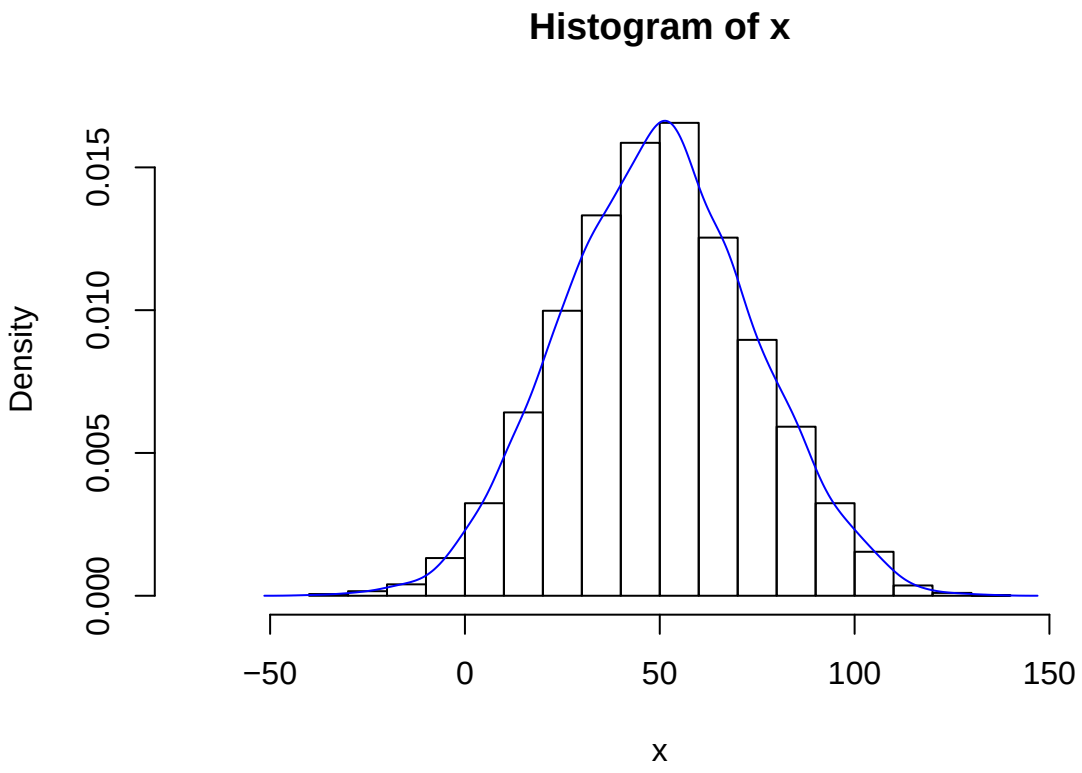
Lecture 02 - R code

WILD 502 - Jay Rotella

Normal Distribution

```
x = rnorm(5000, mean = 50, sd = 25)
hist(x, prob = TRUE, xlim = c(-70, 170))
lines(density(x), col = 'blue')
```

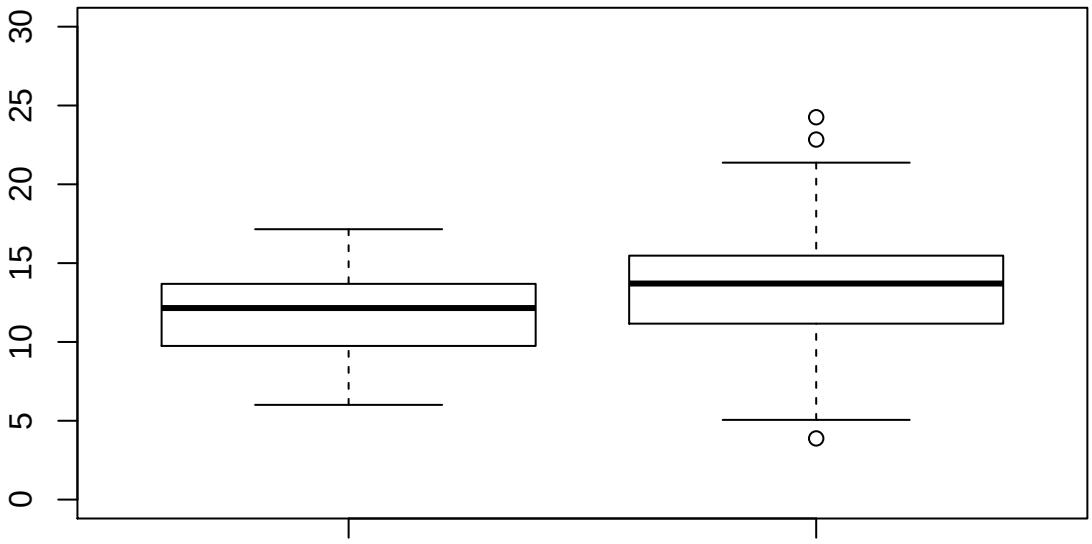


Example of comparing 2 means

```
mu1 = 12.5
mu2 = 14
sig1 = 3
sig2 = 5

x1 = rnorm(25, mu1, sig1)
x2 = rnorm(25, mu2, sig2)

boxplot(x1, x2, ylim = c(0, 30))
```




```
t.test(x1, x2, var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  x1 and x2
## t = -1.5589, df = 48, p-value = 0.1256
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -4.2329222  0.5356598
## sample estimates:
## mean of x mean of y
##  11.90301  13.75165
```

Work with variance, covariance, and correlation

```
# calculation of variance
sum( (x1 - mean(x1)) * (x1 - mean(x1)) ) / (length(x1) - 1)
```

```
## [1] 9.568631
```

```
# compare with output from "var" function
var(x1)
```

```
## [1] 9.568631
```

```
# calculation of covariance
#  notice similarity to variance calculation
sum( (x1 - mean(x1)) * (x2 - mean(x2)) ) / (length(x1) - 1)
```

```
## [1] 3.268068
```

```
# compare with output from "cov" function
cov(x1, x2)
```

```
## [1] 3.268068
```

```
# calculate correlation
cov(x1, x2) / (sd(x1) * sd(x2))
```

```
## [1] 0.2088615
```

```
# compare with output from "cor" function
cor(x1, x2)
```

```
## [1] 0.2088615
```

```
# variance-covariance and correlation matrices
cov(cbind(x1, x2))
```

```
##          x1        x2
## x1 9.568631  3.268068
## x2 3.268068 25.586779
```

```
cor(cbind(x1, x2))
```

```
##           x1        x2
## x1 1.0000000 0.2088615
## x2 0.2088615 1.0000000
```

Example of different hypothesis testing methods

Next, we'll work with a survival analysis of data from deer fawns, which you'll work with soon in lab. The data contain survival information (status = live or dead) for 2 different areas for male and female fawns whose body mass and length were measured.

```
# read in the data
fawns <- read.table(file = "http://www.montana.edu/rotella/documents/502/FawnsData.txt",
                    sep = "\t", header = TRUE)
# create status variable where 1 = lived and 0 = died
fawns$status <- NA
# set status to 1 for those that lived, i.e.,had 10 history
fawns$status[which(fawns$eh == 10)] <- 1
# set status to 0 for those that died, i.e.,had 11 history
fawns$status[which(fawns$eh == 11)] <- 0

# treat area, sex and status as factors and set labels
fawns$area = factor(fawns$area,
                    levels = c(0, 1),
                    labels = c("control","treatment"))

fawns$sex = factor(fawns$sex,
                    levels = c(0, 1),
                    labels = c("female","male"))

fawns$status = factor(fawns$status,
                    levels = c(0, 1),
                    labels = c("died","lived"))

# examine the data and a few simple summaries
head(fawns, 4)
```

```
##    radio eh freq      area    sex mass length status
## 1 191.08 10    1 treatment   male 34.2  125.5  lived
## 2 191.19 11    1   control female 27.5  112.0   died
## 3 191.35 10    1   control female 35.5  128.0  lived
## 4 191.36 11    1   control female 33.0  122.0   died
```

```
summary(fawns)
```

```
##      radio             eh             freq          area        sex
##  Min.   :191.1   Min.   :10.00   Min.   :1   control  :59   female:57
##  1st Qu.:192.0   1st Qu.:10.00   1st Qu.:1   treatment:56   male  :58
##  Median :192.4   Median :11.00   Median :1
##  Mean   :192.2   Mean   :10.59   Mean   :1
##  3rd Qu.:192.7   3rd Qu.:11.00   3rd Qu.:1
##  Max.   :192.8   Max.   :11.00   Max.   :1
##       mass           length       status
##  Min.   :22.80   Min.   :108.0   died :68
##  1st Qu.:31.90   1st Qu.:120.0   lived:47
##  Median :33.60   Median :124.0
##  Mean   :34.38   Mean   :123.2
##  3rd Qu.:36.60   3rd Qu.:127.0
##  Max.   :72.00   Max.   :135.5
```

```
# look at some basic summary statistics
table(fawns$sex, fawns$status)
```

```
##
##          died lived
##   female   28    29
##   male     40    18
```

```
# work with proportions and have row values sum to 1
prop.table(table(fawns$sex, fawns$status),
           margin = 1)
```

```
##
##               died     lived
##   female 0.4912281 0.5087719
##   male   0.6896552 0.3103448
```

```
prop.table(table(fawns$area, fawns$status),
           margin = 1)
```

```
##
##                  died     lived
##   control   0.6101695 0.3898305
##   treatment 0.5714286 0.4285714
```

Run some competing models

```
m0 <- glm(formula = status ~ 1,
          family = binomial(link = logit), data = fawns)

ms <- glm(formula = status ~ sex,
          family = binomial(link = logit), data = fawns)

ml <- glm(formula = status ~ length,
          family = binomial(link = logit), data = fawns)

msl <- glm(formula = status ~ sex + length,
           family = binomial(link = logit), data = fawns)
```

```r
prop.table(table(fawns$status))
```

```
##
##      died     lived
## 0.5913043 0.4086957
```

```r
# check if predictions are for the proportion that lived or died
#  it's important to be certain of which outcome is being predicted
# you can look at the output from the null model to check
mean(predict(m0, type = "response"))
```

```
## [1] 0.4086957
```

Make pairwise comparisons using likelihood-ratio tests

```r
anova(m0, ms,  test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: status ~ 1
## Model 2: status ~ sex
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       114     155.57
## 2       113     150.85  1   4.7186  0.02984 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(m0, ml,  test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: status ~ 1
## Model 2: status ~ length
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       114     155.57
## 2       113     150.61  1   4.9552  0.02601 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(m0, msl, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: status ~ 1
## Model 2: status ~ sex + length
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       114     155.57
## 2       112     143.58  2   11.986 0.002496 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(ms, ml,  test = "Chisq")   # Invalid!
```

```
## Analysis of Deviance Table
##
```

```
## Model 1: status ~ sex
## Model 2: status ~ length
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       113     150.85
## 2       113     150.61  0  0.23662
```

```
anova(ms, msl, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: status ~ sex
## Model 2: status ~ sex + length
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       113     150.85
## 2       112     143.58  1   7.2671 0.007023 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(ml, msl, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: status ~ length
## Model 2: status ~ sex + length
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       113     150.61
## 2       112     143.58  1   7.0305 0.008013 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Use Information-theoretic method for model comparisons

For some of you the use of Akaike's Information Criterion (AIC) will be new. For now, we're just going to expose you to the ideas that it's a useful way to compare models of the *same* response variable fit to the *same* number of observations. It's especially useful for comparing non-nested models (e.g., models **ms** and **ml**). We'll see other examples of non-nested models where you have competing models that each use a different functional form of a covariate of interest (e.g., age, ln(age), age + age^2). Also, AICc is a version of AIC that's adjusted for sample size (more on that in the days ahead). Finally, AIC is not the only information criterion available in data analysis. You might also see others such as BIC, SIC, DIC, and WAIC in the literature. We'll focus on using AICc in this course, but we will also discuss the others and why they are of interest and used in some settings.

```
# need to have the package 'AICcmodavg' installed for this to work
# if don't have the package, can use the following line w/o the '#'
# install.packages("AICcmodavg")
library(AICcmodavg)

candidates = list(m0, ms, ml, msl)
model.names = c("intercept", "sex", "length", "sex+length")
# Be sure that the order of model names matches
#  the order used for the candidate models.
# Check out the help for 'aictab' for another way to
#  implement this function.
aictab(cand.set = candidates, modnames = model.names, sort = TRUE)
```

```
##
```

```
## Model selection based on AICc:
##
##            K   AICc Delta_AICc AICcWt Cum.Wt     LL
## sex+length 3 149.80       0.00   0.85   0.85 -71.79
## length     2 154.72       4.92   0.07   0.92 -75.31
## sex        2 154.96       5.16   0.06   0.98 -75.42
## intercept  1 157.60       7.80   0.02   1.00 -77.78
```

Examine output for best-supported model

There's a lot more one would do in the process of making inferences from these analyses, e.g., goodness-of-fit, diagnostics, etc. However, for now we'll just look at the coefficient estimates and their confidence intervals.

```
# look at output for best-supported model
summary(msl)
```

```
##
## Call:
## glm(formula = status ~ sex + length, family = binomial(link = logit),
##     data = fawns)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5508  -1.0296  -0.6477   1.1354   2.0344
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.99921    5.10338  -2.547   0.0109 *
## sexmale      -1.07568    0.41702  -2.579   0.0099 **
## length        0.10650    0.04162   2.559   0.0105 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 155.57  on 114  degrees of freedom
## Residual deviance: 143.58  on 112  degrees of freedom
## AIC: 149.58
##
## Number of Fisher Scoring iterations: 4
```