> An Introduction to **R**

[1] Tyler K. Perrachione
[4] Gabrieli Lab, MIT

[7] 28 June 2010

[10] tkp@mit.edu

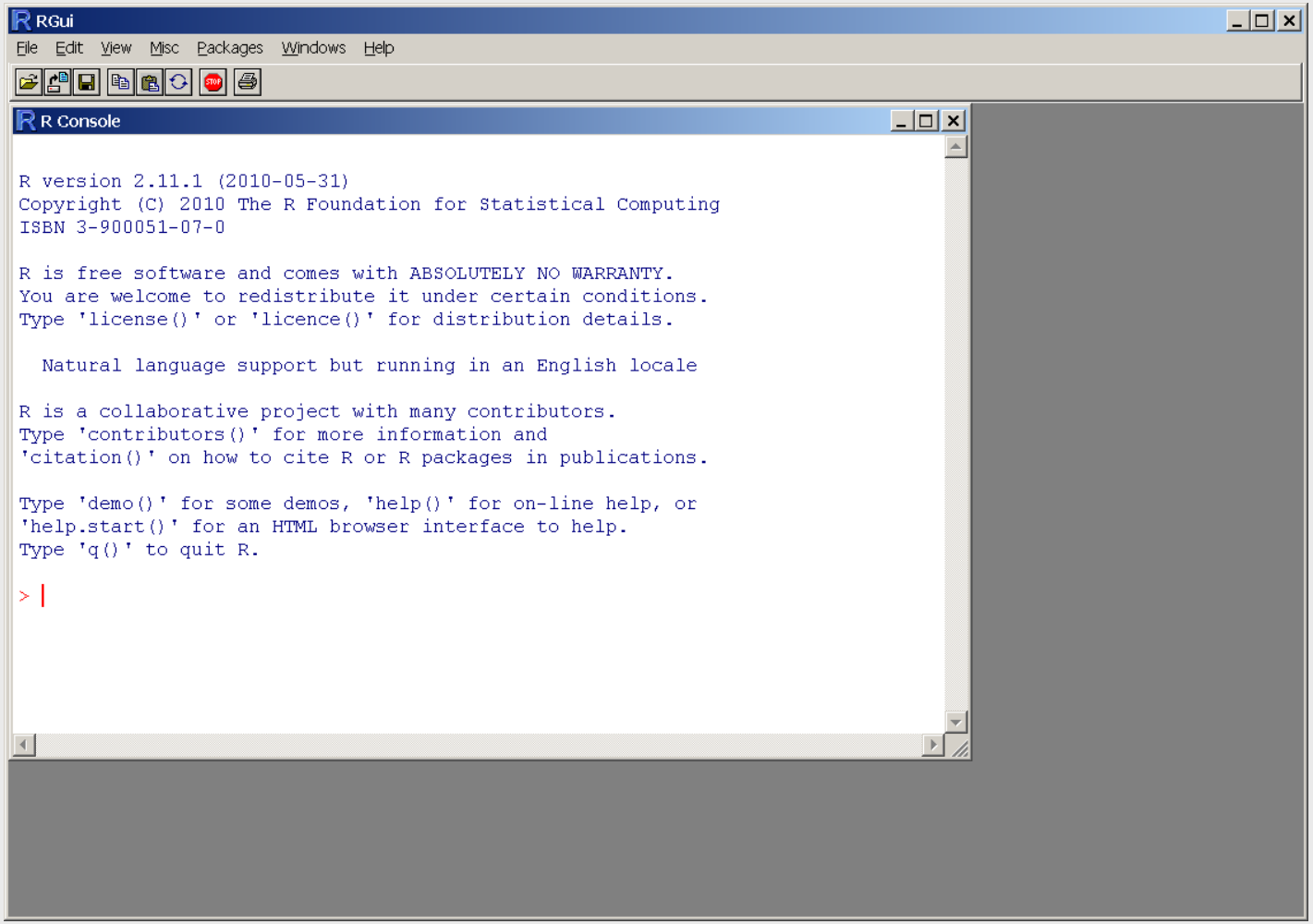
What is R?

- A suite of operators for calculations on arrays, in particular matrices,
- A large, coherent, integrated collection of intermediate tools for data analysis,
- Graphical facilities for data analysis and display either on-screen or on hardcopy, and
- A well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.
- Free (as in beer *and* speech), open-source software

Installing, Running, and Interacting with R

- How to get R:
  - http://www.r-project.org/
  - Google: “R”
  - Windows, Linux, Mac OS X, source
  - On mindhive:
    - `user@ba1:~$> R` [terminal only]
    - `user@ba1:~$> R -g Tk &` [application window]
- Files for this tutorial:
  - http://web.mit.edu/tkp/www/R/R_Tutorial_Data.txt
  - http://web.mit.edu/tkp/www/R/R_Tutorial_Inputs.txt

Installing, Running, and Interacting with R



Installing, Running, and Interacting with R

- All examples are in file “R_Tutorial_Inputs.txt”
- Entering data
  - Math
  - Variables
  - Arrays
  - Math on arrays
  - Functions
- Getting help
- Reading data from files
- Selecting subsets of data

Installing, Running, and Interacting with R

Math:

```
> 1 + 1
[1] 2

> 1 + 1 * 7
[1] 8

> (1 + 1) * 7
[1] 14
```

Variables:

```
> x <- 1
> x
[1] 1
> y = 2
> y
[1] 2
> 3 -> z
> z
[1] 3
> (x + y) * z
[1] 9
```

Installing, Running, and Interacting with R

Arrays:

```
> x <- c(0,1,2,3,4)
> x
[1] 0 1 2 3 4

> y <- 1:5
> y
[1] 1 2 3 4 5

> z <- 1:50
> z
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
[16] 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
[31] 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
[46] 46 47 48 49 50
```

Installing, Running, and Interacting with R

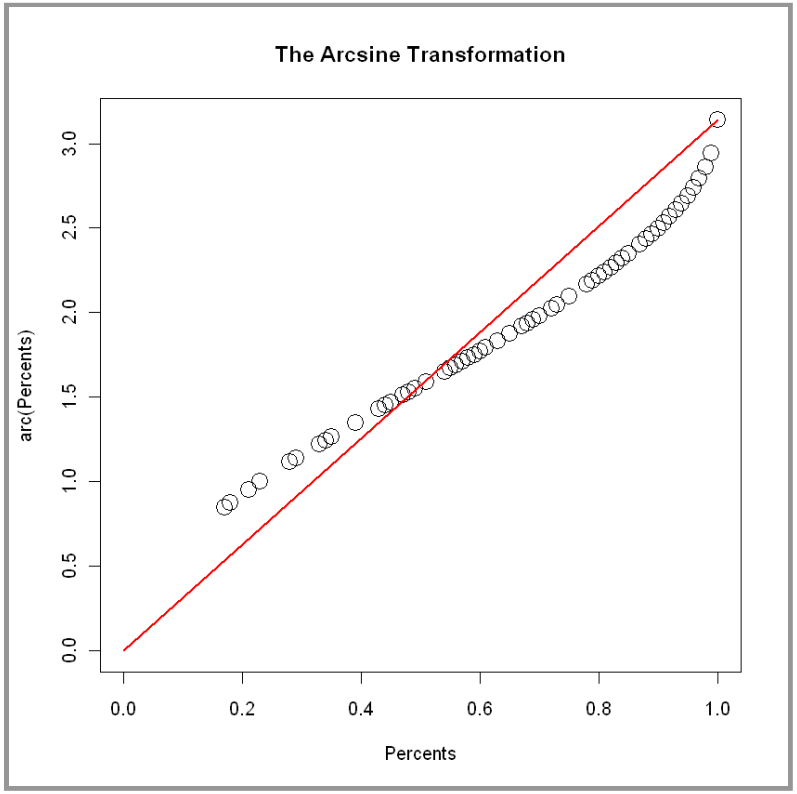
Math on arrays:

```
> x <- c(0,1,2,3,4)
> y <- 1:5
> z <- 1:50
> x + y
[1] 1 3 5 7 9
> x * y
[1]  0  2  6 12 20
> x * z
 [1]   0   2   6  12  20   0   7  16  27  40   0
[12]  12  26  42  60   0  17  36  57  80   0  22
[23]  46  72 100   0  27  56  87 120   0  32  66
[34] 102 140   0  37  76 117 160   0  42  86 132
[45] 180   0  47  96 147 200
```

Installing, Running, and Interacting with R

Functions:

```
> arc <- function(x) 2*asin(sqrt(x))
> arc(0.5)
[1] 1.570796
> x <- c(0,1,2,3,4)
> x <- x / 10
> arc(x)
[1] 0.0000000 0.6435011 0.9272952
[4] 1.1592795 1.3694384
```

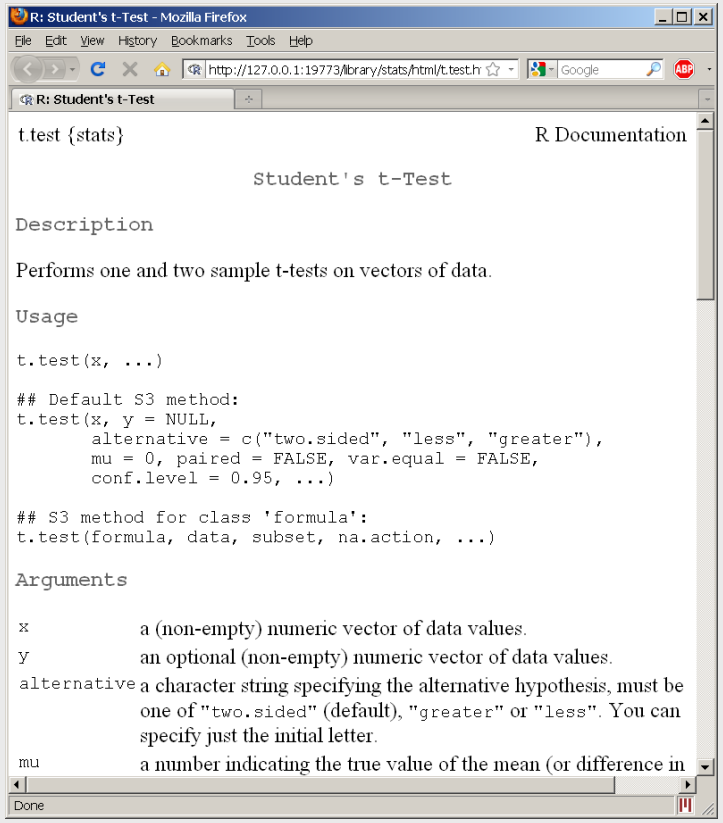


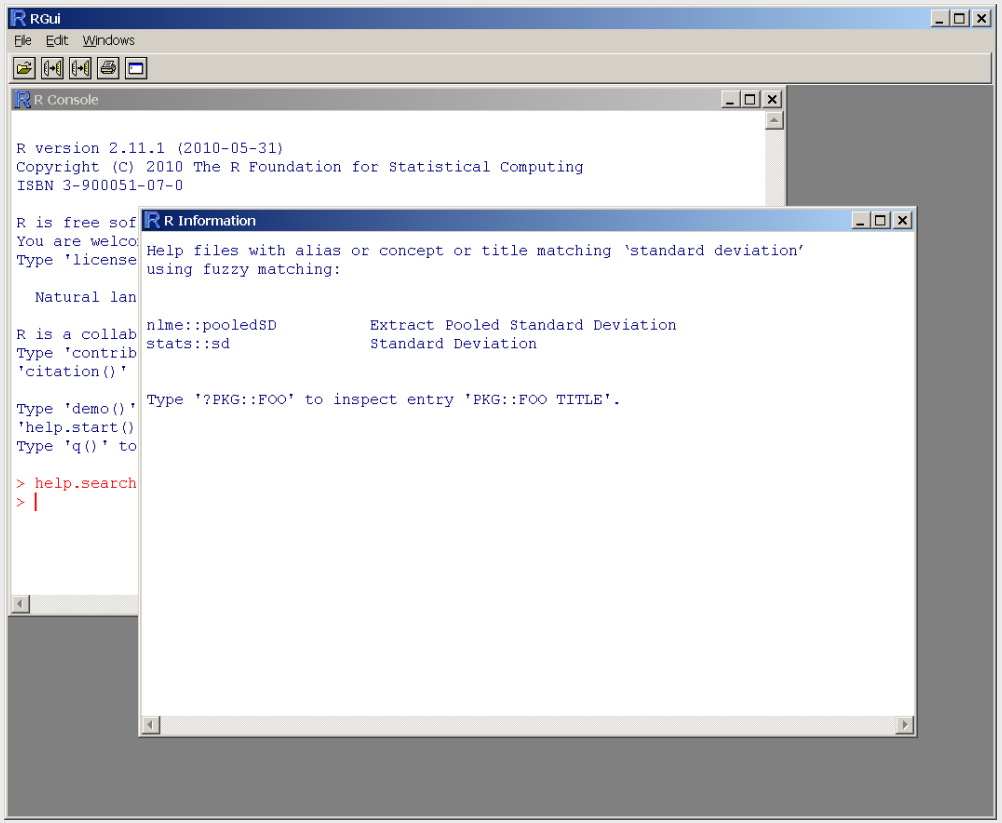

```
> plot(arc(Percents)~Percents,
+ pch=21,cex=2,xlim=c(0,1),ylim=c(0,pi),
+ main="The Arcsine Transformation")
> lines(c(0,1),c(0,pi),col="red",lwd=2)
```

Installing, Running, and Interacting with R

Getting help:

```
> help(t.test)
> help.search("standard deviation")
```





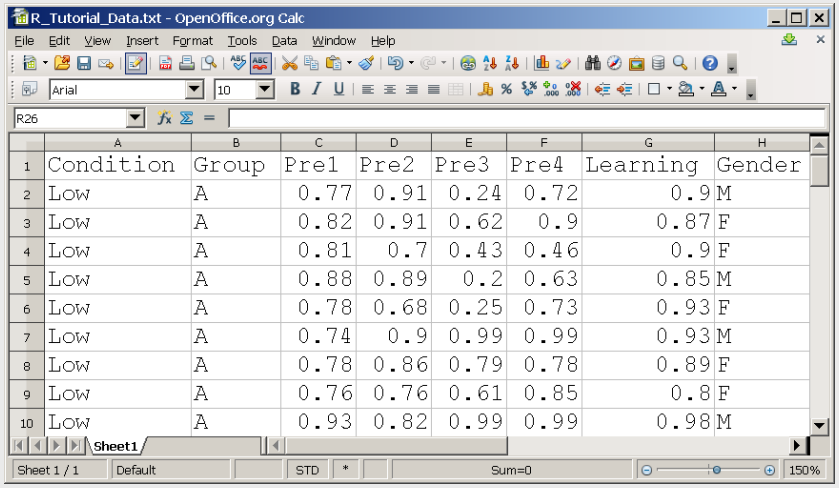
Installing, Running, and Interacting with R

- Example experiment:
  - Subjects learning to perform a new task:
  - Two groups of subjects
    - (“A” and “B”; high and low aptitude learners)
  - Two types of training paradigm
    - (“High variability” and “Low variability”)
  - Four pre-training assessment tests
- Example data in “R_Tutorial_Data.txt”

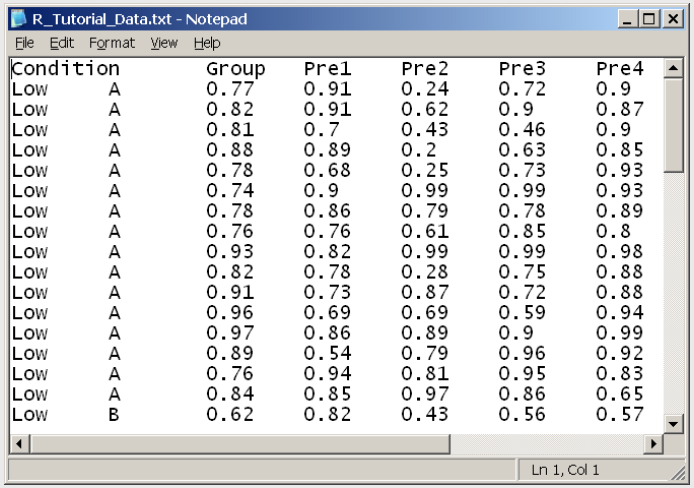
Installing, Running, and Interacting with R

Reading data from files:

```
> myData <- read.table("R_Tutorial_Data.txt",
+ header=TRUE, sep="\t")
> myData
   Condition Group Pre1 Pre2 Pre3 Pre4 Learning
1        Low     A 0.77 0.91 0.24 0.72     0.90
2        Low     A 0.82 0.91 0.62 0.90     0.87
3        Low     A 0.81 0.70 0.43 0.46     0.90
. . .
61      High     B 0.44 0.41 0.84 0.82     0.29
62      High     B 0.48 0.56 0.83 0.85     0.48
63      High     B 0.61 0.82 0.88 0.95     0.28
```



	A	B	C	D	E	F	G	H
1	Condition	Group	Pre1	Pre2	Pre3	Pre4	Learning	Gender
2	Low	A	0.77	0.91	0.24	0.72	0.9	M
3	Low	A	0.82	0.91	0.62	0.9	0.87	F
4	Low	A	0.81	0.7	0.43	0.46	0.9	F
5	Low	A	0.88	0.89	0.2	0.63	0.85	M
6	Low	A	0.78	0.68	0.25	0.73	0.93	F
7	Low	A	0.74	0.9	0.99	0.99	0.93	M
8	Low	A	0.78	0.86	0.79	0.78	0.89	F
9	Low	A	0.76	0.76	0.61	0.85	0.8	F
10	Low	A	0.93	0.82	0.99	0.99	0.98	M

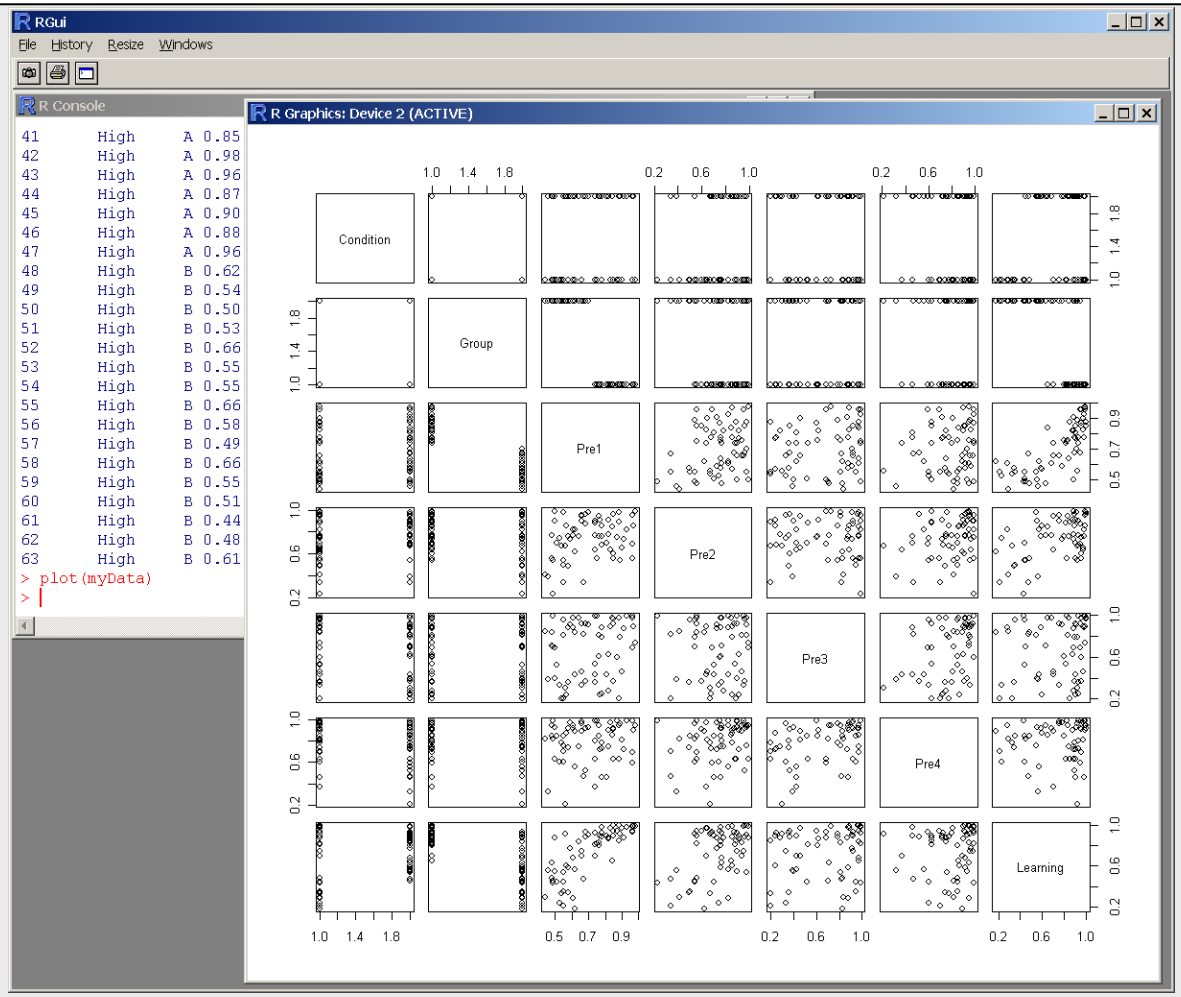


Condition	Group	Pre1	Pre2	Pre3	Pre4
Low	A	0.77	0.91	0.24	0.72	0.9
Low	A	0.82	0.91	0.62	0.9	0.87
Low	A	0.81	0.7	0.43	0.46	0.9
Low	A	0.88	0.89	0.2	0.63	0.85
Low	A	0.78	0.68	0.25	0.73	0.93
Low	A	0.74	0.9	0.99	0.99	0.93
Low	A	0.78	0.86	0.79	0.78	0.89
Low	A	0.76	0.76	0.61	0.85	0.8
Low	A	0.93	0.82	0.99	0.99	0.98
Low	A	0.82	0.78	0.28	0.75	0.88
Low	A	0.91	0.73	0.87	0.72	0.88
Low	A	0.96	0.69	0.69	0.59	0.94
Low	A	0.97	0.86	0.89	0.9	0.99
Low	A	0.89	0.54	0.79	0.96	0.92
Low	A	0.76	0.94	0.81	0.95	0.83
Low	A	0.84	0.85	0.97	0.86	0.65
Low	B	0.62	0.82	0.43	0.56	0.57

Installing, Running, and Interacting with R

Examining datasets:

```
> plot(myData)
```



Installing, Running, and Interacting with R

Selecting subsets of data:

```
> myData$Learning
 [1] 0.90 0.87 0.90 0.85 0.93 0.93 0.89 0.80 0.98
[10] 0.88 0.88 0.94 0.99 0.92 0.83 0.65 0.57 0.55
[19] 0.94 0.68 0.89 0.60 0.63 0.84 0.92 0.56 0.78
[28] 0.54 0.47 0.45 0.59 0.91 0.98 0.82 0.93 0.81
[37] 0.97 0.95 0.70 1.00 0.90 0.99 0.95 0.95 0.97
[46] 1.00 0.99 0.18 0.33 0.88 0.23 0.75 0.21 0.35
[55] 0.70 0.34 0.43 0.75 0.44 0.44 0.29 0.48 0.28
> myData$Learning[myData$Group=="A"]
 [1] 0.90 0.87 0.90 0.85 0.93 0.93 0.89 0.80 0.98
[10] 0.88 0.88 0.94 0.99 0.92 0.83 0.65 0.98 0.82
[19] 0.93 0.81 0.97 0.95 0.70 1.00 0.90 0.99 0.95
[28] 0.95 0.97 1.00 0.99
```

Installing, Running, and Interacting with R

Selecting subsets of data:

```
> myData$Learning
 [1] 0.90 0.87 0.90 0.85 0.93 0.93 0.89 0.80 0.98
[10] 0.88 0.88 0.94 0.99 0.92 0.83 0.65 0.57 0.55
[19] 0.94 0.68 0.89 0.60 0.63 0.84 0.92 0.56 0.78
[28] 0.54 0.47 0.45 0.59 0.91 0.98 0.82 0.93 0.81
[37] 0.97 0.95 0.70 1.00 0.90 0.99 0.95 0.95 0.97
[46] 1.00 0.99 0.18 0.33 0.88 0.23 0.75 0.21 0.35
[55] 0.70 0.34 0.43 0.75 0.44 0.44 0.29 0.48 0.28
> attach(myData)
> Learning
 [1] 0.90 0.87 0.90 0.85 0.93 0.93 0.89 0.80 0.98
[10] 0.88 0.88 0.94 0.99 0.92 0.83 0.65 0.57 0.55
[19] 0.94 0.68 0.89 0.60 0.63 0.84 0.92 0.56 0.78
[28] 0.54 0.47 0.45 0.59 0.91 0.98 0.82 0.93 0.81
[37] 0.97 0.95 0.70 1.00 0.90 0.99 0.95 0.95 0.97
[46] 1.00 0.99 0.18 0.33 0.88 0.23 0.75 0.21 0.35
[55] 0.70 0.34 0.43 0.75 0.44 0.44 0.29 0.48 0.28
```

Installing, Running, and Interacting with R

Selecting subsets of data:

```
> Learning[Group=="A"]
 [1] 0.90 0.87 0.90 0.85 0.93 0.93 0.89 0.80 0.98
[10] 0.88 0.88 0.94 0.99 0.92 0.83 0.65 0.98 0.82
[19] 0.93 0.81 0.97 0.95 0.70 1.00 0.90 0.99 0.95
[28] 0.95 0.97 1.00 0.99
> Learning[Group!="A"]
 [1] 0.57 0.55 0.94 0.68 0.89 0.60 0.63 0.84 0.92
[10] 0.56 0.78 0.54 0.47 0.45 0.59 0.91 0.18 0.33
[19] 0.88 0.23 0.75 0.21 0.35 0.70 0.34 0.43 0.75
[28] 0.44 0.44 0.29 0.48 0.28
> Condition[Group=="B"&Learning<0.5]
 [1] Low  Low  High High High High High High High
[10] High High High High High
Levels: High Low
```

Statistics and Data Analysis

- Parametric Tests
  - Independent sample t-tests
  - Paired sample t-tests
  - One sample t-tests
  - Correlation
- Nonparametric tests
  - Shapiro-Wilks test for normality
  - Wilcoxon signed-rank test (Mann-Whitney U)
  - Chi square test
- Linear Models and ANOVA

Basic parametric inferential statistics

Independent sample t-tests:

```
> t.test(Pre2[Group=="A"],
+ Pre2[Group=="B"],
+ paired=FALSE)

        Welch Two Sample t-test

data:  Learning[Group == "A"] and Learning[Group == "B"]
t = 1.6117, df = 53.275, p-value = 0.1129
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.0179193  0.1645725
sample estimates:
mean of x mean of y
0.7764516 0.7031250
```

Basic parametric inferential statistics

Independent sample t-tests:

```
> t.test(Pre2[Group=="A"],
+ Pre2[Group=="B"],
+ paired=FALSE,
+ var.equal=TRUE)

        Welch Two Sample t-test

data:  Learning[Group == "A"] and Learning[Group == "B"]
t = 1.601, df = 61, p-value = 0.1145
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.0179193  0.1645725
sample estimates:
mean of x mean of y
0.7764516 0.7031250
```

Basic parametric inferential statistics

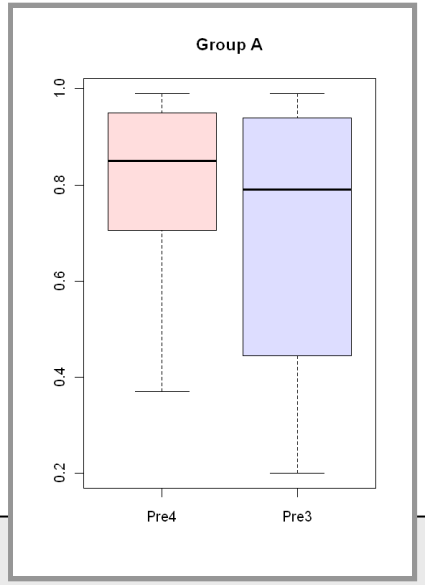
Independent sample t-tests:

```
> t.test(Pre2[Group=="A"],
+ Pre2[Group=="B"],
+ paired=FALSE,
+ var.equal=TRUE,
+ alternative="greater")

        Welch Two Sample t-test

data:  Learning[Group == "A"] and Learning[Group == "B"]
t = 1.601, df = 61, p-value = 0.5727
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 -0.003169388          Inf
sample estimates:
mean of x mean of y
0.7764516 0.7031250
```

Basic parametric inferential statistics

Paired sample t-test:

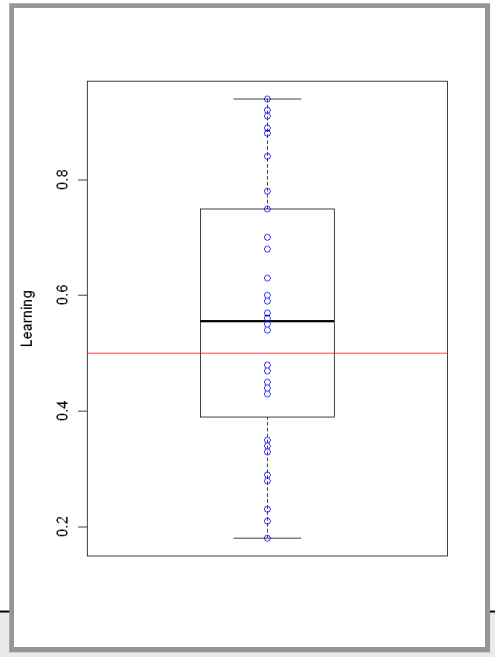
```
> t.test(Pre4[Group=="A"],
+ Pre3[Group=="A"],
+ paired=TRUE)

        Paired t-test

data:  Pre4[Group == "A"] and Pre3[Group == "A"]
t = 2.4054, df = 30, p-value = 0.02253
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.01641059 0.20100876
sample estimates:
mean of the differences
              0.1087097

> boxplot(Pre4[Group=="A"],
+ Pre3[Group=="A"],
+ col=c("#ffdddd","#ddddff"),
+ names=c("Pre4","Pre3"),main="Group A")
```



Basic parametric inferential statistics

One sample t-test:

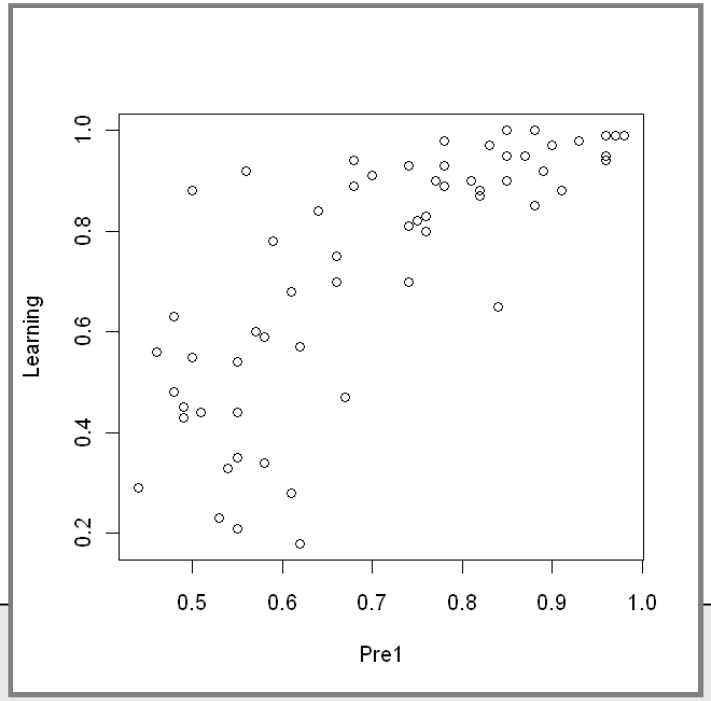
```
> t.test(Learning[Group=="B"], mu=0.5, alternative="greater")

        One Sample t-test

data:  Learning[Group == "B"]
t = 1.5595, df = 31, p-value = 0.06452
alternative hypothesis: true mean is greater than 0.5
95 percent confidence interval:
 0.4945469       Inf
sample estimates:
mean of x
   0.5625

> boxplot(Learning[Group=="B"],
+ names="Group B", ylab="Learning")
> lines(c(0,2), c(0.5, 0.5), col="red")
> points(c(rep(1,length(Learning[Group=="B"]))),
+ Learning[Group=="B"], pch=21, col="blue")
```



Basic parametric inferential statistics

Correlation:

```
> cor.test(Pre1,Learning,method="pearson")

        Pearson's product-moment correlation

data:  Pre1 and Learning
t = 9.2461, df = 61, p-value = 3.275e-13
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6366698 0.8506815
sample estimates:
      cor
0.7639292

> plot(Pre1,Learning)
```

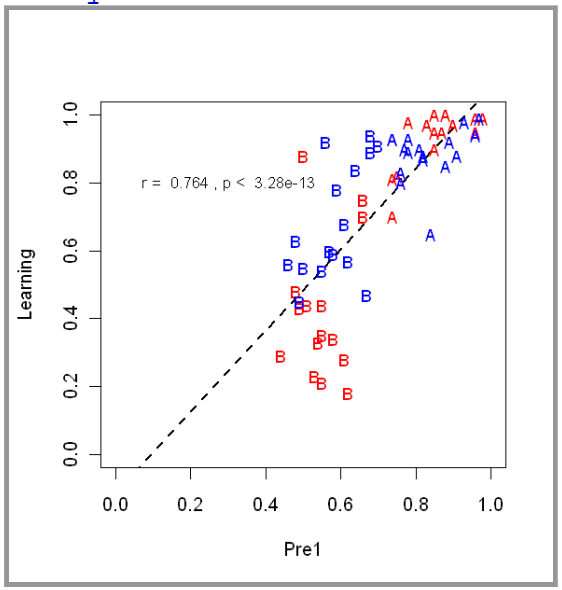


Basic parametric inferential statistics

Correlation (fancier plot example):

```
> cor.test(Pre1,Learning,method="pearson")

        Pearson's product-moment correlation

data:  Pre1 and Learning
t = 9.2461, df = 61, p-value = 3.275e-13
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.6366698 0.8506815
sample estimates:
      cor
0.7639292
```

```
> plot(Learning~Pre1, ylim=c(0,1), xlim=c(0,1), ylab="Learning", xlab="Pre1", type="n")
> abline(lm(Learning~Pre1),col="black",lty=2, lwd=2)
> points(Learning[Group=="A"&Condition=="High"]~Pre1[Group=="A"&Condition=="High"],
+ pch=65, col="red", cex=0.9)
> points(Learning[Group=="A"&Condition=="Low"]~Pre1[Group=="A"&Condition=="Low"],
+ pch=65, col="blue", cex=0.9)
> points(Learning[Group=="B"&Condition=="High"]~Pre1[Group=="B"&Condition=="High"],
+ pch=66, col="red", cex=0.9)
> points(Learning[Group=="B"&Condition=="Low"]~Pre1[Group=="B"&Condition=="Low"],
+ pch=66, col="blue", cex=0.9)
> legend(2.5,1.0, c("LV Training", "HV Training"), pch=c(19), col=c("blue","red"), bty="y")
> yCor <- cor.test(Pre1, Learning, method="pearson")
> text(0.3,0.8, paste("r = ", format(myCor$estimate,digits=3),", p < ", format(myCor$p.value,digits=3)), cex=0.8)
```



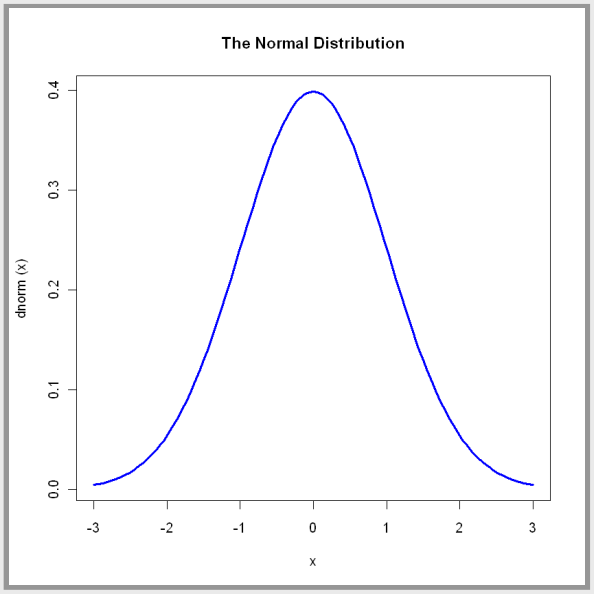
Are my data normally distributed?

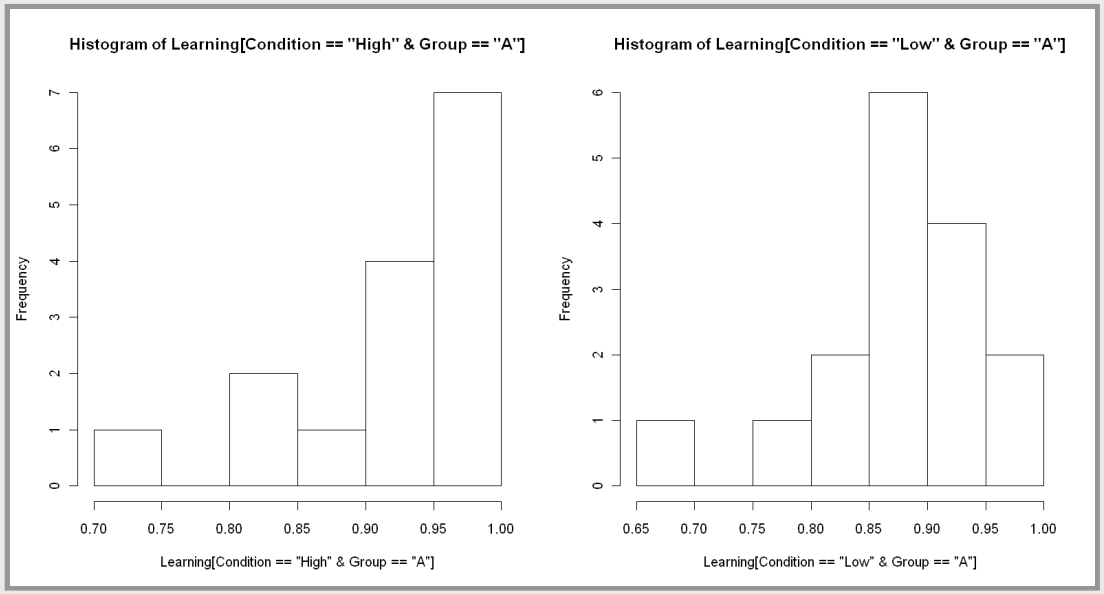
```
> t.test(Learning[Condition=="High"&Group=="A"],
+ Learning[Condition=="Low"&Group=="A"])

        Welch Two Sample t-test

data:  Learning[Condition == "High" & Group == "A"] and
    Learning[Condition == "Low" & Group == "A"]
t = 1.457, df = 28.422, p-value = 0.1561
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.01764821  0.10481488
sample estimates:
mean of x mean of y
0.9273333 0.8837500
```

Statistics and Data Analysis

Are my data normally distributed?

```
> plot(dnorm,-3,3,col="blue",lwd=3,main="The Normal Distribution")
> par(mfrow=c(1,2))
> hist(Learning[Condition=="High"&Group=="A"])
> hist(Learning[Condition=="Low"&Group=="A"])
```





Statistics and Data Analysis

Are my data normally distributed?

```
> shapiro.test(Learning[Condition=="High"&Group=="A"])

        Shapiro-Wilk normality test

data:  Learning[Condition == "High" & Group == "A"]
W = 0.7858, p-value = 0.002431

> shapiro.test(Learning[Condition=="Low"&Group=="A"])

        Shapiro-Wilk normality test

data:  Learning[Condition == "Low" & Group == "A"]
W = 0.8689, p-value = 0.02614
```

Basic nonparametric inferential statistics

Wilcoxon signed-rank / Mann-Whitney U tests:

```
> wilcox.test(Learning[Condition=="High"&Group=="A"],
+ Learning[Condition=="Low"&Group=="A"],
+ exact=FALSE,
+ paired=FALSE)

        Wilcoxon rank sum test with continuity correction

data:  Learning[Condition == "High" & Group == "A"] and
    Learning[Condition == "Low" & Group == "A"]
W = 173.5, p-value = 0.03580
alternative hypothesis: true location shift is not equal to 0
```

Basic nonparametric inferential statistics

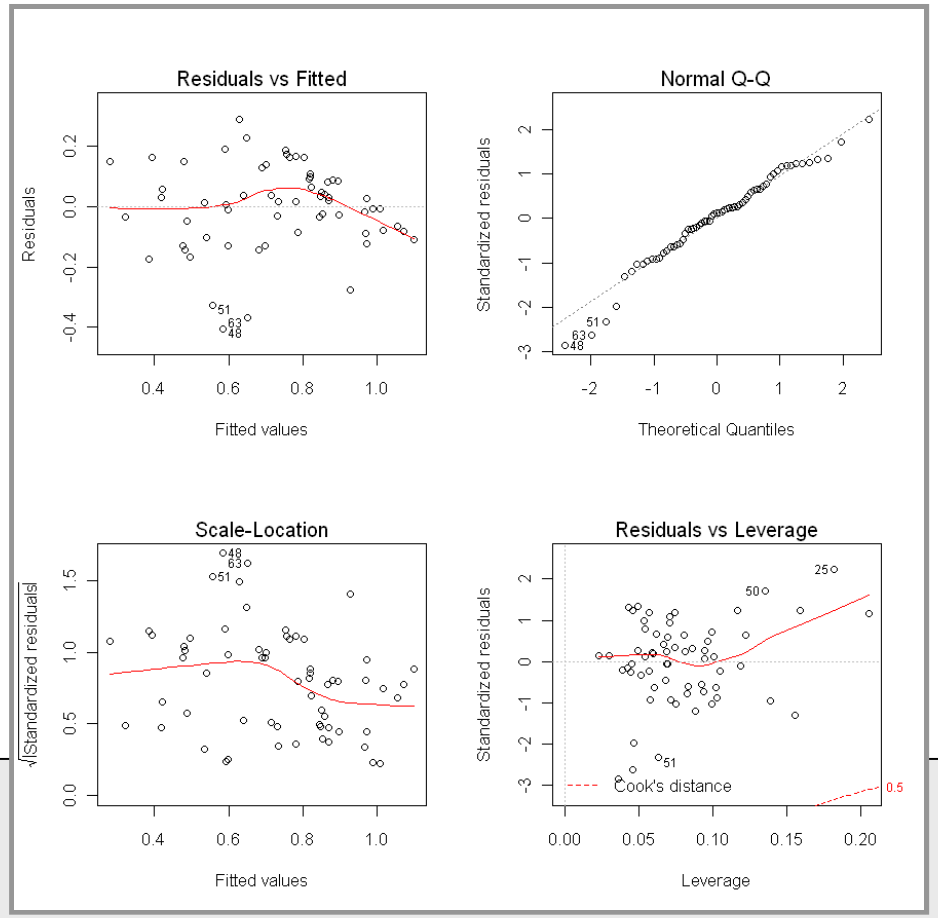
Chi square test:

```
> x <- matrix(c(
+ length(Learning[Group=="A"&Condition=="High"&Gender=="F"]),
+ length(Learning[Group=="A"&Condition=="Low"&Gender=="F"]),
+ length(Learning[Group=="B"&Condition=="High"&Gender=="F"]),
+ length(Learning[Group=="B"&Condition=="Low"&Gender=="F"])),
+ ncol=2)
> x
     [,1] [,2]
[1,]    4   12
[2,]   10    7
> chisq.test(x)

        Pearson's Chi-squared test with Yates' continuity correction

data:  x
X-squared = 2.5999, df = 1, p-value = 0.1069
```

Linear models and ANOVA

Linear models:

```
> myModel <- lm(Learning ~ Pre1 + Pre2 + Pre3 + Pre4)
> par(mfrow=c(2,2))
> plot(myModel)
```



Linear models and ANOVA

Linear models:

```
> summary(myModel)

Call:
lm(formula = Learning ~ Pre1 + Pre2 + Pre3 + Pre4)

Residuals:
     Min       1Q   Median       3Q      Max
-0.40518 -0.08460  0.01707  0.09170  0.29074

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.22037    0.11536  -1.910 0.061055 .
Pre1         1.05299    0.12636   8.333 1.70e-11 ***
Pre2         0.41298    0.10926   3.780 0.000373 ***
Pre3         0.07339    0.07653   0.959 0.341541
Pre4        -0.18457    0.11318  -1.631 0.108369
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1447 on 58 degrees of freedom
Multiple R-squared: 0.6677,     Adjusted R-squared: 0.6448
F-statistic: 29.14 on 4 and 58 DF,  p-value: 2.710e-13
```

Linear models and ANOVA

Linear models:

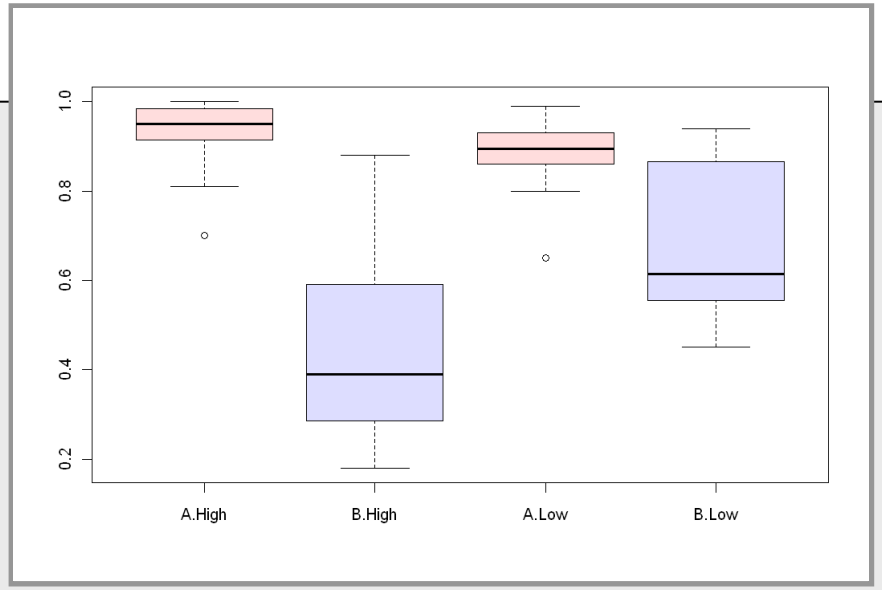
```
> step(myModel, direction="backward")
Start:  AIC=-238.8
Learning ~ Pre1 + Pre2 + Pre3 + Pre4

       Df Sum of Sq    RSS     AIC
- Pre3  1   0.01925 1.2332 -239.81
<none>              1.2140 -238.80
- Pre4  1   0.05566 1.2696 -237.98
- Pre2  1   0.29902 1.5130 -226.93
- Pre1  1   1.45347 2.6675 -191.21

Step:  AIC=-239.81
Learning ~ Pre1 + Pre2 + Pre4

       Df Sum of Sq    RSS     AIC
- Pre4  1   0.03810 1.2713 -239.89
<none>              1.2332 -239.81
- Pre2  1   0.28225 1.5155 -228.83
- Pre1  1   1.54780 2.7810 -190.58

                ...
```

```
                ...

Step:  AIC=-239.89
Learning ~ Pre1 + Pre2

       Df Sum of Sq    RSS     AIC
<none>              1.2713 -239.89
- Pre2  1   0.24997 1.5213 -230.59
- Pre1  1   1.52516 2.7965 -192.23

Call:
lm(formula = Learning ~ Pre1 + Pre2)

Coefficients:
(Intercept)         Pre1         Pre2

    -0.2864       1.0629       0.3627
```

Linear models and ANOVA

ANOVA:

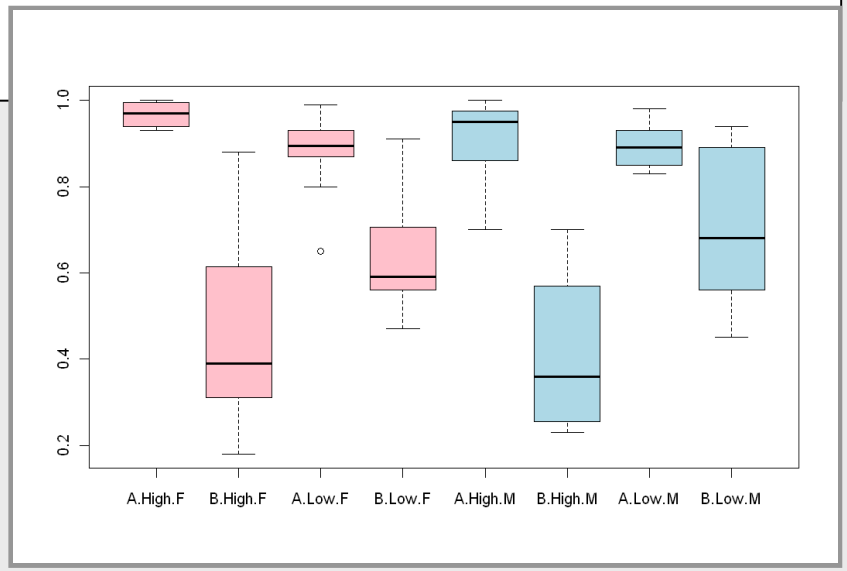
```
> myANOVA <- aov(Learning~Group*Condition)
> summary(myANOVA)
                Df Sum Sq Mean Sq F value    Pr(>F)
Group            1 1.8454 1.84537 81.7106 9.822e-13 ***
Condition        1 0.1591 0.15910  7.0448 0.0102017 *
Group:Condition  1 0.3164 0.31640 14.0100 0.0004144 ***
Residuals       59 1.3325 0.02258
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> boxplot(Learning~Group*Condition,col=c("#ffdddd","#ddddff"))
```



Linear models and ANOVA

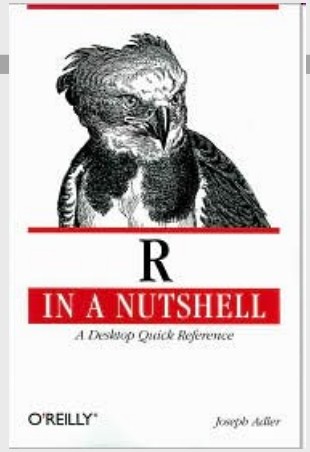
ANOVA:

```
> myANOVA2 <- aov(Learning~Group*Condition+Gender)
> summary(myANOVA2)
                 Df  Sum Sq Mean Sq F value    Pr(>F)
Group             1 1.84537 1.84537 80.3440 1.523e-12 ***
Condition         1 0.15910 0.15910  6.9270  0.010861 *
Gender            1 0.04292 0.04292  1.8688  0.176886
Group:Condition   1 0.27378 0.27378 11.9201  0.001043 **
Residuals        58 1.33216 0.02297
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> boxplot(Learning~Group*Condition+Gender,
+ col=c(rep("pink",4),rep("light blue",4)))
```



How to find R help and resources on the internet

- R wiki:
  http://rwiki.sciviews.org/doku.php
- R graph gallery:
  http://addictedtor.free.fr/graphiques/thumbs.php
- Kickstarting R:
  http://cran.r-project.org/doc/contrib/Lemon-kickstart/



ISBN: 9780596801700