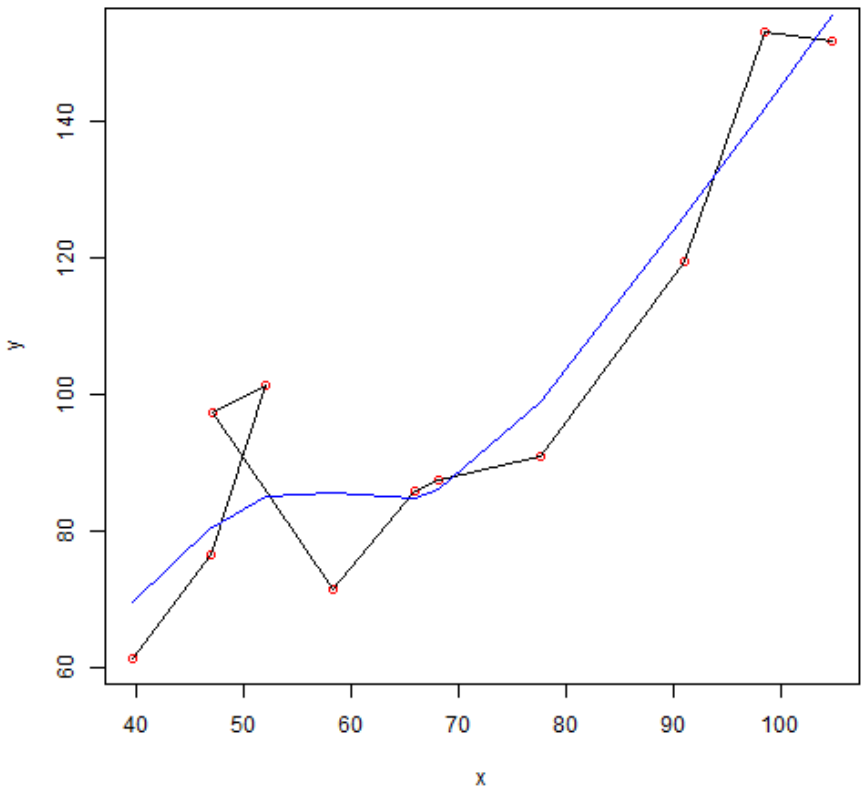
```
bkdata <- read.csv("BerkshireBookValue.csv",sep=",")
plot2d(bkdata$year,bkdata$bookval)
```



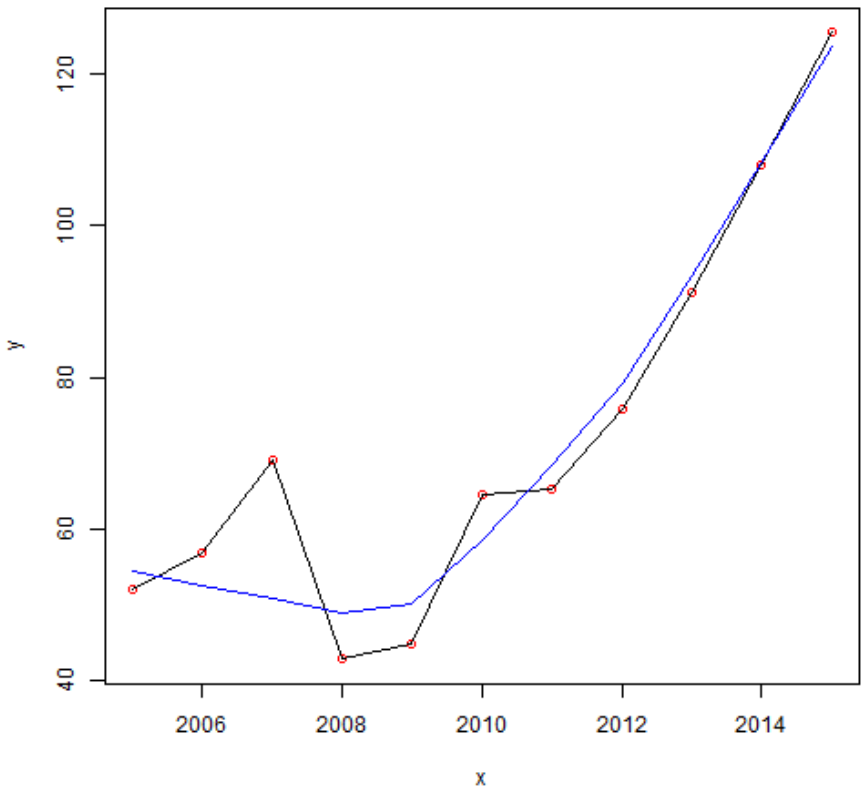

```
plot2d(bkdata$bookval,bkdata$lowpric,TRUE,TRUE)
```




```
plot2d(bkdata$bookval,bkdata$highpric,TRUE,TRUE)
```




```
plot2d(bkdata$year,bkdata$lowpric,TRUE,TRUE)
```



Examining correlations between different data points in Berkshire book value, low price and high price

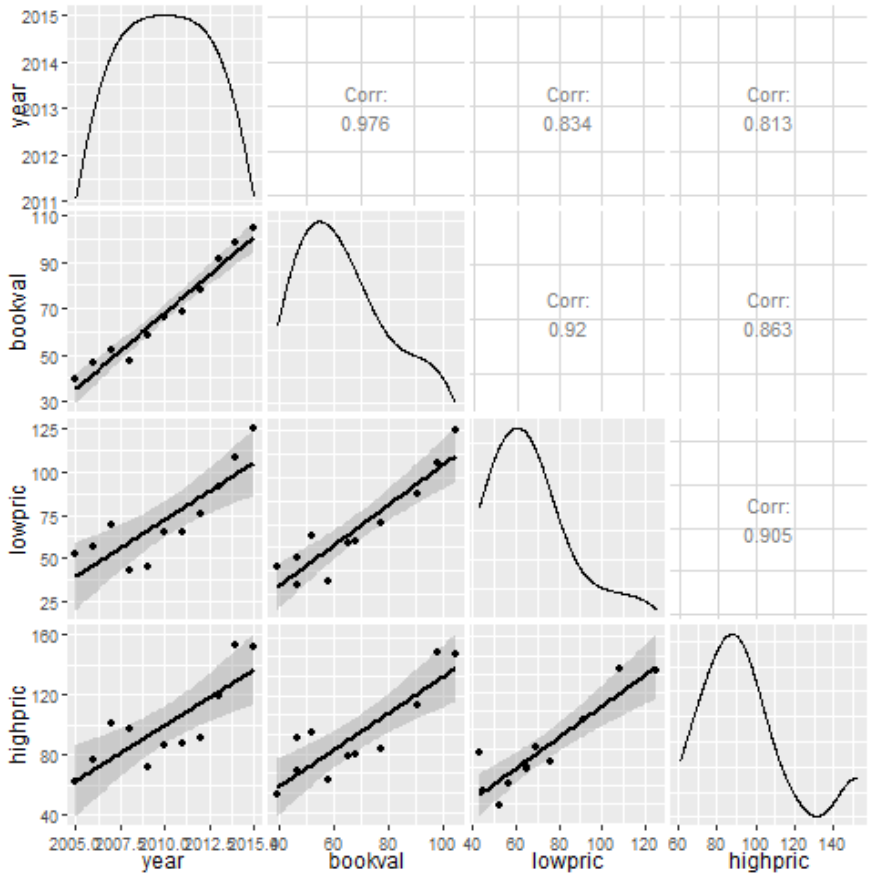
```
source("DrawPairPlot.R")
```

```
## Warning: package 'GGally' is in use and will not be installed
```

```
## Warning: package 'ggplot2' is in use and will not be installed
```

```
## Warning: package 'UsingR' is in use and will not be installed
```

```
drawpairplot(bkdata)
```



Mileage by weight, transmission type - mtcars data set

This data set shows the miles per gallon obtained by different types of vehicles with different types of engines, weights and transmissions. We explore the relationship between various factors using the drawpairplot function.

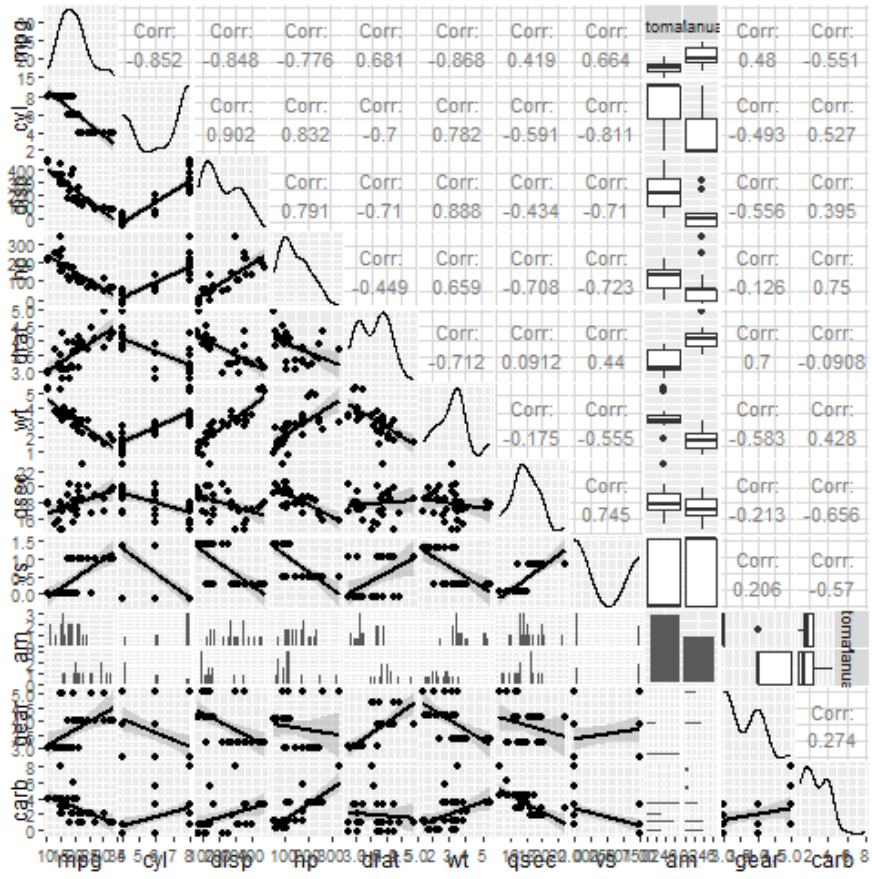
```
head(mtcars,n=1)
```

```
##           mpg cyl disp  hp drat   wt  qsec vs     am gear carb
## Mazda RX4  21   6  160 110  3.9 2.62 16.46  0 Manual    4    4
```

```
drawpairplot(mtcars)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

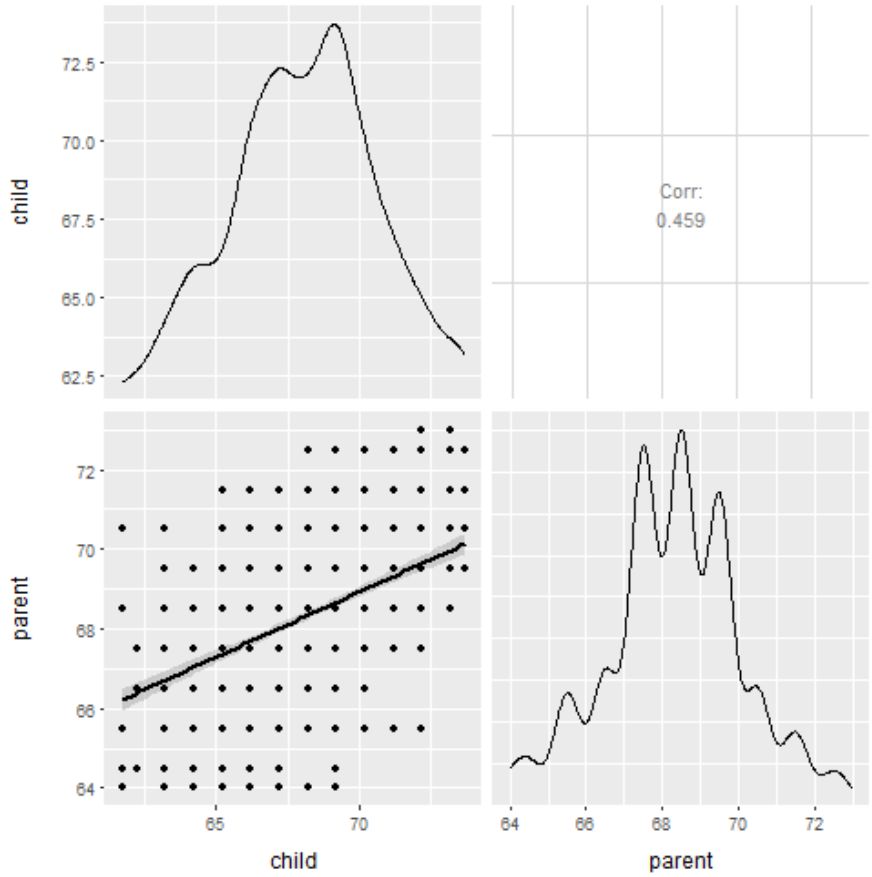
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Pairplot for Galton data set on the heights of parents and children

In this example, we explore the relationship between parents and childrens heights using drawpairplot function.

```
drawpairplot(galton)
```

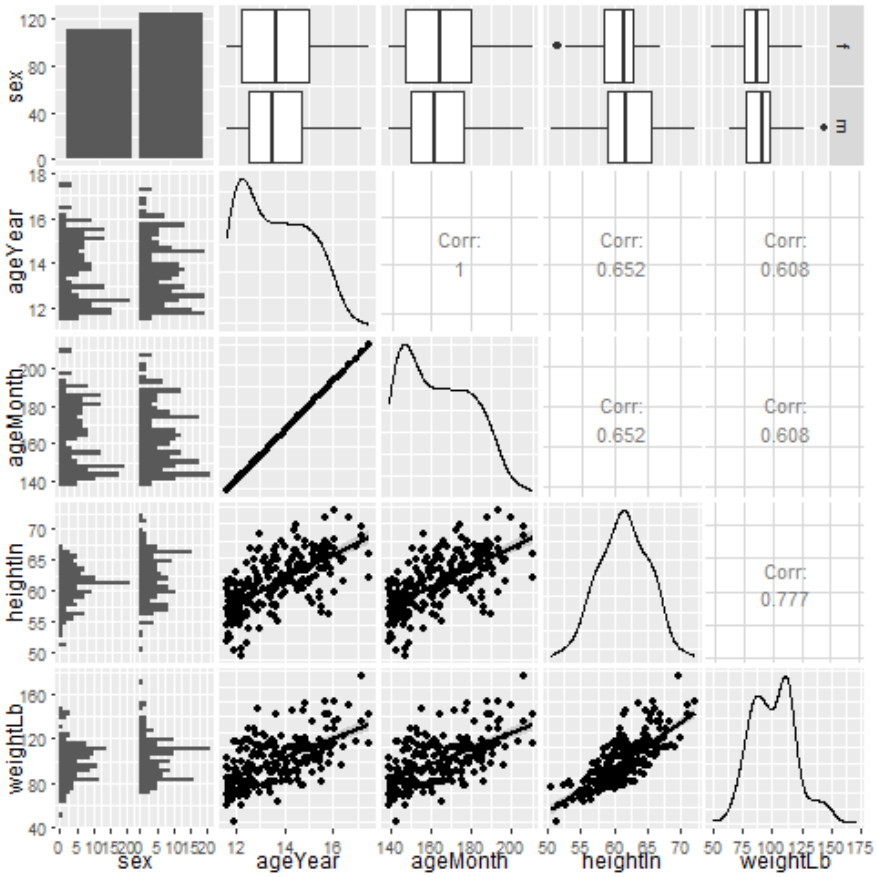


Pairplot for Height and Weight of School Aged Children

In this example, we explore the relationship between height and weight of school aged children using the pairplot function.

```
drawpairplot(heightweight)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


sex
ageYear
ageMonth
heightIn
weightLb
F
M
Corr:
1
Corr:
0.652
Corr:
0.608
Corr:
0.652
Corr:
0.608
Corr:
0.777