Four Statistical Tests and Four Statistical Framworks

my name

2024-02-07

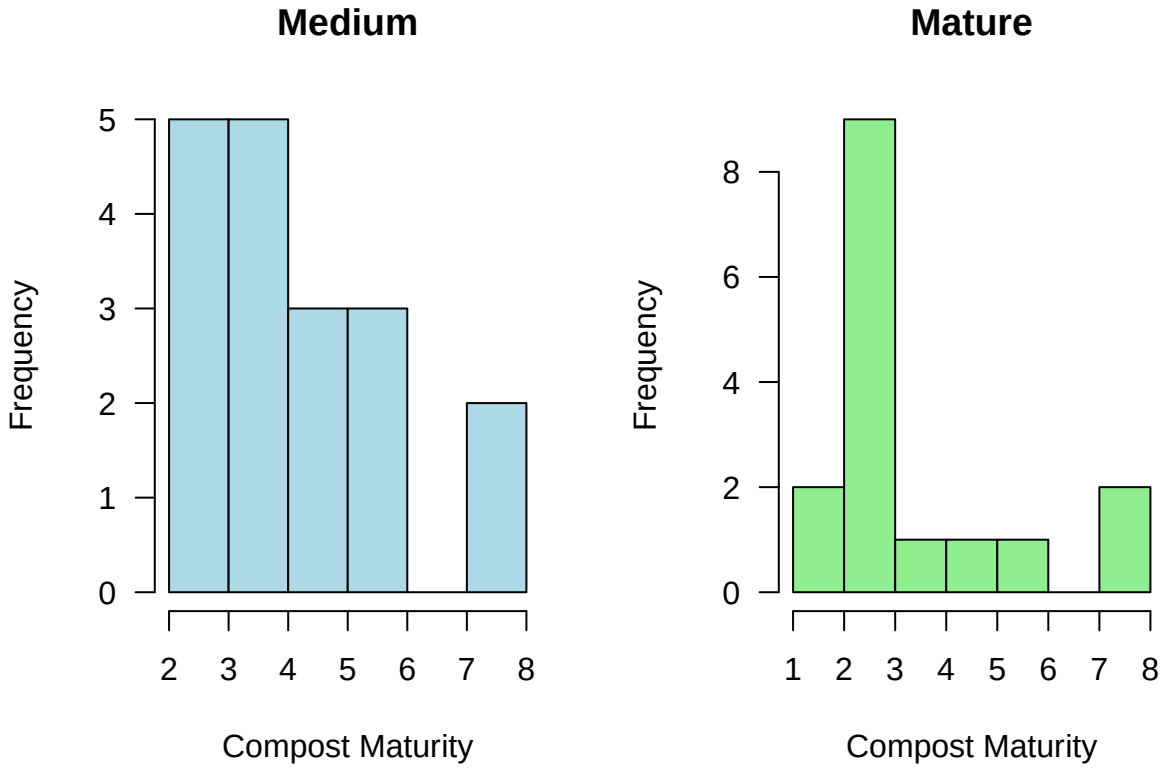
t-test

Compost Maturity

The second test is the paired t-test. This test is used to compare the means of two groups.

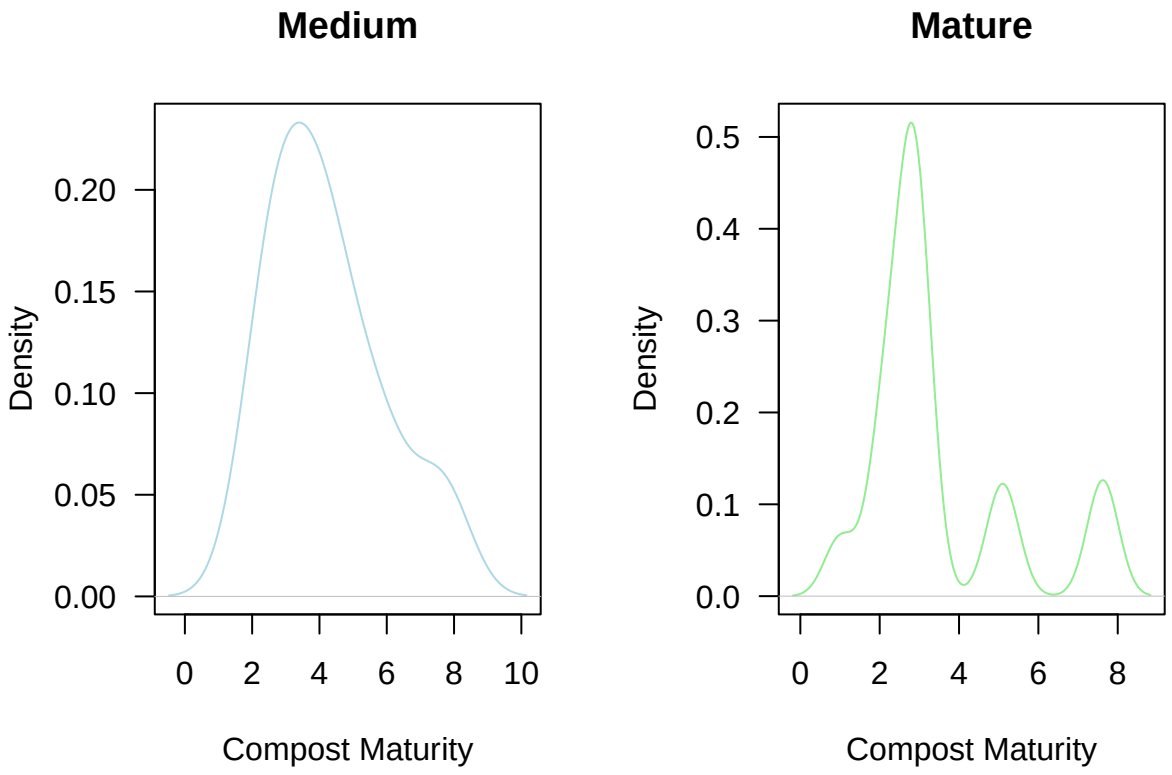
```
medium = c(4, 2, 3.25, 3.25, 2.55, 2.34, 3.7, 3.7, 2.69, 2.77, 4.3, 5.2, 7.63, 7.68, 6, 6, 4.7, 4.5)
mature = c(2, 1, 2.65, 2.95, 2.26, 2.12, 5, 5.2, 2.85, 2.69, 3.1, 2.8, 7.64, 7.61, 3, 2.9)

par(mfrow=c(1,2), las=1)
hist(medium, main="Medium", xlab="Compost Maturity", col="lightblue")
hist(mature, main="Mature", xlab="Compost Maturity", col="lightgreen")
```



showing these in a density distribution

```
par(mfrow=c(1,2), las=1)
plot(density(medium), main="Medium", xlab="Compost Maturity", col="lightblue")
plot(density(mature), main="Mature", xlab="Compost Maturity", col="lightgreen")
```

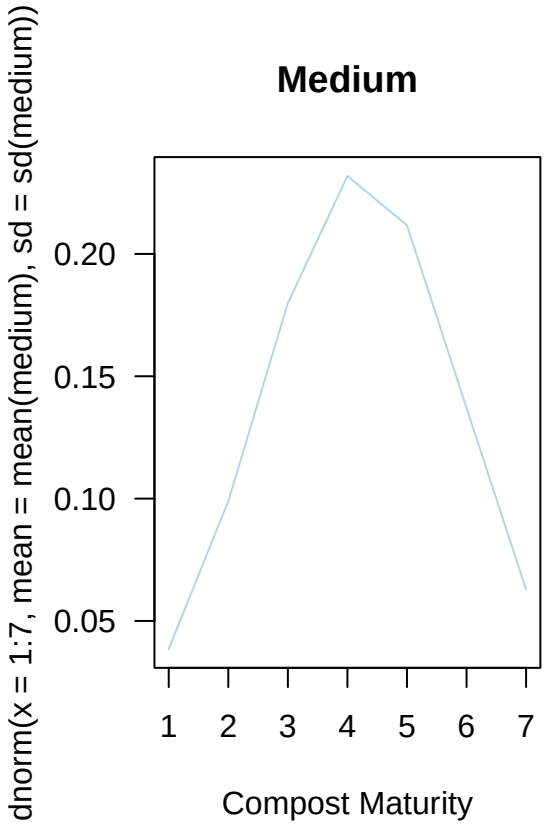


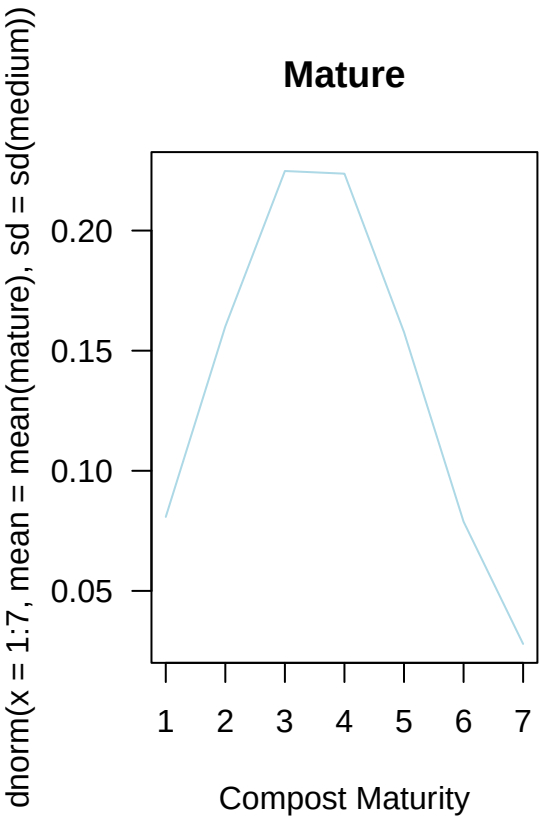

```
  t.test(medium, mature, paired=FALSE)
```

```
##
##  Welch Two Sample t-test
##
## data:  medium and mature
## t = 1.2052, df = 30.366, p-value = 0.2374
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5209723  2.0230557
## sample estimates:
## mean of x mean of y
##  4.236667  3.485625
```

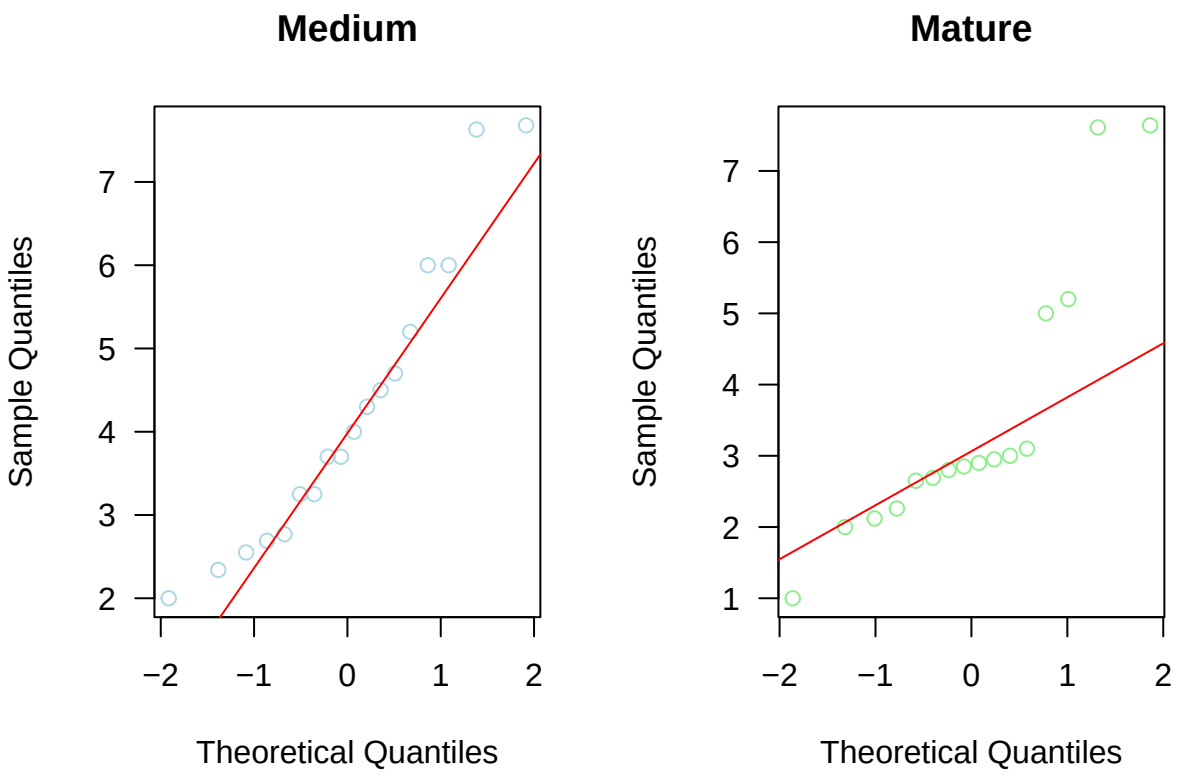
Ploting the data using a normal distribution

```
par(mfrow=c(1,2), las=1)
plot(dnorm(x=1:7, mean=mean(medium), sd=sd(medium)), main="Medium", xlab="Compost Maturity", ty="l", co
plot(dnorm(x=1:7, mean=mean(mature), sd=sd(medium)), main="Mature", xlab="Compost Maturity", ty="l", co
```






```
qqnorm(medium, main="Medium", col="lightblue")
qqline(medium, col="red")
qqnorm(mature, main="Mature", col="lightgreen")
qqline(mature, col="red")
```



ANOVA

For this assignment, you will using the following data sets to analyze and interpret the results of four statistical tests and four statistical frameworks.

Testing if Treatment Means are Equal

The first test is the Analysis of Variances (ANOVA) test. This test is used to compare the means of three or more groups.

```
treatments = c("A", "B", "C", "D")
```

Creating a Dataset

The data set that you will be using is XXX and 10 replicated measurements for each treatment.

To create the data set, use the following code that defines the treatments and generates the data for each treatment. The data is generated using the `rnorm` function. The `rnorm` function generates random numbers from a normal distribution. The `mean` and `sd` parameters are used to specify the mean and standard deviation of the normal distribution. The `rnorm` function is used to generate 10 random numbers for each treatment. The mean and standard deviation of the normal distribution are set to 10 and 2, respectively.

```
print(xtable(cbind(replicates, a, b, c, d)), type="latex")
```

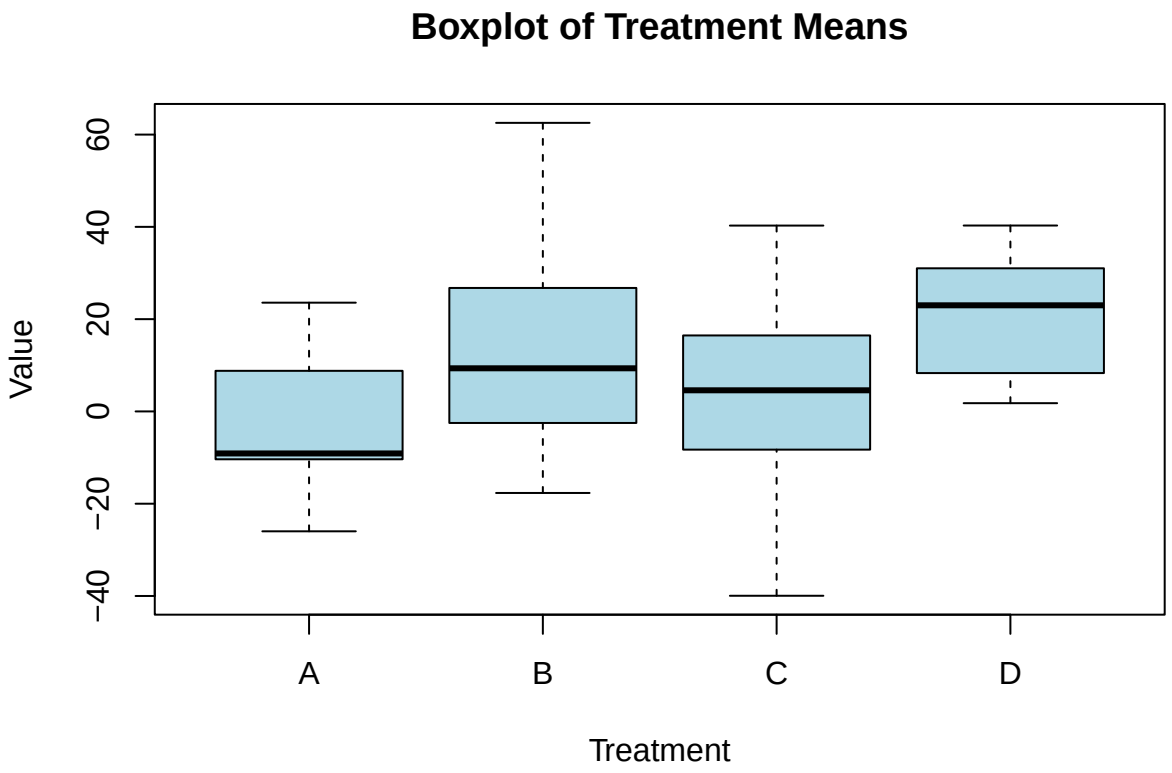
```
## % latex table generated in R 4.2.2 by xtable 1.8-4 package
## % Wed Feb  7 18:12:59 2024
## \begin{table}[ht]
```

```
## \centering
## \begin{tabular}{rrrrrr}
##   \hline
##  & replicates & a & b & c & d \\
##   \hline
## 1 & 1.00 & -9.59 & 8.71 & 40.28 & 22.08 \\
##   2 & 2.00 & 8.80 & -7.21 & -12.34 & 31.02 \\
##   3 & 3.00 & 6.53 & 12.92 & -39.95 & 26.85 \\
##   4 & 4.00 & 17.57 & 62.54 & 8.01 & 40.29 \\
##   5 & 5.00 & -20.56 & -17.67 & -8.28 & 7.30 \\
##   6 & 6.00 & -10.37 & 3.97 & -2.20 & 31.26 \\
##   7 & 7.00 & -25.98 & 26.76 & 1.15 & 8.31 \\
##   8 & 8.00 & 23.57 & 9.98 & 15.04 & 14.01 \\
##   9 & 9.00 & -8.62 & 36.67 & 32.14 & 23.92 \\
##   10 & 10.00 & -10.27 & -2.50 & 16.47 & 1.79 \\
##   11 & 1.00 & -9.59 & 8.71 & 40.28 & 22.08 \\
##   12 & 2.00 & 8.80 & -7.21 & -12.34 & 31.02 \\
##   13 & 3.00 & 6.53 & 12.92 & -39.95 & 26.85 \\
##   14 & 4.00 & 17.57 & 62.54 & 8.01 & 40.29 \\
##   15 & 5.00 & -20.56 & -17.67 & -8.28 & 7.30 \\
##   16 & 6.00 & -10.37 & 3.97 & -2.20 & 31.26 \\
##   17 & 7.00 & -25.98 & 26.76 & 1.15 & 8.31 \\
##   18 & 8.00 & 23.57 & 9.98 & 15.04 & 14.01 \\
##   19 & 9.00 & -8.62 & 36.67 & 32.14 & 23.92 \\
##   20 & 10.00 & -10.27 & -2.50 & 16.47 & 1.79 \\
##   21 & 1.00 & -9.59 & 8.71 & 40.28 & 22.08 \\
##   22 & 2.00 & 8.80 & -7.21 & -12.34 & 31.02 \\
##   23 & 3.00 & 6.53 & 12.92 & -39.95 & 26.85 \\
##   24 & 4.00 & 17.57 & 62.54 & 8.01 & 40.29 \\
##   25 & 5.00 & -20.56 & -17.67 & -8.28 & 7.30 \\
##   26 & 6.00 & -10.37 & 3.97 & -2.20 & 31.26 \\
##   27 & 7.00 & -25.98 & 26.76 & 1.15 & 8.31 \\
##   28 & 8.00 & 23.57 & 9.98 & 15.04 & 14.01 \\
##   29 & 9.00 & -8.62 & 36.67 & 32.14 & 23.92 \\
##   30 & 10.00 & -10.27 & -2.50 & 16.47 & 1.79 \\
##   31 & 1.00 & -9.59 & 8.71 & 40.28 & 22.08 \\
##   32 & 2.00 & 8.80 & -7.21 & -12.34 & 31.02 \\
##   33 & 3.00 & 6.53 & 12.92 & -39.95 & 26.85 \\
##   34 & 4.00 & 17.57 & 62.54 & 8.01 & 40.29 \\
##   35 & 5.00 & -20.56 & -17.67 & -8.28 & 7.30 \\
##   36 & 6.00 & -10.37 & 3.97 & -2.20 & 31.26 \\
##   37 & 7.00 & -25.98 & 26.76 & 1.15 & 8.31 \\
##   38 & 8.00 & 23.57 & 9.98 & 15.04 & 14.01 \\
##   39 & 9.00 & -8.62 & 36.67 & 32.14 & 23.92 \\
##   40 & 10.00 & -10.27 & -2.50 & 16.47 & 1.79 \\
##    \hline
## \end{tabular}
## \end{table}
```

Boxplot

The boxplot is used to visualize the data

```r
boxplot(Value ~ Treatment, data = anovadata, col = "lightblue", main = "Boxplot of Treatment Means", xla
```



Testing the Assumptions of ANOVA

The first step in the ANOVA test is to test the assumptions of the test. The assumptions of the ANOVA test are that the data is normally distributed and that the variances of the groups are equal.

The normality of the data is tested using the Shapiro-Wilk test. The Shapiro-Wilk test is used to test the null hypothesis that the data is normally distributed. The alternative hypothesis is that the data is not normally distributed.

```r
summary(aov(Value ~ Treatment, data = anovadata))
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Treatment    3   3132  1043.8   2.793 0.0542 .
## Residuals   36  13452   373.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The null hypothesis is that the means of the three species are equal. The alternative hypothesis is that the means of the three species are not equal. The ANOVA test is used to test the null hypothesis.

Test of Association

Polluters and Mobility

The third test is the Pearson correlation test. This test is used to test the association between two or more categories of count data.

```r
set.seed(78889)
# 2 x 2 contigency table
frequency = round(c(c(105+119)/2, c(93+150+97+179)/4, c(60+35+51)/3, 85), 0) #down, then over
dimnames = list(c( "Non-mobile", "Mobile"), c("Polluter", "Non-polluter"))
bubbles = as.table(matrix(frequency, nrow=2, dimnames=dimnames))
bubbles
```

```
##            Polluter Non-polluter
## Non-mobile      112           49
## Mobile          130           85
```

```r
chisq.test(bubbles)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  bubbles
## X-squared = 2.9388, df = 1, p-value = 0.08647
```

Logistic Regression

The third test is the logistic regression test. This test is used to test the association between a binary response variable and one or more predictor variables.

Distance and Success

Small square in tape on floor, 10x10 cm, between 20-300 cm, try to get in the square, by rolling or sliding the socket extension.

```r
set.seed(78889)
Distance = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10) + rnorm(20, 2, 1.5)
Success = c(0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1)

group1 = c(95, 80, 55, 105, 200, 45, 305)
success1 = c(1, 1, 0, 0, 1,1, 1)
group2 = c(10, 20, 40, 100, 150, 250)
success2 = c(1, 1, 1, 1 , 1 , 0)
group3 = c(130, 65, 100, 120, 185, 240)
success3 = c(1, 1, 1, 0, 1, 1)
group4 = c(152, 92, 90, 95, 92, 215)
success4 = c(1, 1, 0, 1, 1, 0)
group5 = c(25, 33, 44, 105, 200, 222, 133, 243, 310)
success5 = c(1, 1, 1, 0, 0,0, 0, 0, 0)

mydata = data.frame(Distance = c(group1,group2, group3, group4, group5), Success = c(success1, success2

#mydata = data.frame(Distance, out)
mydata <- mydata[order(mydata$Distance),]
mydata$logical <- as.logical(mydata$Success)
head(mydata)
```

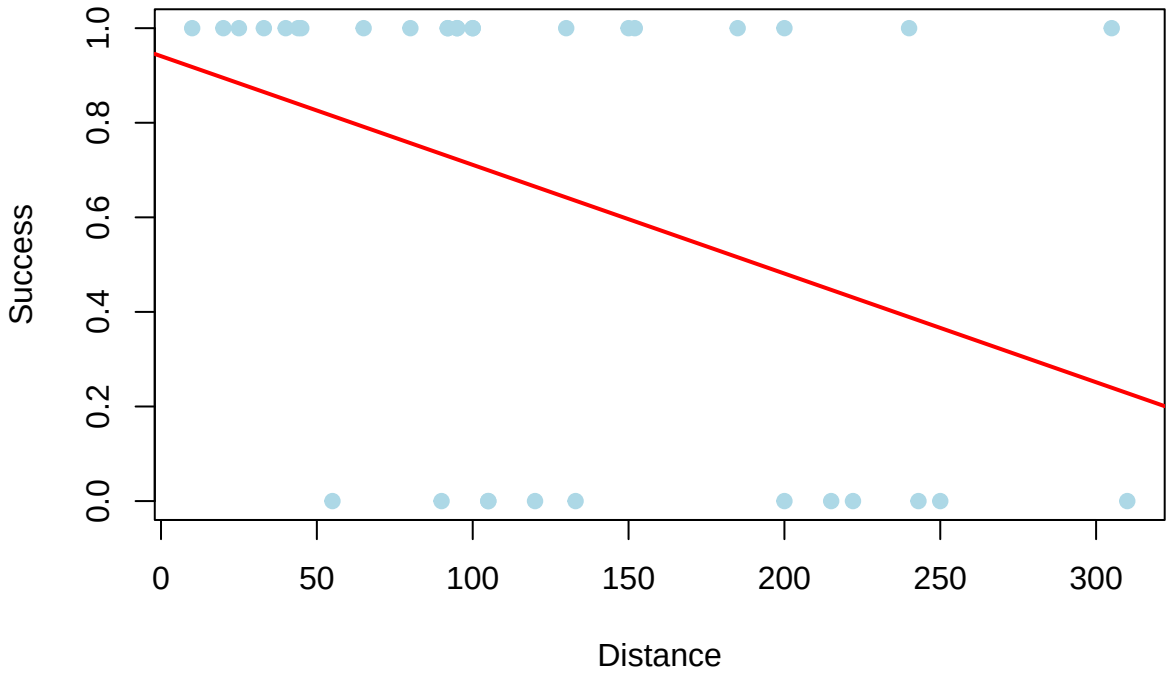
```
##    Distance Success logical
## 8        10       1    TRUE
```

```
## 9        20       1    TRUE
## 26       25       1    TRUE
## 27       33       1    TRUE
## 10       40       1    TRUE
## 28       44       1    TRUE
```

```
str(mydata)
```

```
## 'data.frame':    34 obs. of  3 variables:
##  $ Distance: num  10 20 25 33 40 44 45 55 65 80 ...
##  $ Success : num  1 1 1 1 1 1 1 0 1 1 ...
##  $ logical : logi  TRUE TRUE TRUE TRUE TRUE TRUE ...
```

```
plot(mydata$Distance, mydata$Success, col="lightblue", pch=19, ylab="Success", xlab="Distance")
abline(coef(lm(Success ~ Distance, data = mydata)), col="red", lwd=2)
```




```
mylogit <- glm(logical ~ Distance, data = mydata, family = "binomial")
summary(mylogit)
```

```
##
## Call:
## glm(formula = logical ~ Distance, family = "binomial", data = mydata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8301  -1.0213   0.6047   0.7922   1.7168
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.057053   0.805903   2.552   0.0107 *
## Distance    -0.010723   0.005042  -2.127   0.0334 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 44.149  on 33  degrees of freedom
## Residual deviance: 38.877  on 32  degrees of freedom
## AIC: 42.877
##
## Number of Fisher Scoring iterations: 4
```

Plot Results

```
plot(mydata$Distance, mydata$logical, col="lightblue", pch=19, ylab=c("Success"), xlab=c("Distance"), yl
lines(mydata$Distance, fitted(mylogit), col="red", lwd=2)
```