Getting the data into R

• Put frequencies directly into tables
• Read a data frame with frequencies, number of rows = number of cells
• Read a raw data file, number of rows = n
Put frequencies directly into tables

**Table 3-2**
Counts for Structural Habitat Categories for *Anolis* Lizards of Bimini: *sagrei* Adult Males vs. *angusticeps* Adult Males (Schoener [1968])

(a) Observed data

<table>
<thead>
<tr>
<th>perch height (feet)</th>
<th>perch diameter (inches)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><em>sagrei</em></td>
<td>&gt; 2.5</td>
<td>≤ 2.5</td>
</tr>
<tr>
<td>&gt; 5.0</td>
<td>15</td>
<td>18</td>
<td>21</td>
</tr>
<tr>
<td>≤ 5.0</td>
<td>48</td>
<td>84</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td><em>angusticeps</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&gt; 2.5</td>
<td>1</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>≤ 2.5</td>
<td>2</td>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>
> lizards <- numeric(8); lizards
[1] 0 0 0 0 0 0 0 0
> dim(lizards) <- c(2,2,2) # Now a 2x2x2 table: Rows, cols, layers
> lizards
 , , 1

    [,1] [,2]
[1,]  0  0
[2,]  0  0

 , , 2

    [,1] [,2]
[1,]  0  0
[2,]  0  0

> # 1 = Perch Height, 2 = Perch Diameter, 3 = Species
> lizards[,,1] <- rbind( c(15,18),
+                      c(48,84)  )
> lizards[,,2] <- rbind( c(21,1),
+                      c(3,2)   )
Need Labels

> lizards

, , 1

[,1] [,2]
[1,]  15  18
[2,]  48  84

, , 2

[,1] [,2]
[1,]  21   1
[2,]   3   2
Labels are dimnames of the array: A list

> # Labels are dimnames: A list
> lizlabels <- list() # An empty list
> lizlabels$Height <- c("gt 5.0","le 5.0")
> lizlabels$Diameter <- c("le 2.5","gt 2.5")
> lizlabels$Species <- c("Sagrei","Angusticeps")
> lizlabels
$Height
[1] "gt 5.0" "le 5.0"

$Diameter
[1] "le 2.5" "gt 2.5"

$Species
[1] "Sagrei" "Angusticeps"
This is Better

> dimnames(lizards) <- lizlabels ; lizards
 , , Species = Sagrei

Diameter
Height  le 2.5 gt 2.5
  gt 5.0   15   18
  le 5.0   48   84

, , Species = Angusticeps

Diameter
Height  le 2.5 gt 2.5
  gt 5.0   21   1
  le 5.0   3   2

Table 3-2
Counts for Structural Habitat Categories for Anolis Lizards of Bimini: sagrei Adult Males vs. angusticeps Adult Males (Schoener [1968])
(a) Observed data

<table>
<thead>
<tr>
<th>perch diameter (inches)</th>
<th>perch height (feet)</th>
<th>sagrei</th>
<th>angusticeps</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; 2.5</td>
<td>&gt; 5.0</td>
<td>15</td>
<td>18</td>
</tr>
<tr>
<td>&gt; 2.5</td>
<td>&gt; 5.0</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td>&gt; 2.5</td>
<td>&gt; 5.0</td>
<td>48</td>
<td>84</td>
</tr>
<tr>
<td>&gt; 2.5</td>
<td>&gt; 5.0</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
**margin.table**

```r
> species_by_height = margin.table(lizards, margin=c(3,1))
> species_by_height

<table>
<thead>
<tr>
<th>Height</th>
<th>Species</th>
<th>gt 5.0</th>
<th>le 5.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sagreii</td>
<td>33</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>Angusticeps</td>
<td>22</td>
<td>5</td>
</tr>
</tbody>
</table>

> spec_by_height_by_diam = margin.table(lizards, margin=c(3,1,2))
> spec_by_height_by_diam

, , Diameter = le 2.5

<table>
<thead>
<tr>
<th>Height</th>
<th>Species</th>
<th>gt 5.0</th>
<th>le 5.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sagreii</td>
<td>15</td>
<td>48</td>
</tr>
<tr>
<td></td>
<td>Angusticeps</td>
<td>21</td>
<td>3</td>
</tr>
</tbody>
</table>

, , Diameter = gt 2.5

<table>
<thead>
<tr>
<th>Height</th>
<th>Species</th>
<th>gt 5.0</th>
<th>le 5.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sagreii</td>
<td>18</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>Angusticeps</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
Method 2: Read a data frame

> # See what a data frame looks like
> as.data.frame.table(lizards)

<table>
<thead>
<tr>
<th>Height</th>
<th>Diameter</th>
<th>Species</th>
<th>Freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>gt 5.0</td>
<td>le 2.5</td>
<td>Sagrei</td>
</tr>
<tr>
<td>2</td>
<td>le 5.0</td>
<td>le 2.5</td>
<td>Sagrei</td>
</tr>
<tr>
<td>3</td>
<td>gt 5.0</td>
<td>gt 2.5</td>
<td>Sagrei</td>
</tr>
<tr>
<td>4</td>
<td>le 5.0</td>
<td>gt 2.5</td>
<td>Sagrei</td>
</tr>
<tr>
<td>5</td>
<td>gt 5.0</td>
<td>le 2.5</td>
<td>Angusticeps</td>
</tr>
<tr>
<td>6</td>
<td>le 5.0</td>
<td>le 2.5</td>
<td>Angusticeps</td>
</tr>
<tr>
<td>7</td>
<td>gt 5.0</td>
<td>gt 2.5</td>
<td>Angusticeps</td>
</tr>
<tr>
<td>8</td>
<td>le 5.0</td>
<td>gt 2.5</td>
<td>Angusticeps</td>
</tr>
</tbody>
</table>
Read data frame from an external file, say a plain text file

<table>
<thead>
<tr>
<th>Admit</th>
<th>Gender</th>
<th>Dept</th>
<th>Freq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>Male</td>
<td>A</td>
<td>512</td>
</tr>
<tr>
<td>Rejected</td>
<td>Male</td>
<td>A</td>
<td>313</td>
</tr>
<tr>
<td>Admitted</td>
<td>Female</td>
<td>A</td>
<td>89</td>
</tr>
<tr>
<td>Rejected</td>
<td>Female</td>
<td>A</td>
<td>19</td>
</tr>
<tr>
<td>Admitted</td>
<td>Male</td>
<td>B</td>
<td>353</td>
</tr>
<tr>
<td>Rejected</td>
<td>Male</td>
<td>B</td>
<td>207</td>
</tr>
<tr>
<td>Admitted</td>
<td>Female</td>
<td>B</td>
<td>17</td>
</tr>
<tr>
<td>Rejected</td>
<td>Female</td>
<td>B</td>
<td>8</td>
</tr>
<tr>
<td>Admitted</td>
<td>Male</td>
<td>C</td>
<td>120</td>
</tr>
<tr>
<td>Rejected</td>
<td>Male</td>
<td>C</td>
<td>205</td>
</tr>
<tr>
<td>Admitted</td>
<td>Female</td>
<td>C</td>
<td>202</td>
</tr>
<tr>
<td>Rejected</td>
<td>Female</td>
<td>C</td>
<td>391</td>
</tr>
<tr>
<td>Admitted</td>
<td>Male</td>
<td>D</td>
<td>138</td>
</tr>
<tr>
<td>Rejected</td>
<td>Male</td>
<td>D</td>
<td>279</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Berkeley data are on the class website

```r
> berkeley <- read.table("http://www.utstat.toronto.edu/~brunner/312f12/code_n_data/BerkeleyFrame.txt")
>
> berkeley

       Admit Gender Dept Freq
    1  Admitted  Male   A  512
    2  Rejected  Male   A  313
    3  Admitted Female  A  89
    4  Rejected Female  A  19
    5  Admitted  Male   B  353
    6  Rejected  Male   B  207
    7  Admitted Female  B  17
    8  Rejected Female  B  8
    9  Admitted  Male   C  120
   10  Rejected  Male   C  205

etc.
```
xtabs(Counts ~ Vars separated by + signs, data = Name of data frame)

> UCB <- xtabs(Freq ~ Dept + Admit + Gender, data = berkeley); UCB

, , Gender = Female

<table>
<thead>
<tr>
<th>Dept</th>
<th>Admitted</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>89</td>
<td>19</td>
</tr>
<tr>
<td>B</td>
<td>17</td>
<td>8</td>
</tr>
<tr>
<td>C</td>
<td>202</td>
<td>391</td>
</tr>
<tr>
<td>D</td>
<td>131</td>
<td>244</td>
</tr>
<tr>
<td>E</td>
<td>94</td>
<td>299</td>
</tr>
<tr>
<td>F</td>
<td>24</td>
<td>317</td>
</tr>
</tbody>
</table>

, , Gender = Male

<table>
<thead>
<tr>
<th>Dept</th>
<th>Admitted</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>512</td>
<td>313</td>
</tr>
<tr>
<td>B</td>
<td>353</td>
<td>207</td>
</tr>
<tr>
<td>C</td>
<td>120</td>
<td>205</td>
</tr>
<tr>
<td>D</td>
<td>138</td>
<td>279</td>
</tr>
<tr>
<td>E</td>
<td>53</td>
<td>138</td>
</tr>
<tr>
<td>F</td>
<td>22</td>
<td>351</td>
</tr>
</tbody>
</table>
Marginal Tables are Easy

> GenderAdmit <- xtabs(Freq ~ Gender + Admit, data = berkeley)
> GenderAdmit

<table>
<thead>
<tr>
<th>Gender</th>
<th>Admitted</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>557</td>
<td>1278</td>
</tr>
<tr>
<td>Male</td>
<td>1198</td>
<td>1493</td>
</tr>
</tbody>
</table>

> prop.table(GenderAdmit,1)

<table>
<thead>
<tr>
<th>Gender</th>
<th>Admitted</th>
<th>Rejected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>0.3035422</td>
<td>0.6964578</td>
</tr>
<tr>
<td>Male</td>
<td>0.4451877</td>
<td>0.5548123</td>
</tr>
</tbody>
</table>

Probably a data frame and xtabs is the easiest way to import data from a published table, if the table has more than 2 dimensions.
Method 3: External raw data file

Programs and data sets

- **Death Penalty Data**
  - Col 1 = Prisoner's Race (1=W, 2=B)
  - Col 2 = Victim's Race (1=W, 2=B)
  - Col 3 = Death Penalty (1=Yes 2=No)
Read data into a data frame, Number of rows = n

```r
> florida <- read.table(file='http://www.utstat.toronto.edu/~brunner/312f12/code_n_data/deathpen.data',col.names=c('Prace','Vrace','DeathPen'))
> florida
     Prace Vrace DeathPen
 1      1     1       1
 2      1     1       1
 3      1     1       1
 4      1     1       1
 5      1     1       1
 6      1     1       1
```

The table function

```r
> PR_by_DP = table(florida$Prace, florida$DeathPen); PR_by_DP

   1 2
 1 19 141
 2 17 149

> # This is pretty hard to read. We can do better by creating factors.
> Prace <- factor(florida$Prace, labels=c('White','Black')) # In order 1,2
> Vrace <- factor(florida$Vrace, labels=c('White','Black'))
> DeathPen <- factor(florida$DeathPen, labels=c('Yes','No'))
> PR_by_DP = table(Prace, DeathPen); PR_by_DP

   DeathPen
Prace   Yes No
  White   19 141
     Black 17 149
```
3-Dimensional table

```r
> deathrow <- table(Prace, DeathPen, Vrace); deathrow
, , Vrace = White

       DeathPen
Prace     Yes  No
  White   19 132
  Black   11  52

, , Vrace = Black

       DeathPen
Prace     Yes  No
  White    0  9
  Black    6 97

> prop.table(deathrow[,1],1) # Proportions of row totals

       DeathPen
Prace      Yes     No
  White 0.1258278 0.8741722
  Black 0.1746032 0.8253968
```
Fitting and testing models with the loglin function

• Hierarchical models only
• Very close to bracket notation
• Give it a table and a list of vectors
• Vectors are vars in a bracket, like c(1,2,4) means [1 2 4]
• Iterative proportional model fitting
• Returns estimated expected frequencies as an option
loglin(table, margin, fit=F, param=F)

> lizards
, , Species = Sagreii

<table>
<thead>
<tr>
<th>Diameter</th>
<th>Height</th>
<th>le 2.5</th>
<th>gt 2.5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>gt 5.0</td>
<td>15</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>le 5.0</td>
<td>48</td>
<td>84</td>
</tr>
</tbody>
</table>

, , Species = Angusticeps

<table>
<thead>
<tr>
<th>Diameter</th>
<th>Height</th>
<th>le 2.5</th>
<th>gt 2.5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>gt 5.0</td>
<td>21</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>le 5.0</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

> lizmodel1 <- loglin(lizards, list(1, c(2, 3)))  # [1] [23]
2 iterations: deviation 0
> lizmodel1
$lrt
[1] 43.87073

$pearson
[1] 47.46099

$df
[1] 3

$margin
$margin[[1]]
[1] "Height"

$margin[[2]]
[1] "Diameter" "Species"

> 1-pchisq(43.87073, df=3)
[1] 1.607684e-09
> 1-pchisq(lizmodel1$lrt, df=lizmodel1$df)
[1] 1.607688e-09
Some options

> lizmodel1b <- loglin(lizards, list('Height', c('Diameter', 'Species')), 
>                       fit=T, param=T)

2 iterations: deviation 0
> lizmodel1b$lrt
[1] 43.87073
> # Same as before, of course
> lizmodel1b$fit # Estimated expected values

, , Species = Sagrei

\[
\begin{array}{ccc}
\text{Diameter} & \text{le 2.5} & \text{gt 2.5} \\
\text{Height} & \text{gt 5.0} & 18.04688 & 29.21875 \\
& \text{le 5.0} & 44.95312 & 72.78125 \\
\end{array}
\]

, , Species = Angusticeps

\[
\begin{array}{ccc}
\text{Diameter} & \text{le 2.5} & \text{gt 2.5} \\
\text{Height} & \text{gt 5.0} & 6.875 & 0.859375 \\
& \text{le 5.0} & 17.125 & 2.140625 \\
\end{array}
\]
## Parameter estimates

```r
> lizmodel1b$param
$'(Intercept)'
[1] 2.467355

$Height
gt 5.0   le 5.0
-0.4563239 0.4563239

$Diameter
le 2.5   gt 2.5
0.3994009 -0.3994009

$Species
Sagrei Angusticeps
1.122860 -1.122860

$Diameter.Species
Species
Diameter Sagrei Angusticeps
le 2.5 -0.6403199 0.6403199
gt 2.5 0.6403199 -0.6403199
```
Two more models

> # [1] [2] [3] Complete independence
> lizmodel2 <- loglin(lizards,list(1,2,3))
2 iterations: deviation 0
> lizmodel2$lrt; lizmodel2$df
[1] 70.07975
[1] 4

> # [1 2] [1 3] [2 3] All 2-way relationships
> lizmodel3 <- loglin(lizards,list(c(1,2),c(1,3),c(2,3)))
4 iterations: deviation 0.07225579
> lizmodel3$lrt; lizmodel3$df
[1] 2.706392
[1] 1