R Notebook

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

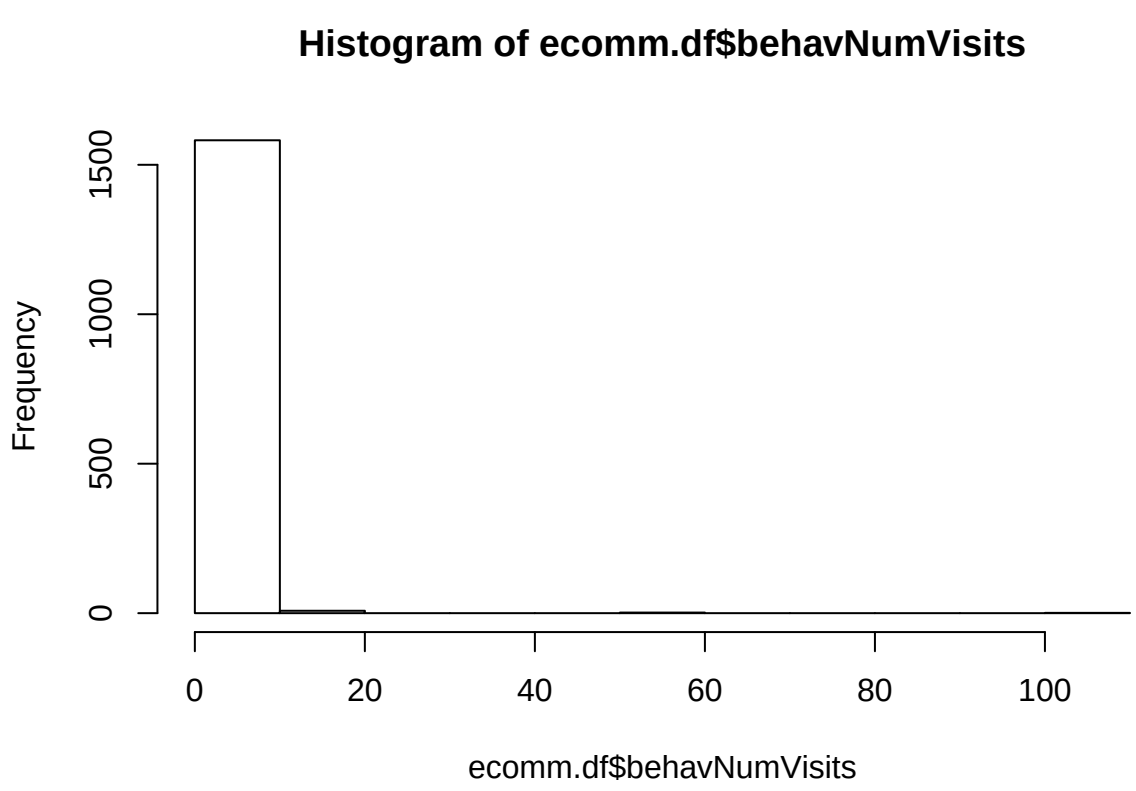
Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

```
ecomm.df<-read.csv("ecommerce-data.csv")
str(ecomm.df)
```

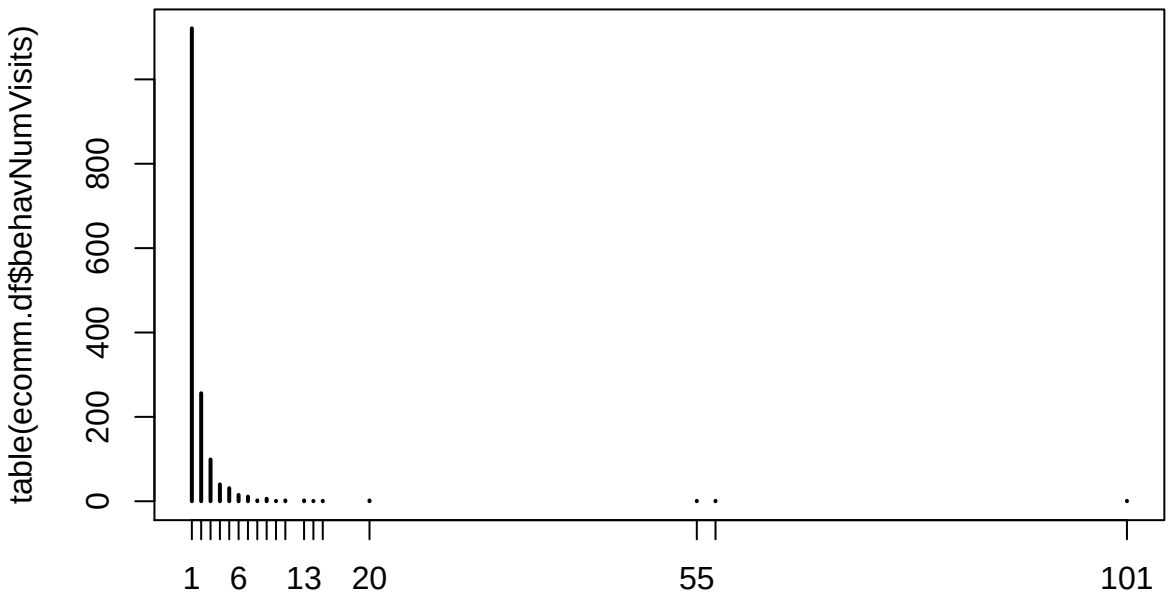
```
## 'data.frame':    1593 obs. of  45 variables:
##  $ dateTime                   : Factor w/ 1558 levels "7/25/2014 14:10",..: 1 2 3 4 5 6 14 7 8 9 ...
##  $ country                    : Factor w/ 44 levels "Australia","Barbados",..: 44 44 44 44 44 44 17 4
##  $ city                       : Factor w/ 980 levels "","Abilene","Abingdon",..: 563 25 76 158 132 4
##  $ region                     : Factor w/ 110 levels "","0","1","10",..: 67 94 102 104 35 40 10 80 1
##  $ screenRed                  : Factor w/ 91 levels "1012x569","1024x552",..: 29 34 76 35 17 43 76 7
##  $ surveyType                 : Factor w/ 3 levels "At Arrival and Exit",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ purposeProductInfo         : Factor w/ 2 levels "","Products": 2 1 1 2 1 2 1 2 1 1 ...
##  $ purposeBuyFromSite         : Factor w/ 2 levels "","Buy from this site": 1 2 1 1 1 1 1 1 1 2 ...
##  $ purposeComparePricing      : Factor w/ 2 levels "","Compare pricing": 1 2 2 1 1 2 1 1 1 1 ...
##  $ purposeInfoAndResources    : Factor w/ 2 levels "","Resources": 2 1 1 1 2 1 2 1 1 1 ...
##  $ purposeInfoOnOrder         : Factor w/ 2 levels "","Order info": 1 1 1 1 1 1 1 1 1 1 ...
##  $ purposeOther               : Factor w/ 2 levels "","Other": 1 1 1 1 1 1 1 1 2 1 ...
##  $ taskFindWhatLookingFor     : Factor w/ 4 levels "","Most or all of it",..: 1 1 1 2 2 2 4 2 4 2 ..
##  $ concernShippingCost        : Factor w/ 2 levels "","Shipping costs": 1 1 1 2 1 1 1 1 1 1 ...
##  $ concernDeliverySpeed       : Factor w/ 2 levels "","Fast delivery": 1 1 1 1 1 1 1 1 1 1 ...
##  $ concernWarranties          : Factor w/ 2 levels "","Warranties/product guarantees": 1 1 1 1 1 1 1
##  $ concernEaseToReturnProduct : Factor w/ 2 levels "","Ease of returning (if I am not satisfied with
##  $ concernProductSafety       : Factor w/ 2 levels "","Product safety": 1 1 1 1 1 1 1 1 1 1 ...
##  $ concernRightForMyChild     : Factor w/ 2 levels "","Whether this is right for my child": 1 1 1 1 1
##  $ concernProductQuality      : Factor w/ 2 levels "","Product durability/quality": 2 1 1 1 1 1 1 1 1
##  $ concernProductEffectiveness: Factor w/ 2 levels "","Product effectiveness/will it work": 2 1 1 1 1
##  $ concernOther               : Factor w/ 2 levels "","Other": 1 1 1 1 1 1 1 1 1 1 ...
##  $ concernNone                : Factor w/ 2 levels "","None / no uncertainties": 1 1 1 1 1 1 1 1 2 1
##  $ intentWasPlanningToBuy     : Factor w/ 4 levels "","No","Partially (I was considering it)",..: 1 4
##  $ profile                    : Factor w/ 8 levels "0","Friend/family friend",..: 5 5 5 6 8 6 3 4 8 5
##  $ whenSiteUsed               : Factor w/ 6 levels "","In the past month",..: 3 4 6 6 6 6 3 3 6 6 ..
##  $ purchasedBefore            : Factor w/ 4 levels "","No","Yes, more than once",..: 4 4 1 1 1 1 2 4
##  $ purchasedWhen              : Factor w/ 5 levels "","In the past month",..: 2 4 1 1 1 1 1 2 1 1 ..
##  $ productKnewWhatWanted      : Factor w/ 4 levels "","No","Somewhat",..: 4 4 4 3 1 3 1 1 1 4 ...
##  $ productSiteHasWhatWanted   : Factor w/ 5 levels "","No","Not sure",..: 1 1 1 5 1 5 1 1 1 5 ...
##  $ purchaseExpectInNextMonth  : int  5 3 3 3 5 3 5 NA 5 4 ...
##  $ siteFirstHeardAbout        : Factor w/ 6 levels "","In the past hour",..: 4 6 5 2 5 2 3 1 5 1 ...
##  $ age                        : Factor w/ 9 levels "","18-24","25-34",..: 3 4 4 3 6 2 6 1 5 1 ...
##  $ gender                     : Factor w/ 4 levels "","Female","Male",..: 2 2 2 2 2 4 2 1 2 1 ...
##  $ behavNumVisits             : int  13 3 2 1 1 1 4 1 2 2 ...
##  $ behavReferral              : Factor w/ 9 levels "","Branded Search",..: 3 9 9 9 6 8 3 9 6 9 ...
```

```
##  $ behavPageviews            : Factor w/ 6 levels "0","1","10+",..: 5 2 3 3 2 3 3 5 3 6 ...
##  $ behavHomePage             : int  1 0 0 0 0 1 0 1 1 1 ...
##  $ behavDetailProdA          : int  1 0 0 1 0 1 1 0 1 1 ...
##  $ behavDetailProdB          : int  0 0 0 1 0 1 1 1 1 0 ...
##  $ behavDetailProdC          : int  0 0 0 0 0 0 1 0 1 0 ...
##  $ behavAnySolution          : int  0 0 1 1 0 0 1 0 1 0 ...
##  $ behavAnySale              : int  0 0 1 0 0 0 1 0 1 1 ...
##  $ behavCart                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ behavConversion           : int  0 0 0 0 0 0 0 0 0 0 ...
```

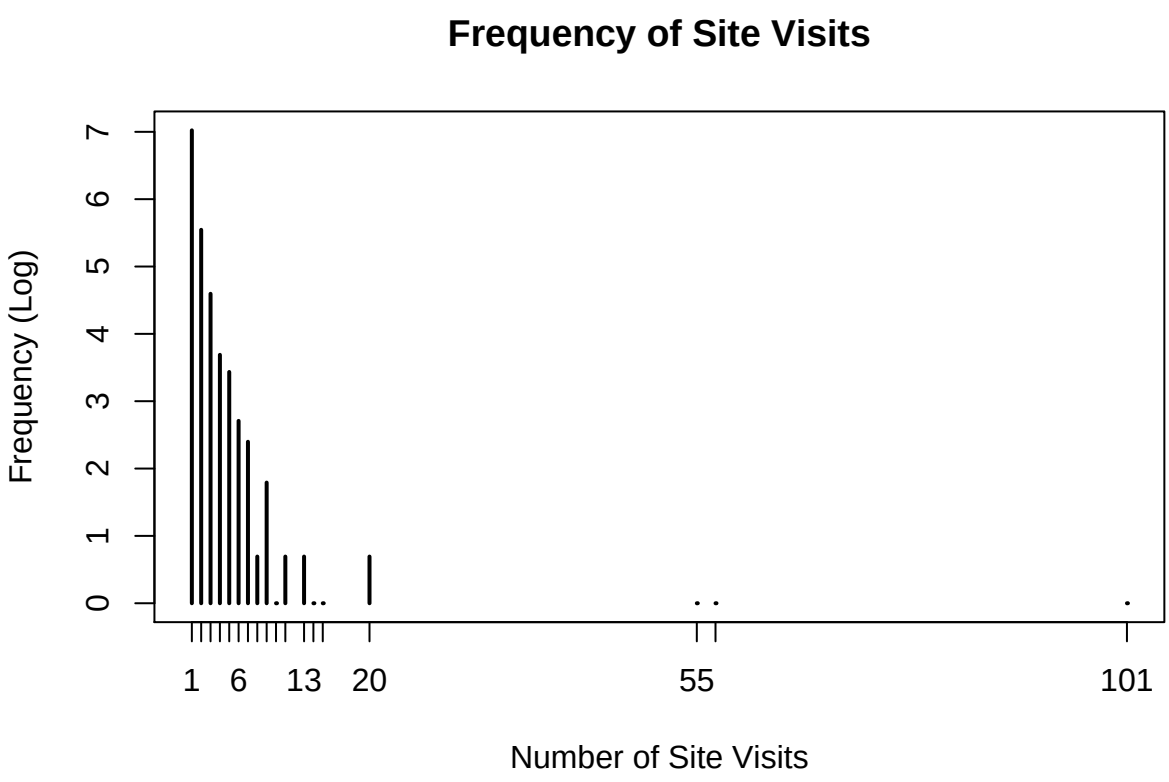
```r
#1a plotting a histogram of behavNumVisits
hist(ecomm.df$behavNumVisits)
```



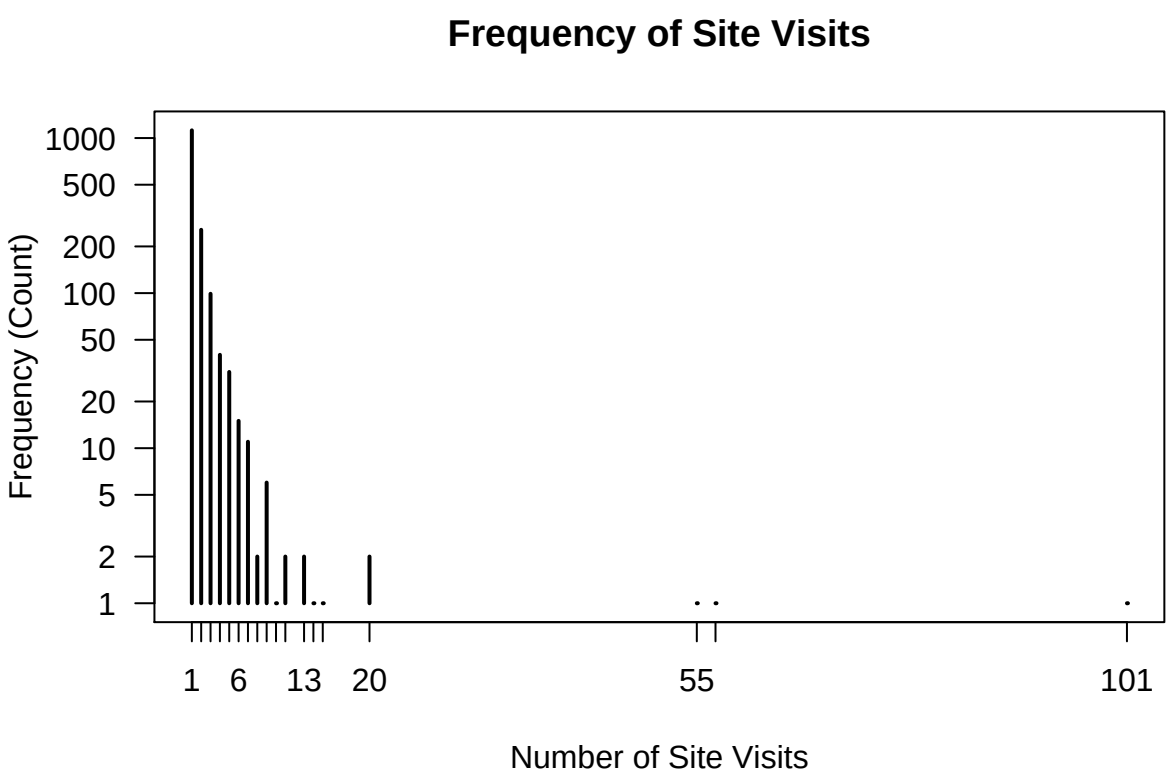

```r
#1b plotting a table of frquencies
plot(table(ecomm.df$behavNumVisits))
```




```
#2 adjusting plot
plot(log(table(ecomm.df$behavNumVisits)),main ="Frequency of Site Visits",xlab = "Number of Site Visits
```

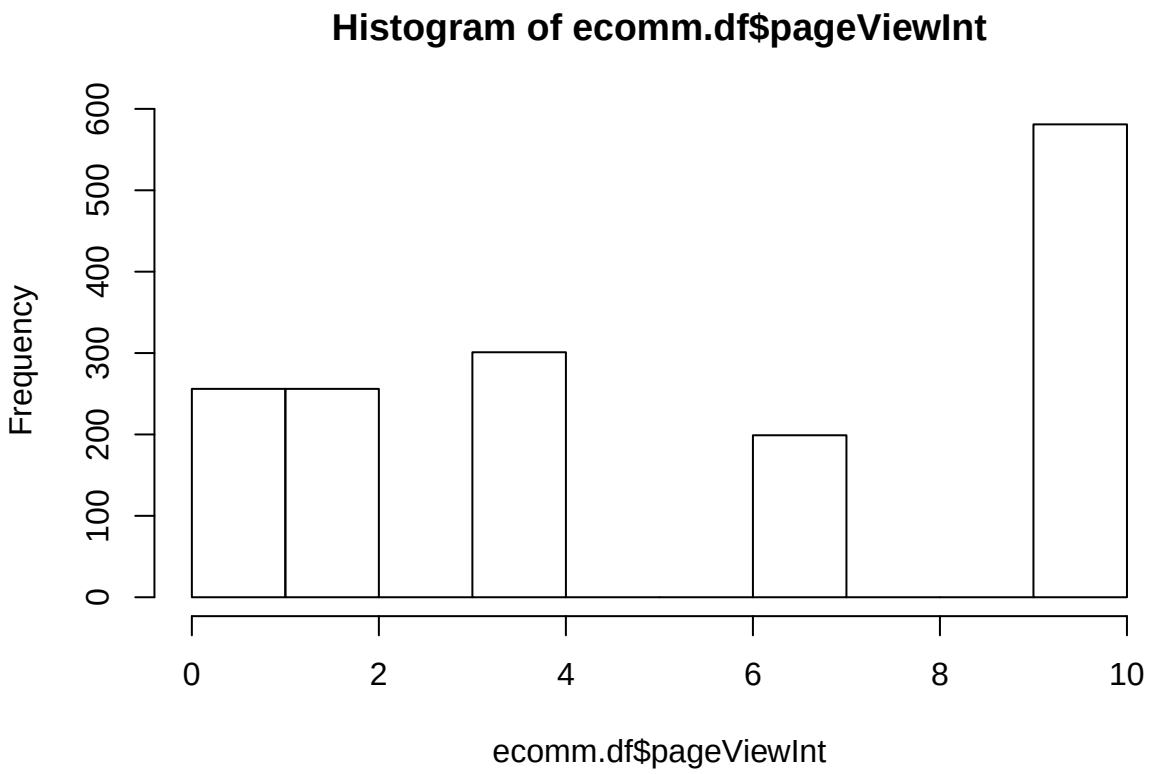



```r
#3 removing and replacing Y Lables
plot(log(table(ecomm.df$behavNumVisits)),
     main ="Frequency of Site Visits",
     xlab = "Number of Site Visits",
     ylab = "Frequency (Count)",
     yaxt="n")
logbreaks <- c(1, 2, 5, 10, 20, 50, 100, 200, 500, 1000)
axis(side=2, at=log(logbreaks), labels=logbreaks, las=1)
```

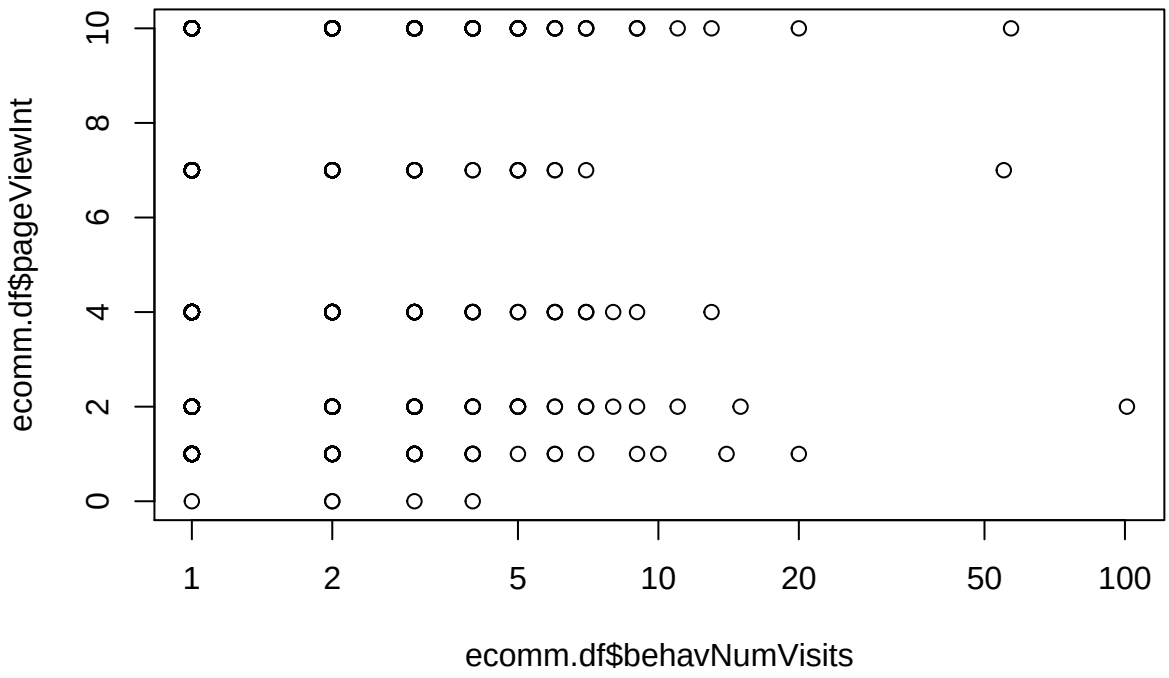



```
#4 creating a new interger variable
pageViewInt <- rep(NA, length(ecomm.df$behavPageviews))
pageViewInt[ecomm.df$behavPageviews=="0"]      <- 0
pageViewInt[ecomm.df$behavPageviews=="1"]      <- 1
pageViewInt[ecomm.df$behavPageviews=="2 to 3"] <- 2
pageViewInt[ecomm.df$behavPageviews=="4 to 6"] <- 4
pageViewInt[ecomm.df$behavPageviews=="7 to 9"] <- 7
pageViewInt[ecomm.df$behavPageviews=="10+"]    <- 10
ecomm.df$pageViewInt <- pageViewInt
rm(pageViewInt)
```

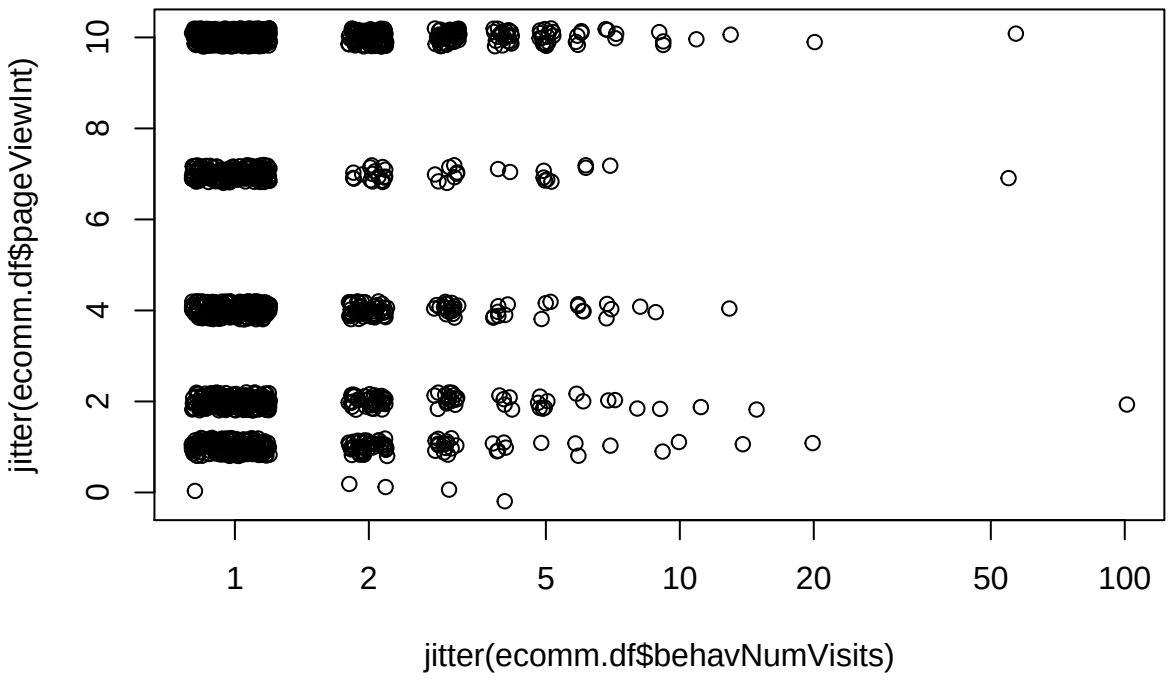
```
#5 plotting new histogram of PageViewInt
hist(ecomm.df$pageViewInt)
```




```
#6 scatterplot of integer estimate of page views
plot(ecomm.df$behavNumVisits, ecomm.df$pageViewInt, log="x")
```




```
#7 Jitter visualization of X and Y values
plot(jitter(ecomm.df$behavNumVisits), jitter(ecomm.df$pageViewInt), log="x")
```




```
#8 What is the Pearson's r correlation coefficient between number of visits and the integer estimate of
cor(ecomm.df$behavNumVisits, ecomm.df$pageViewInt)
```

```
## [1] 0.005626593
```

```
#Answer8: The correlation between two variables is 0.005626593, which is a weak positive relation.
# finding correlation of log of visits
cor(log(ecomm.df$behavNumVisits), ecomm.df$pageViewInt)
```

```
## [1] 0.04003549
```

```
#9 is the correlation from the previous test significant ?
cor.test(ecomm.df$behavNumVisits, ecomm.df$pageViewInt)
```

```
##
##  Pearson's product-moment correlation
##
## data:  ecomm.df$behavNumVisits and ecomm.df$pageViewInt
## t = 0.22443, df = 1591, p-value = 0.8224
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.04349882  0.05472487
## sample estimates:
##         cor
## 0.005626593
```

```r
#Answer9 a: The p-value for the correlation between behaveNumVisits and pageViewInt is greater than 0.0
cor.test(log(ecomm.df$behavNumVisits), ecomm.df$pageViewInt)
```

```
##
##  Pearson's product-moment correlation
##
## data:  log(ecomm.df$behavNumVisits) and ecomm.df$pageViewInt
## t = 1.5982, df = 1591, p-value = 0.1102
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.009095793  0.088973938
## sample estimates:
##        cor
## 0.04003549
```

```r
#Answer9 b: The p-value for the correlation between log(behaveNumVisits) and pageViewInt is greater tha
```

```r
#10 installing "Car" package and using the package
# install the package by uncommenting the below line
#install.packages("car")
library(car)
```

```
## Loading required package: carData
```

```r
# Loading Salaries data from "car" package
data(Salaries)

# getting to know the type of data and variables in dataframe
str(Salaries)
```

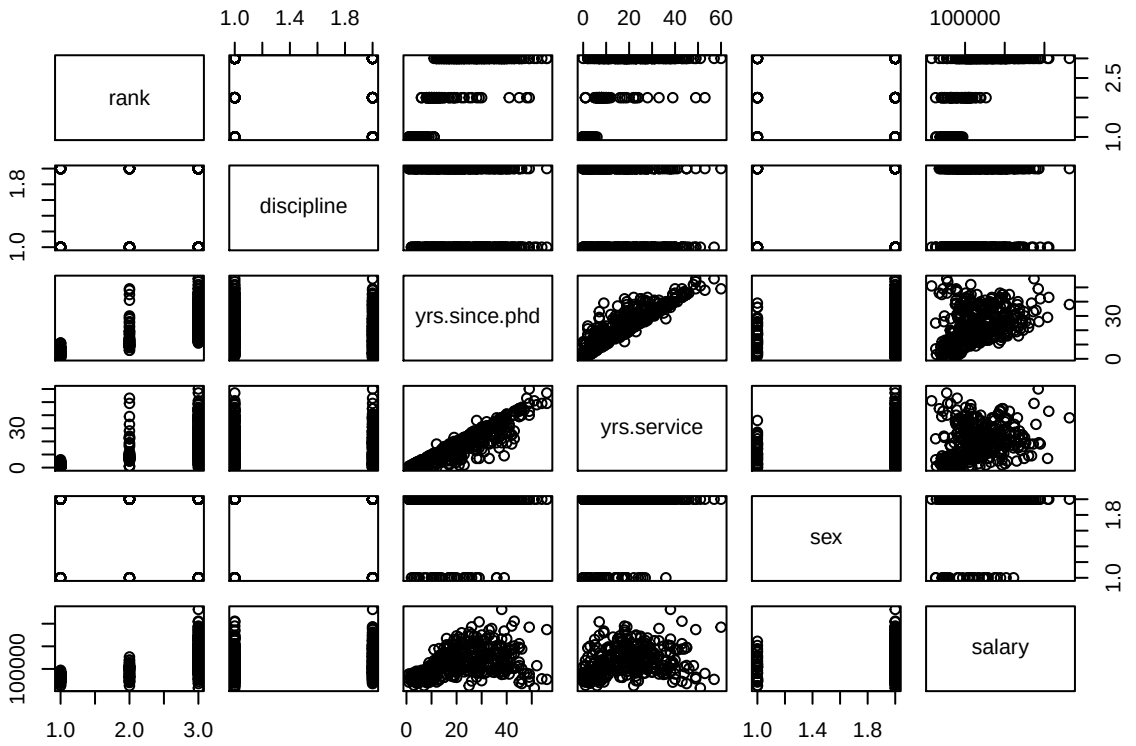
```
## 'data.frame':    397 obs. of  6 variables:
##  $ rank         : Factor w/ 3 levels "AsstProf","AssocProf",..: 3 3 1 3 3 2 3 3 3 3 ...
##  $ discipline   : Factor w/ 2 levels "A","B": 2 2 2 2 2 2 2 2 2 2 ...
##  $ yrs.since.phd: int  19 20 4 45 40 6 30 45 21 18 ...
##  $ yrs.service  : int  18 16 3 39 41 6 23 45 20 18 ...
##  $ sex          : Factor w/ 2 levels "Female","Male": 2 2 2 2 2 2 2 2 2 1 ...
##  $ salary       : int  139750 173200 79750 115000 141500 97000 175000 147765 119250 129000 ...
```

```r
# Getting descriptive statistics for the data
summary(Salaries)
```

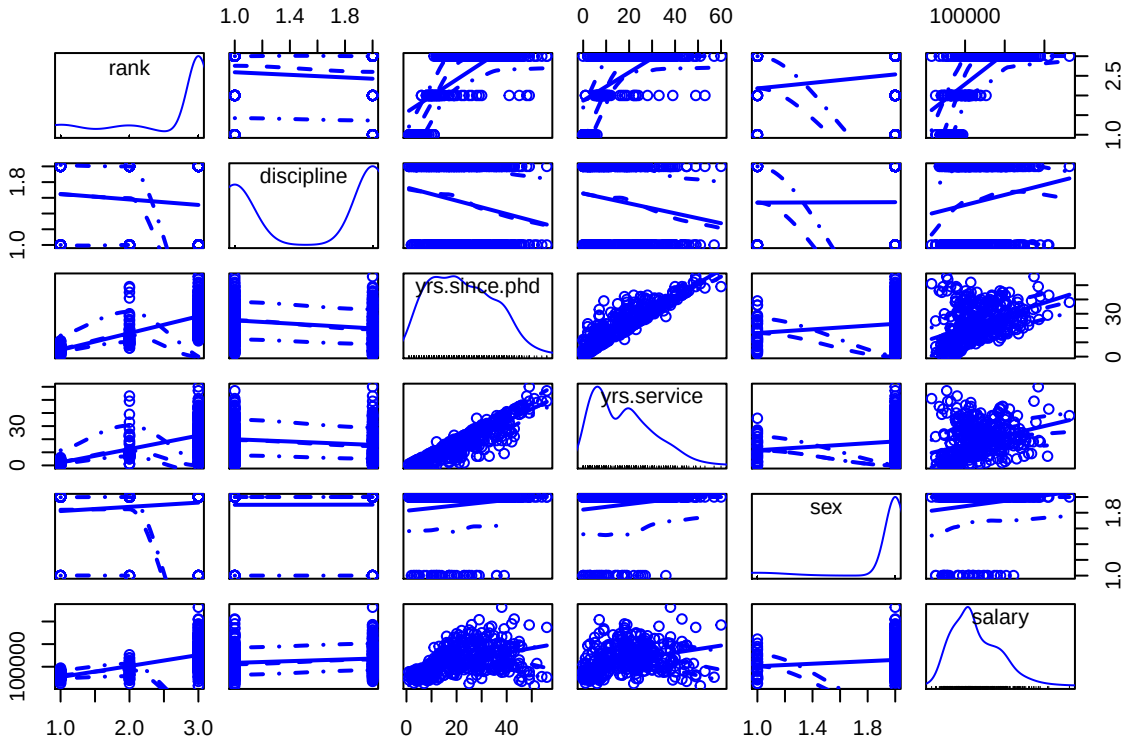
```
##         rank     discipline yrs.since.phd    yrs.service        sex
##  AsstProf : 67   A:181      Min.   : 1.00   Min.   : 0.00   Female: 39
##  AssocProf: 64   B:216      1st Qu.:12.00   1st Qu.: 7.00   Male  :358
##  Prof     :266              Median :21.00   Median :16.00
##                             Mean   :22.31   Mean   :17.61
##                             3rd Qu.:32.00   3rd Qu.:27.00
##                             Max.   :56.00   Max.   :60.00
##      salary
##  Min.   : 57800
##  1st Qu.: 91000
```

```
##  Median :107300
##  Mean   :113706
##  3rd Qu.:134185
##  Max.   :231545
```

```
#11 The two functions for scatterplot is: Matrix and pairs
pairs(formula = ~ rank+discipline+yrs.since.phd+yrs.service+sex+salary, data = Salaries)
```




```
scatterplotMatrix(formula = ~ rank+discipline+yrs.since.phd+yrs.service+sex+salary, data = Salaries)
```



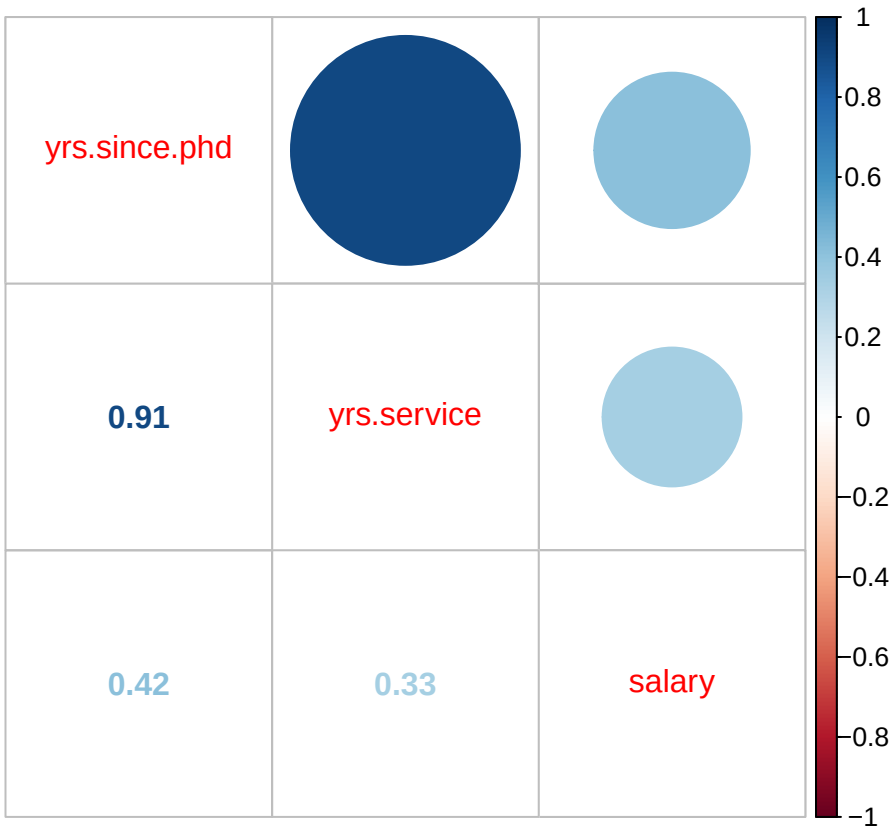

```
#The scatterplotMatrix() function adds a number of features over pairs(), including adding smoothed lin
```

```
#12.Numeric variables in the Salaries data set?

# install the package by uncommenting the below line
#install.packages("corrplot")
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
corrplot.mixed(corr=cor(Salaries[,c(3,4,6)], use="complete.obs"))
```




```
#Answer12a: Variable yrs.since.phd, yrs.service and Salary are variable in Salaries data set
```

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Cmd+Option+I*.

When you save the notebook, an HTML file containing the code and output will be saved alongside it (click the *Preview* button or press *Cmd+Shift+K* to preview the HTML file).

The preview shows you a rendered HTML copy of the contents of the editor. Consequently, unlike *Knit*, *Preview* does not run any R code chunks. Instead, the output of the chunk when it was last run in the editor is displayed.