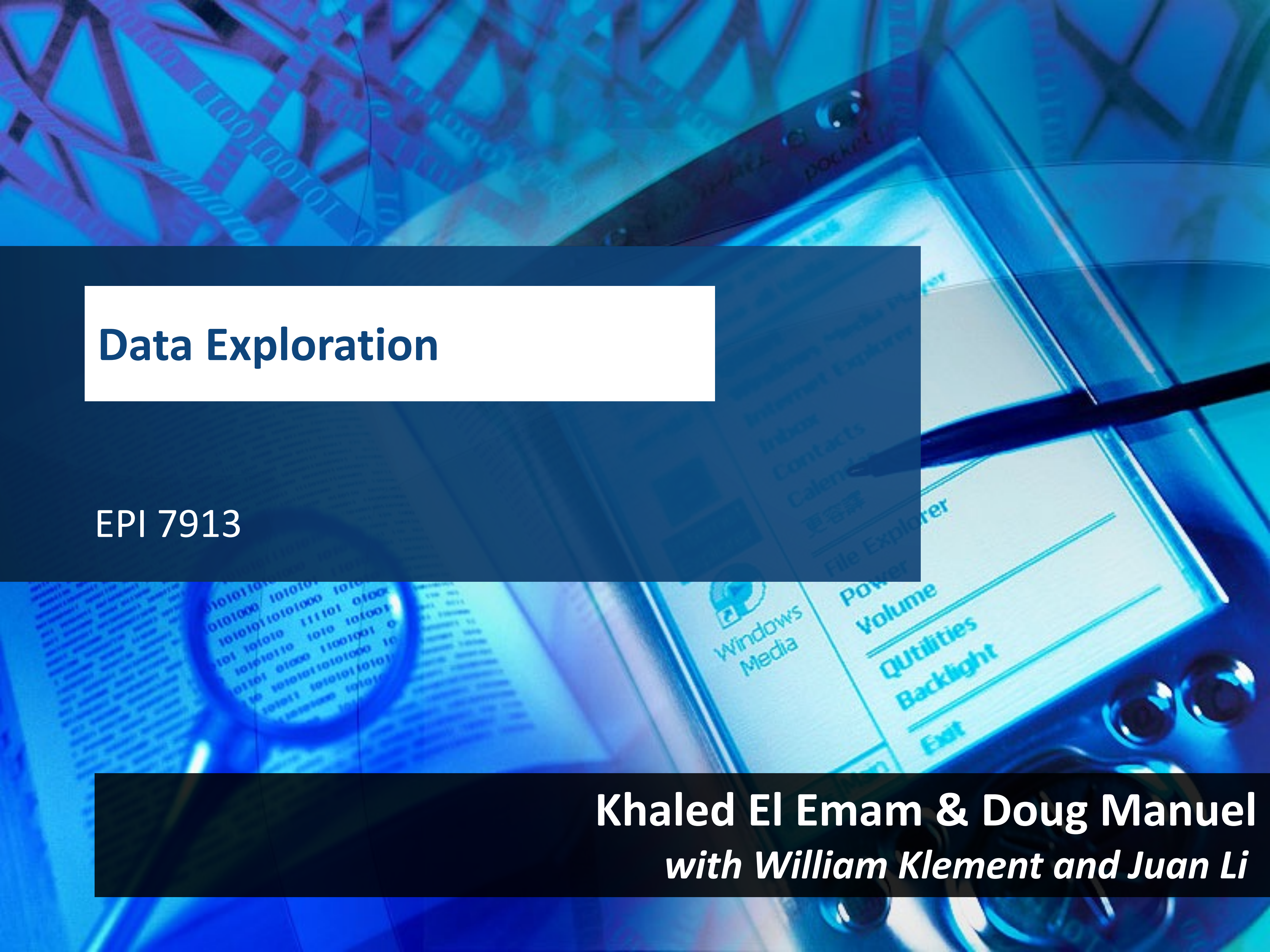


Data Exploration

EPI 7913

Khaled El Emam & Doug Manuel

with William Klement and Juan Li

Data preparation check-list

I. Profile, visualize and identify:
   - structure and distribution
   - format and consistency
   - issues and deficiencies

II. Cleanse and repair:
   - data values: (convert, map, etc.)
   - missing values: (convert, remove, impute)
   - rare values: (combine, remove, ?)

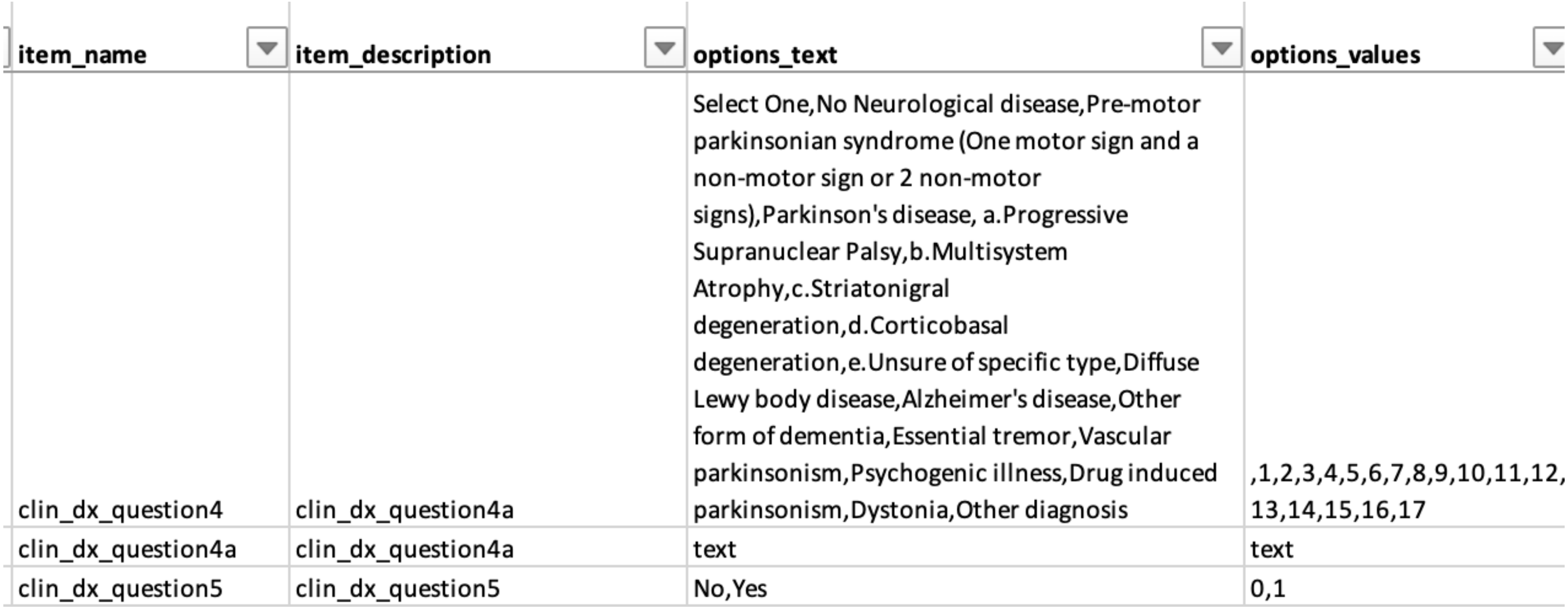
III. Organize structure for better modelling
   - detect outliers

https://towardsdatascience.com/data-cleaning-in-r-made-simple-1b77303b0b17

https://statisticsglobe.com/data-cleaning-r

Data dictionary

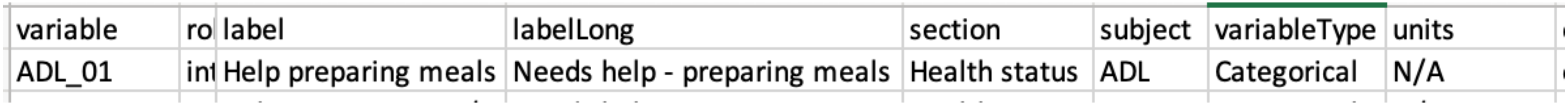
cohort data/ clinical research data



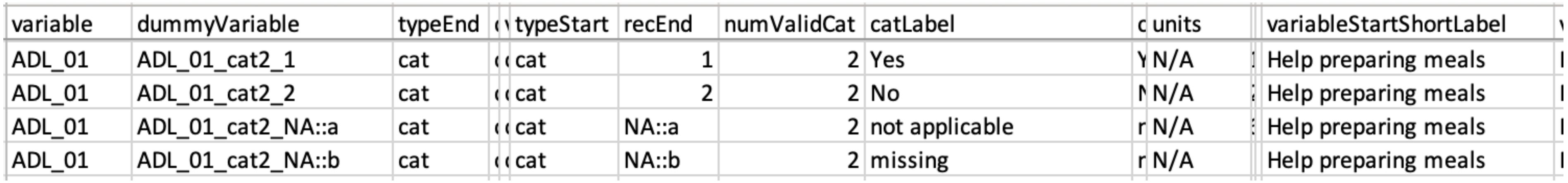
item_name	item_description	options_text	options_values
clin_dx_question4	clin_dx_question4a	Select One,No Neurological disease,Pre-motor parkinsonian syndrome (One motor sign and a non-motor sign or 2 non-motor signs),Parkinson's disease, a.Progressive Supranuclear Palsy,b.Multisystem Atrophy,c.Striatonigral degeneration,d.Corticobasal degeneration,e.Unsure of specific type,Diffuse Lewy body disease,Alzheimer's disease,Other form of dementia,Essential tremor,Vascular parkinsonism,Psychogenic illness,Drug induced parkinsonism,Dystonia,Other diagnosis	,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17
clin_dx_question4a	clin_dx_question4a	text	text
clin_dx_question5	clin_dx_question5	No,Yes	0,1

Data dictionary

variables and variable-details sheets (Canadian Community Health Survey (CCHS))



variable	ro	label	labelLong	section	subject	variableType	units	c
ADL_01	int	Help preparing meals	Needs help - preparing meals	Health status	ADL	Categorical	N/A	c



variable	dummyVariable	typeEnd	c	v	typeStart	recEnd	numValidCat	catLabel	c	units		variableStartShortLabel	v
ADL_01	ADL_01_cat2_1	cat	c	c	cat	1	2	Yes	Y	N/A	1	Help preparing meals	I
ADL_01	ADL_01_cat2_2	cat	c	c	cat	2	2	No	N	N/A	2	Help preparing meals	I
ADL_01	ADL_01_cat2_NA::a	cat	c	c	cat	NA::a	2	not applicable	r	N/A	3	Help preparing meals	I
ADL_01	ADL_01_cat2_NA::b	cat	c	c	cat	NA::b	2	missing	r	N/A		Help preparing meals	I

https://github.com/Big-Life-Lab/cchsflow/tree/master

Data dictionary

survival::lung

inst:	Institution code
time:	Survival time in days
status:	censoring status 1=censored, 2=dead
age:	Age in years
sex:	Male=1 Female=2
ph.ecog:	ECOG performance score as rated by the physician. 0=asymptomatic, 1= symptomatic but completely ambulatory, 2= in bed <50% of the day, 3= in bed > 50% of the day but not bedbound, 4 = bedbound
ph.karno:	Karnofsky performance score (bad=0-good=100) rated by physician
pat.karno:	Karnofsky performance score as rated by patient
meal.cal:	Calories consumed at meals
wt.loss:	Weight loss in last six months (pounds)

Data cleansing tips:

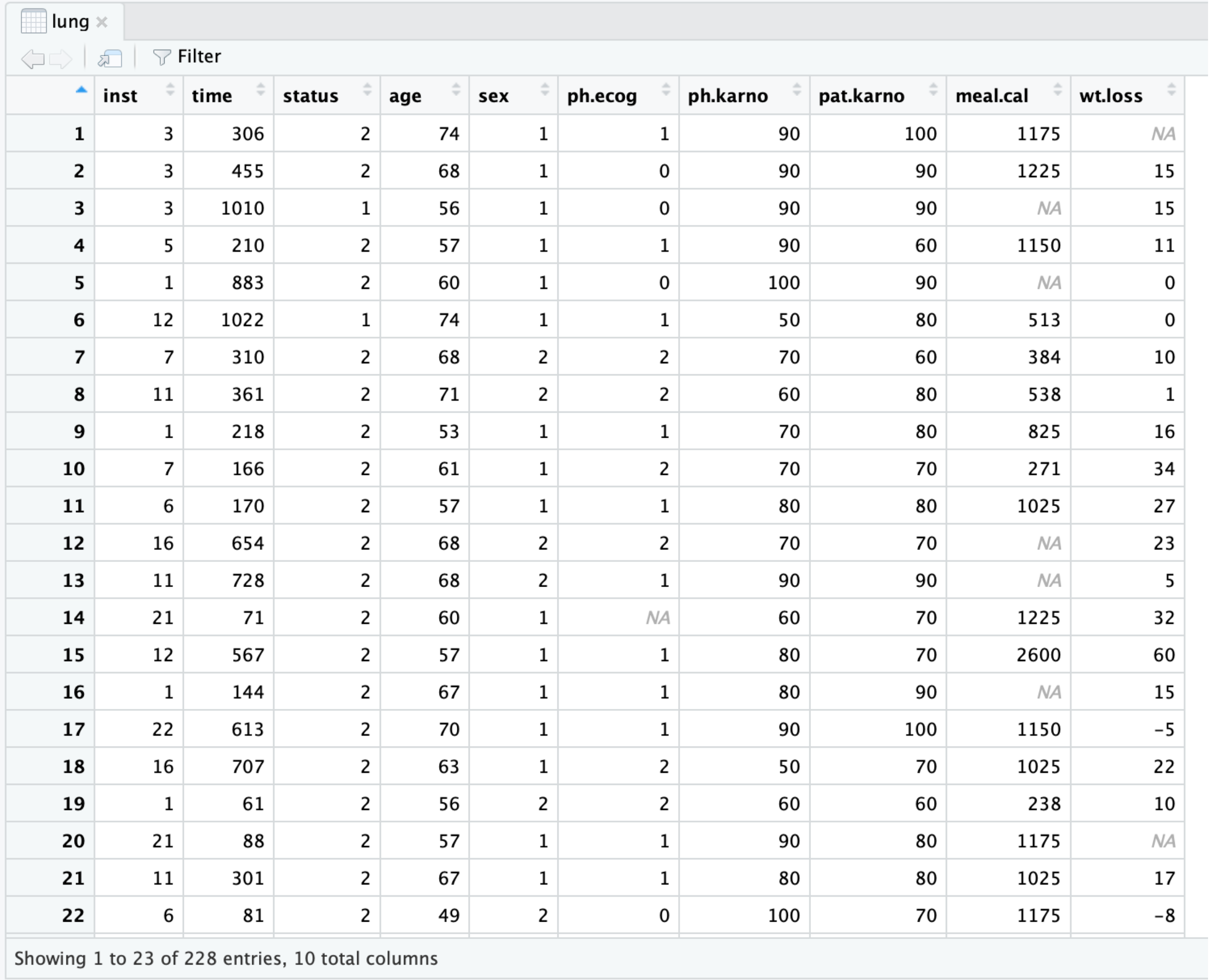
1. Learn the data: find what the data presents using simple methods of interrogation
2. look for structural errors (data types)
3. look for data irregularities (values)
4. decide how to deal with missing values (remove, substitute, impute, etc.)
5. keep records of changes you make

The lung cancer data set

inst:	Institution code
time:	Survival time in days
status:	censoring status 1=censored, 2=dead
age:	Age in years
sex:	Male=1 Female=2
ph.ecog:	ECOG performance score as rated by the physician. 0=asymptomatic, 1= symptomatic but completely ambulatory, 2= in bed <50% of the day, 3= in bed > 50% of the day but not bedbound, 4 = bedbound
ph.karno:	Karnofsky performance score (bad=0-good=100) rated by physician
pat.karno:	Karnofsky performance score as rated by patient
meal.cal:	Calories consumed at meals
wt.loss:	Weight loss in last six months (pounds)

- Describes survival in patients with advanced lung cancer.
- To load the data:

```
## first install the epi7913A package
lung <- epi7913::lung
```

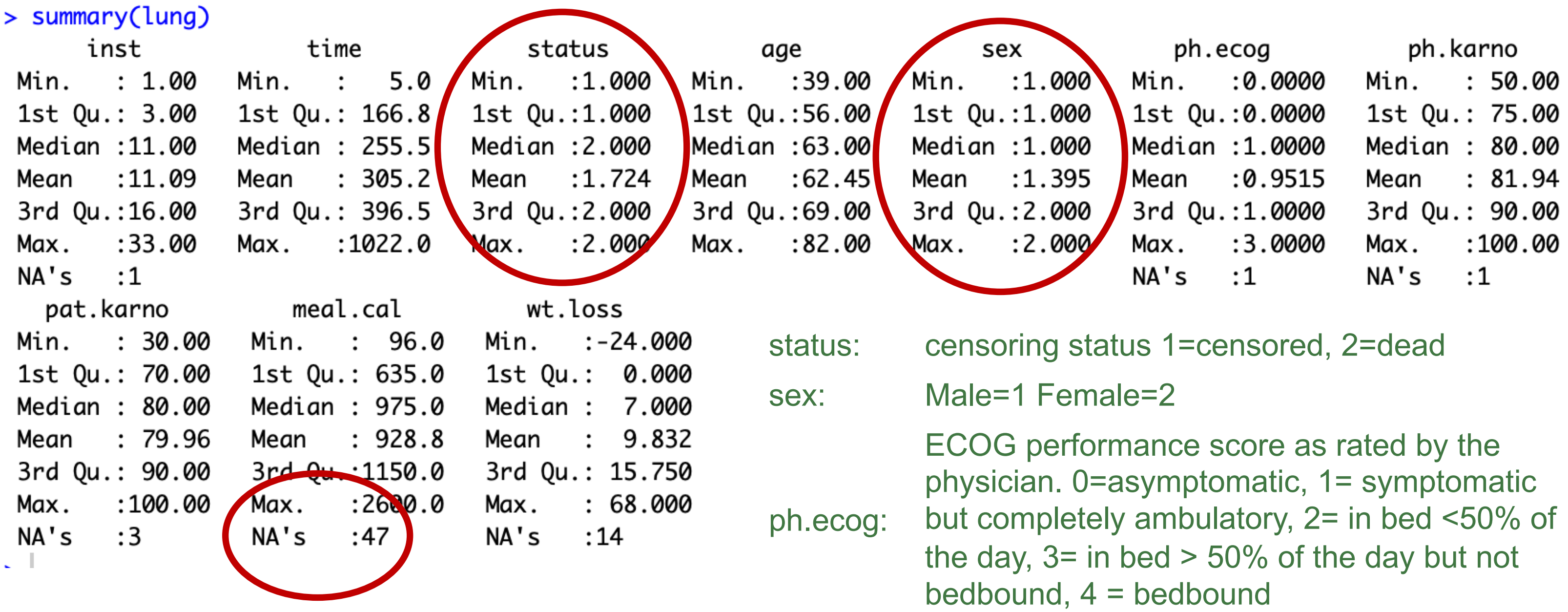


	inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss
1	3	306	2	74	1	1	90	100	1175	NA
2	3	455	2	68	1	0	90	90	1225	15
3	3	1010	1	56	1	0	90	90	NA	15
4	5	210	2	57	1	1	90	60	1150	11
5	1	883	2	60	1	0	100	90	NA	0
6	12	1022	1	74	1	1	50	80	513	0
7	7	310	2	68	2	2	70	60	384	10
8	11	361	2	71	2	2	60	80	538	1
9	1	218	2	53	1	1	70	80	825	16
10	7	166	2	61	1	2	70	70	271	34
11	6	170	2	57	1	1	80	80	1025	27
12	16	654	2	68	2	2	70	70	NA	23
13	11	728	2	68	2	1	90	90	NA	5
14	21	71	2	60	1	NA	60	70	1225	32
15	12	567	2	57	1	1	80	70	2600	60
16	1	144	2	67	1	1	80	90	NA	15
17	22	613	2	70	1	1	90	100	1150	-5
18	16	707	2	63	1	2	50	70	1025	22
19	1	61	2	56	2	2	60	60	238	10
20	21	88	2	57	1	1	90	80	1175	NA
21	11	301	2	67	1	1	80	80	1025	17
22	6	81	2	49	2	0	100	70	1175	-8

Showing 1 to 23 of 228 entries, 10 total columns

```
View(lung)
head(lung)
str(lung)
```

Simple summary statistics



```
> summary(lung)
      inst            time            status            age             sex            ph.ecog           ph.karno     
 Min.   : 1.00   Min.   :   5.0   Min.   :1.000   Min.   :39.00   Min.   :1.000   Min.   :0.0000   Min.   : 50.00  
 1st Qu.: 3.00   1st Qu.: 166.8   1st Qu.:1.000   1st Qu.:56.00   1st Qu.:1.000   1st Qu.:0.0000   1st Qu.: 75.00  
 Median :11.00   Median : 255.5   Median :2.000   Median :63.00   Median :1.000   Median :1.0000   Median : 80.00  
 Mean   :11.09   Mean   : 305.2   Mean   :1.724   Mean   :62.45   Mean   :1.395   Mean   :0.9515   Mean   : 81.94  
 3rd Qu.:16.00   3rd Qu.: 396.5   3rd Qu.:2.000   3rd Qu.:69.00   3rd Qu.:2.000   3rd Qu.:1.0000   3rd Qu.: 90.00  
 Max.   :33.00   Max.   :1022.0   Max.   :2.000   Max.   :82.00   Max.   :2.000   Max.   :3.0000   Max.   :100.00  
 NA's   :1                                                                        NA's   :1        NA's   :1       
   pat.karno         meal.cal         wt.loss       
 Min.   : 30.00   Min.   :  96.0   Min.   :-24.000  
 1st Qu.: 70.00   1st Qu.: 635.0   1st Qu.:  0.000  
 Median : 80.00   Median : 975.0   Median :  7.000  
 Mean   : 79.96   Mean   : 928.8   Mean   :  9.832  
 3rd Qu.: 90.00   3rd Qu.:1150.0   3rd Qu.: 15.750  
 Max.   :100.00   Max.   :2600.0   Max.   : 68.000  
 NA's   :3        NA's   :47       NA's   :14       
```

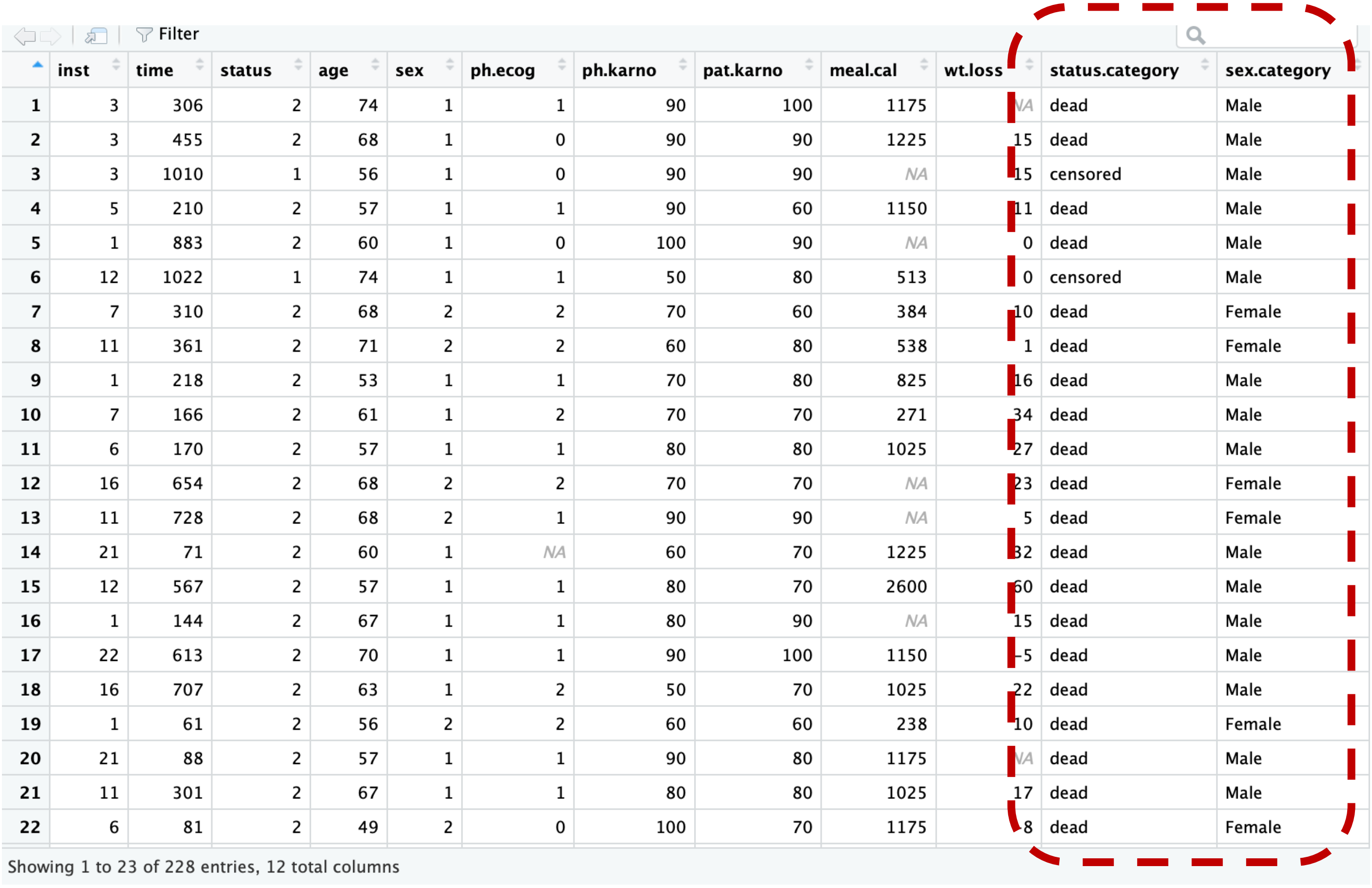
status: censoring status 1=censored, 2=dead

sex: Male=1 Female=2

ph.ecog: ECOG performance score as rated by the physician. 0=asymptomatic, 1= symptomatic but completely ambulatory, 2= in bed <50% of the day, 3= in bed > 50% of the day but not bedbound, 4 = bedbound

ph.karno: Karnofsky performance score (bad=0-good=100) rated by physician

- Status is a categorical variable that is encoded by a numeric value
- Sex also needs to be converted to categorical values
- meal.cal has the most missing values
- a balanced distribution is evident in continuous data when mean is close to median (time?)



	inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss	status.category	sex.category
1	3	306	2	74	1	1	90	100	1175	NA	dead	Male
2	3	455	2	68	1	0	90	90	1225	15	dead	Male
3	3	1010	1	56	1	0	90	90	NA	15	censored	Male
4	5	210	2	57	1	1	90	60	1150	11	dead	Male
5	1	883	2	60	1	0	100	90	NA	0	dead	Male
6	12	1022	1	74	1	1	50	80	513	0	censored	Male
7	7	310	2	68	2	2	70	60	384	10	dead	Female
8	11	361	2	71	2	2	60	80	538	1	dead	Female
9	1	218	2	53	1	1	70	80	825	16	dead	Male
10	7	166	2	61	1	2	70	70	271	34	dead	Male
11	6	170	2	57	1	1	80	80	1025	27	dead	Male
12	16	654	2	68	2	2	70	70	NA	23	dead	Female
13	11	728	2	68	2	1	90	90	NA	5	dead	Female
14	21	71	2	60	1	NA	60	70	1225	32	dead	Male
15	12	567	2	57	1	1	80	70	2600	60	dead	Male
16	1	144	2	67	1	1	80	90	NA	15	dead	Male
17	22	613	2	70	1	1	90	100	1150	-5	dead	Male
18	16	707	2	63	1	2	50	70	1025	22	dead	Male
19	1	61	2	56	2	2	60	60	238	10	dead	Female
20	21	88	2	57	1	1	90	80	1175	NA	dead	Male
21	11	301	2	67	1	1	80	80	1025	17	dead	Male
22	6	81	2	49	2	0	100	70	1175	8	dead	Female

Showing 1 to 23 of 228 entries, 12 total columns

Convert the numeric values with categorical values?

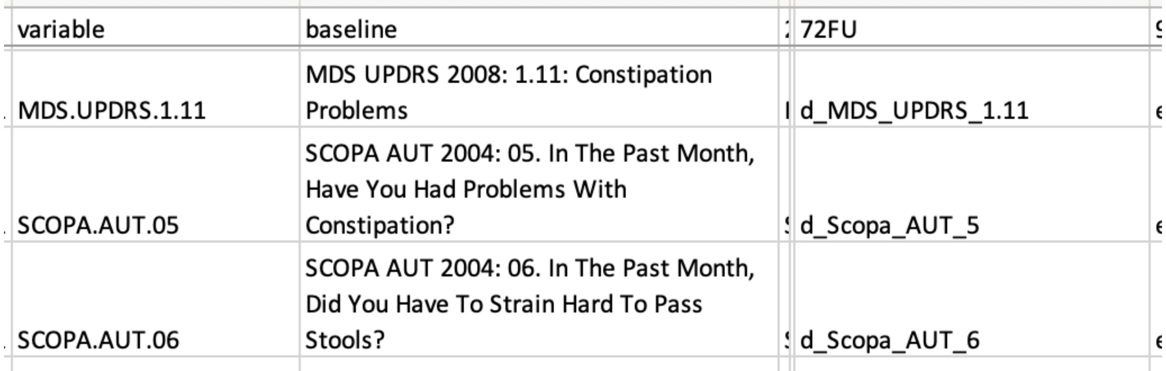
```
lung <- within(lung, { status.category <- NA; #initialize a new column
    status.category[status==1] <- "censored"
    status.category[status==2] <- "dead" } )

lung <- within(lung, { sex.category <- NA
      sex.category[sex==1] <- "Male"
      sex.category[sex==2] <- "Female" } )
```

A few useful examples

- check column names:
  - are the names intuitive (easy to understand)?
  - do they follow a convention? What is it?
  - change them as you see fit
  - for example, is `lung$time` intuitive to represent “Survival”?

```
# Modify Column names - these are case sensitive!
names(lung) < c("Inst","Time","Status","Age","Sex","PH.ecog","PH.karno","PAT.karno",
                "Meal.cal","WT.loss","Status.category","Sex.category")

# change a specific column (col 2) name
colnames(lung)[2] <- "time"
```

  - consider using a separate csv file like the variable sheet



variable	baseline	72FU
MDS.UPDRS.1.11	MDS UPDRS 2008: 1.11: Constipation Problems	d_MDS_UPDRS_1.11
SCOPA.AUT.05	SCOPA AUT 2004: 05. In The Past Month, Have You Had Problems With Constipation?	d_Scopa_AUT_5
SCOPA.AUT.06	SCOPA AUT 2004: 06. In The Past Month, Did You Have To Strain Hard To Pass Stools?	d_Scopa_AUT_6

A few useful examples

- check column names:
  - are the names intuitive (easy to understand)?
  - do they follow a convention? What is it?
  - change them as you see fit
  - for example, is `lung$time` intuitive to represent “Survival”?

```
# Modify Column names - these are case sensitive!
names(lung) < c("Inst","Time","Status","Age","Sex","PH.ecog","PH.karno","PAT.karno",
                "Meal.cal","WT.loss","Status.category","Sex.category")

# change a specific column (col 2) name
colnames(lung)[2] <- "time"
```

- do the columns have proper data types?

```
# check the class of the data set (data types of the columns)
sapply(lung, class)

# alternatively
str(lung)
```

A few useful examples

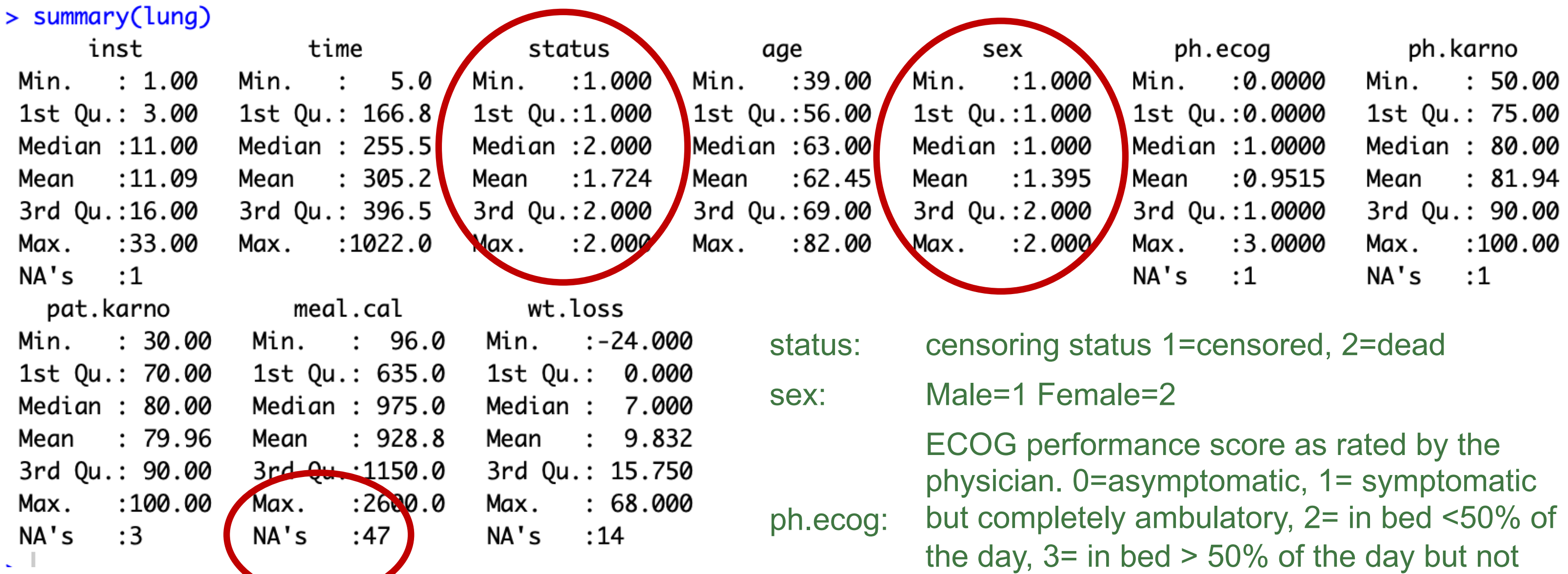
- Are there any missing values? If yes, what are they?

```
# Replace empty cells with NAs
lung[lung == ""] <- NA

is.na(lung) # returns TRUE or FALSE for each cell

which(is.na(lung)) # returns the positions in the entire data set

which(is.na(lung$time)) # returns the positions of NA in lung$time

sum(is.na(lung)) # get the count of how many NAs in the data
```

- How is “empty” represented? (“ ”, “9999”, 9999, NA, NIL)

```
# replace ONLY Surve.time with NA for those rows where time is 9999
lung[(lung$time == 9999)  & !is.na(lung$time),2] <- NA

# replace ONLY Surve.time with NA for those rows where time is “9999”
lung[(lung$time == “9999”)  & !is.na(lung$time),2] <- NA

# Replace negative "time" with NA
lung[lung$time < 0 !is.na(lung$time), 2] <- NA

# PLEASE BE CAEFUL!
# replace the ENTIRE ROW with NAs for those rows where time is 9999
lung[(lung$time == 9999)  & !is.na(lung$time),] <- NA
```

Simple summary statistics



```
> summary(lung)
      inst            time            status           age             sex           ph.ecog          ph.karno     
 Min.   : 1.00   Min.   :   5.0   Min.   :1.000   Min.   :39.00   Min.   :1.000   Min.   :0.0000   Min.   : 50.00  
 1st Qu.: 3.00   1st Qu.: 166.8   1st Qu.:1.000   1st Qu.:56.00   1st Qu.:1.000   1st Qu.:0.0000   1st Qu.: 75.00  
 Median :11.00   Median : 255.5   Median :2.000   Median :63.00   Median :1.000   Median :1.0000   Median : 80.00  
 Mean   :11.09   Mean   : 305.2   Mean   :1.724   Mean   :62.45   Mean   :1.395   Mean   :0.9515   Mean   : 81.94  
 3rd Qu.:16.00   3rd Qu.: 396.5   3rd Qu.:2.000   3rd Qu.:69.00   3rd Qu.:2.000   3rd Qu.:1.0000   3rd Qu.: 90.00  
 Max.   :33.00   Max.   :1022.0   Max.   :2.000   Max.   :82.00   Max.   :2.000   Max.   :3.0000   Max.   :100.00  
 NA's   :1                                                                        NA's   :1        NA's   :1       
   pat.karno         meal.cal         wt.loss       
 Min.   : 30.00   Min.   :  96.0   Min.   :-24.000  
 1st Qu.: 70.00   1st Qu.: 635.0   1st Qu.:  0.000  
 Median : 80.00   Median : 975.0   Median :  7.000  
 Mean   : 79.96   Mean   : 928.8   Mean   :  9.832  
 3rd Qu.: 90.00   3rd Qu.:1150.0   3rd Qu.: 15.750  
 Max.   :100.00   Max.   :2600.0   Max.   : 68.000  
 NA's   :3        NA's   :47       NA's   :14       
```

status: censoring status 1=censored, 2=dead

sex: Male=1 Female=2

ph.ecog: ECOG performance score as rated by the physician. 0=asymptomatic, 1= symptomatic but completely ambulatory, 2= in bed <50% of the day, 3= in bed > 50% of the day but not bedbound, 4 = bedbound

ph.karno: Karnofsky performance score (bad=0-good=100) rated by physician

- Status is a categorical variable that is encoded by a numeric value
- Sex also needs to be converted to categorical values
- meal.cal has the most missing values
- a balanced distribution is evident in continuous data when mean is close to median (time?)

A few useful examples

```
# try this …
str(lung)
lung[1,4] <- ”9999”
str(lung)


# get the mean of survival time (NAs will cause error)
mean(lung$time)

# get mean survival time by removing NAs
mean(lung$time, na.rm = TRUE)

# retrieve all rows with complete values (no NAs)
lung[complete.cases(lung),]

# retrieve all rows with incomplete values
lung[! complete.cases(lung),]
```

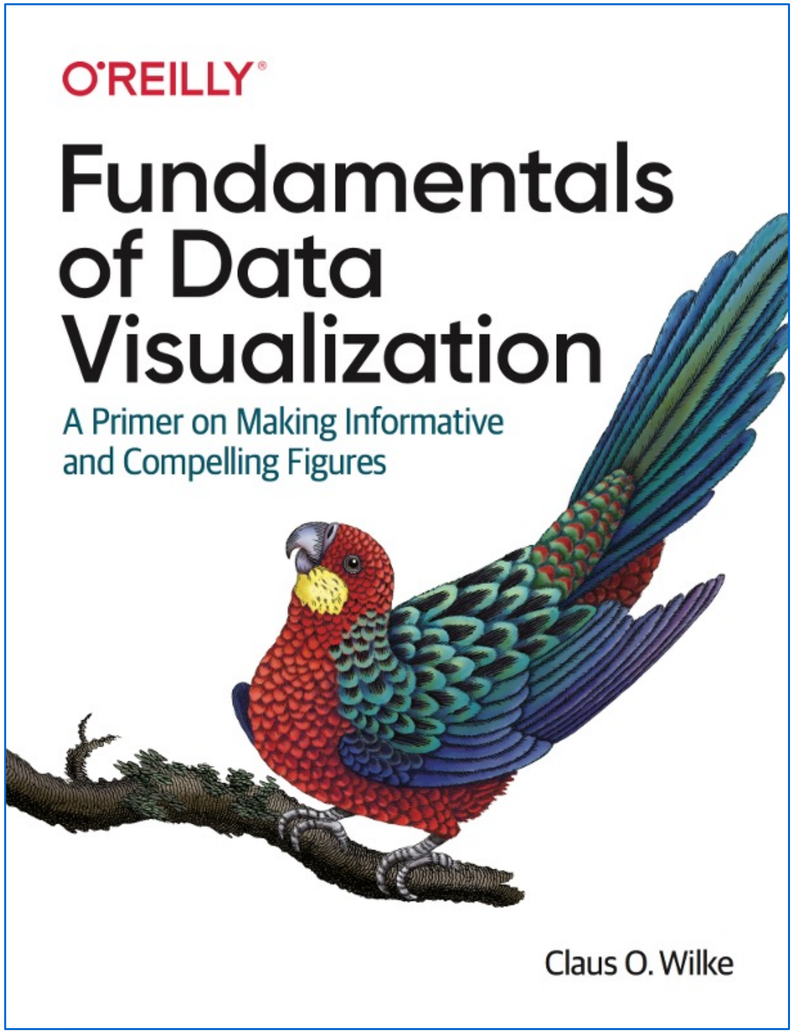
# load data from class package epi7913A for practice

```
epi7913A::lung_junk

epi7913A::lung_na
```

Available tools

- Univariate visualizations:
  - Categorical plots (bar plots, pie charts, tree maps)
  - Quantitative plots: (histograms, density functions, dot charts).
- Bivariate visualizations:
  - Categorical vs. categorical: (stacked and grouped bar plots)
  - Quantitative vs. Quantitative: (scatter plots, best fit)
  - Quantitative vs. categorical: (boxplots, line plots, other)
  - Multivariant visualizations: (grouping)
- Tabulations:
  - Frequencies and contingency tables
  - Tests of independence: (Chi-square test, Fisher exact test)

A useful book



O'REILLY

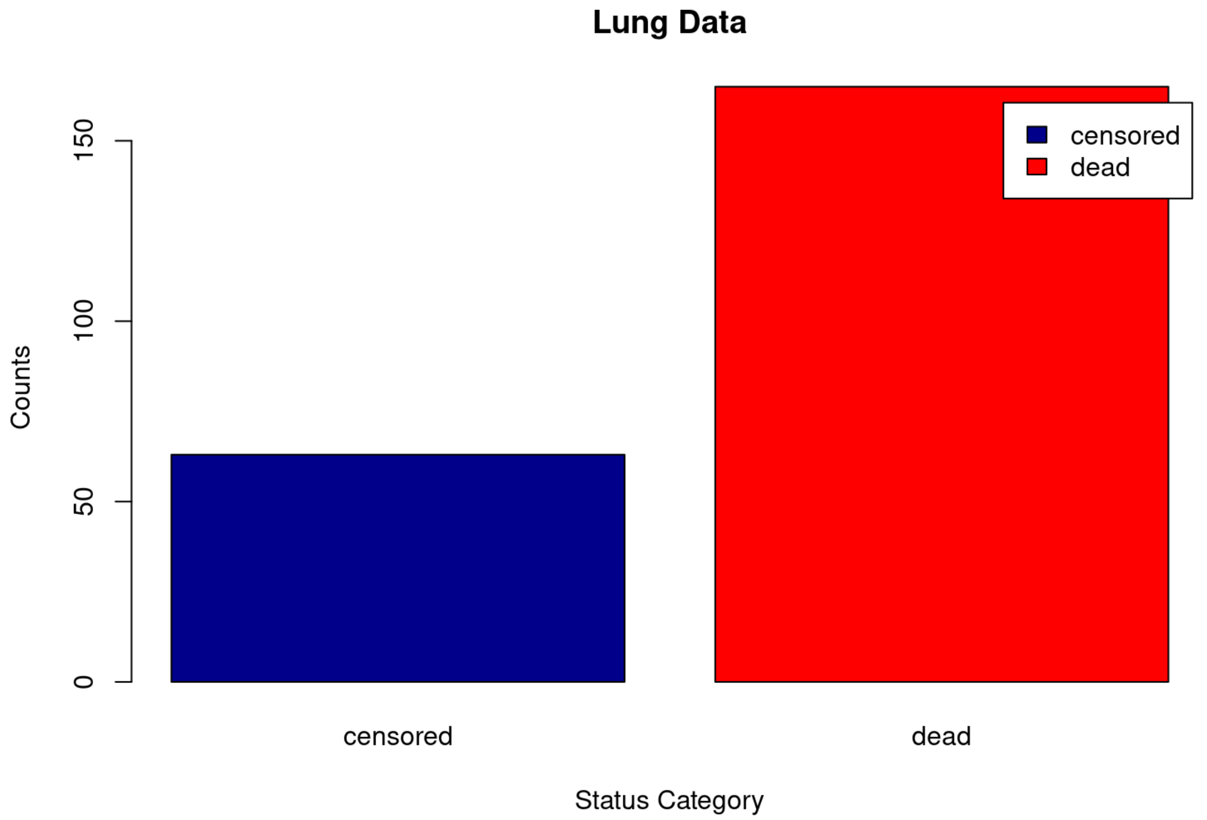
Fundamentals of Data Visualization

A Primer on Making Informative and Compelling Figures

Claus O. Wilke

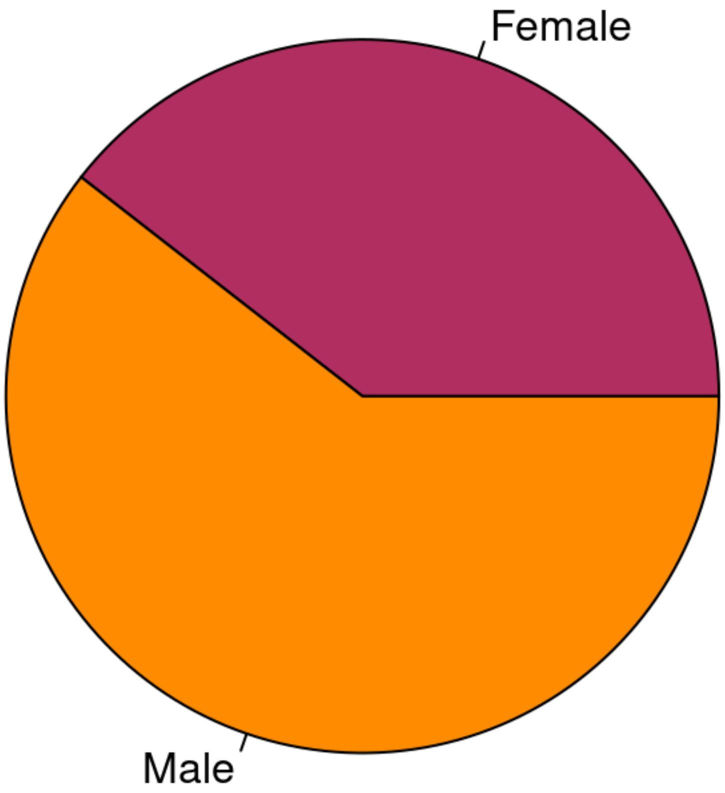
https://clauswilke.com/dataviz/

Univariate visualizations: frequency by bar plots



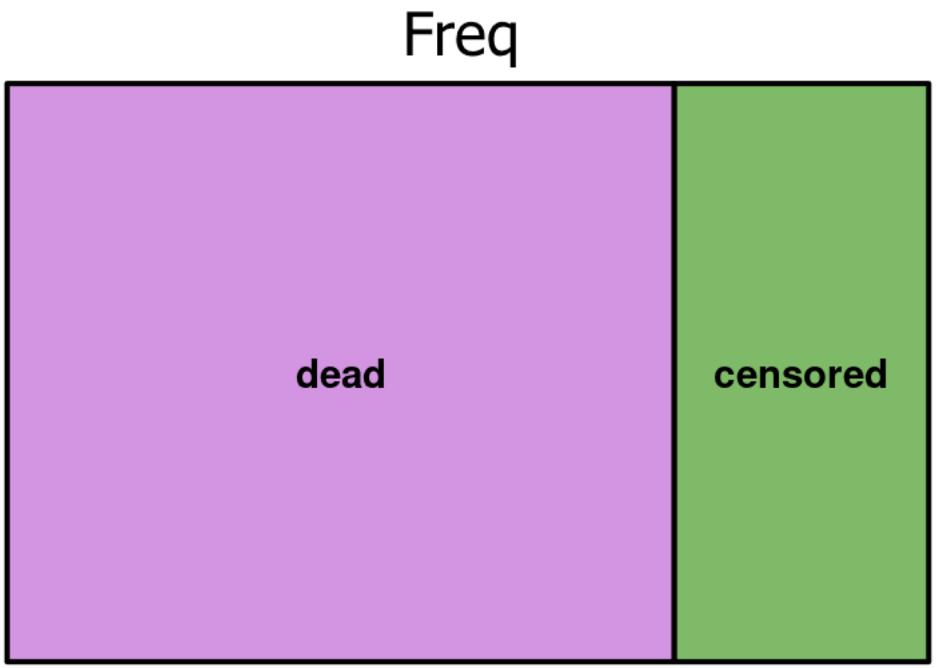
```
counts <- table(lung$status.category)
barplot(counts, main="Lung Data",
        xlab="Status Category", ylab="Counts",
        col=c("darkblue","red"), legend = rownames(counts))
```

Univariate visualizations: frequency in a pie chart

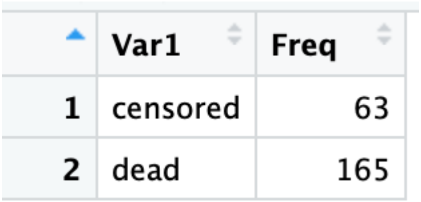


```
mytable <- table(lung$sex.category)
pie(mytable, col=c("maroon","darkorange"))
```

Univariate visualizations: frequency in a simple treemap (visualize proportions)



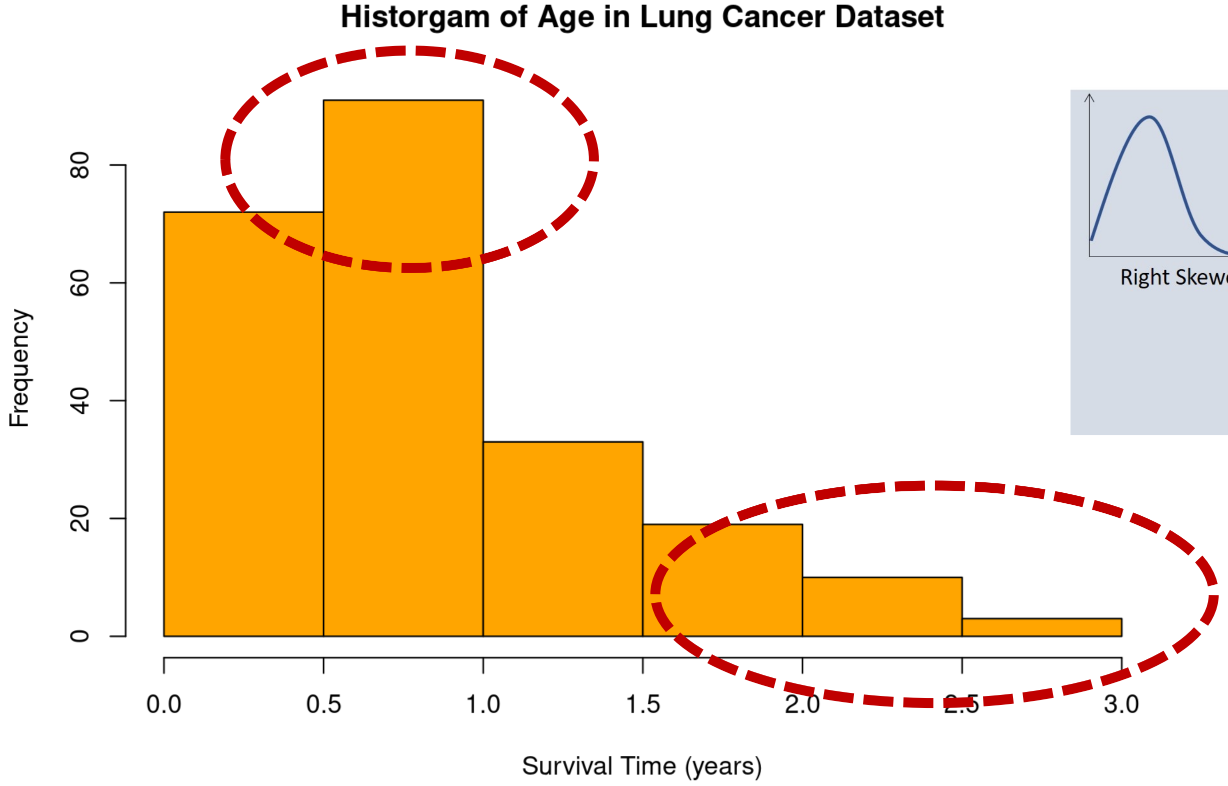
mytable

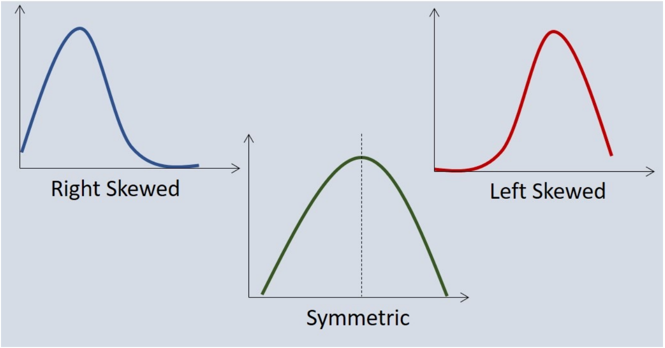


	Var1	Freq
1	censored	63
2	dead	165

```
mytable <- data.frame(table(lung$status.category))
treemap(mytable, index =c("Var1"), vSize = "Freq")
```

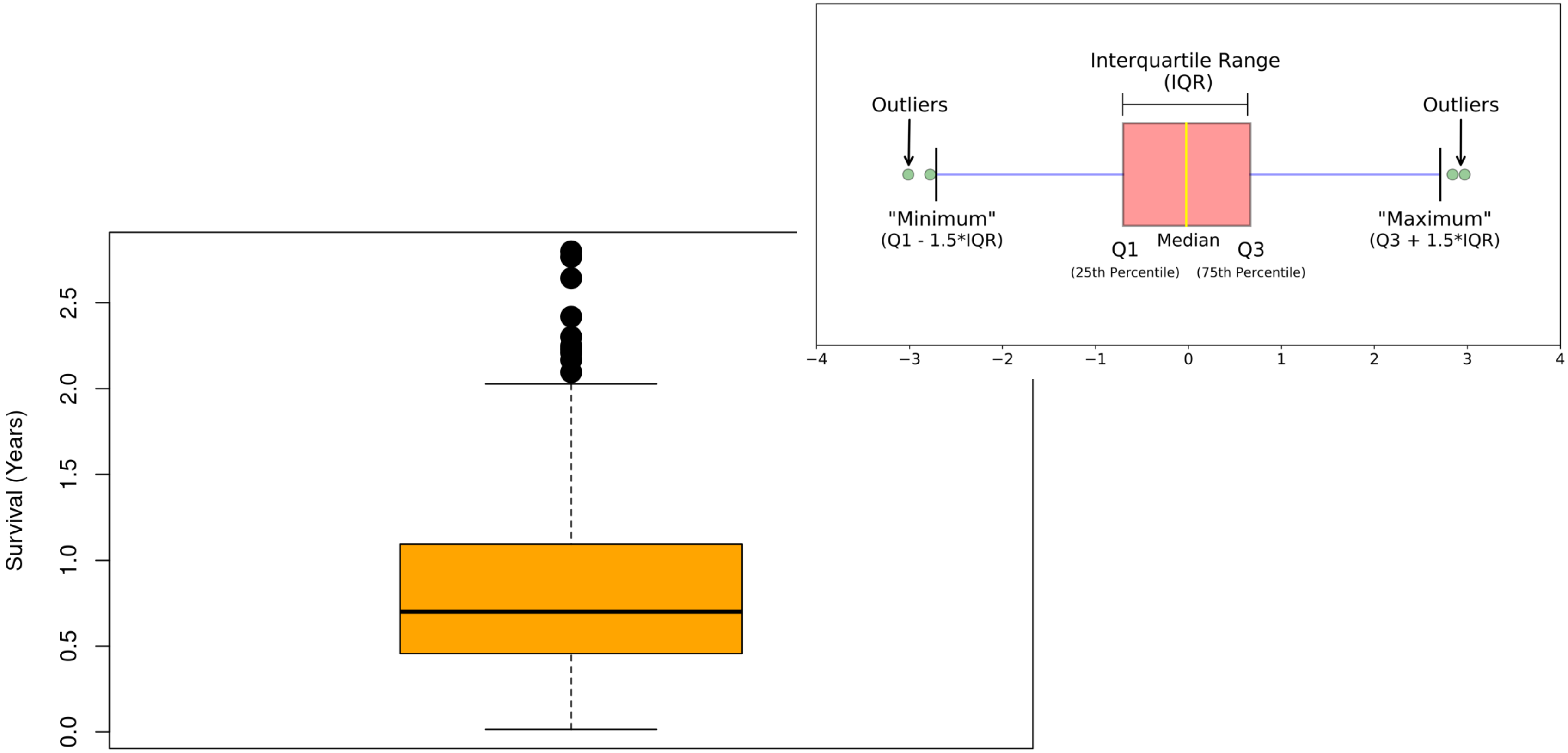
Univariate visualizations: histograms





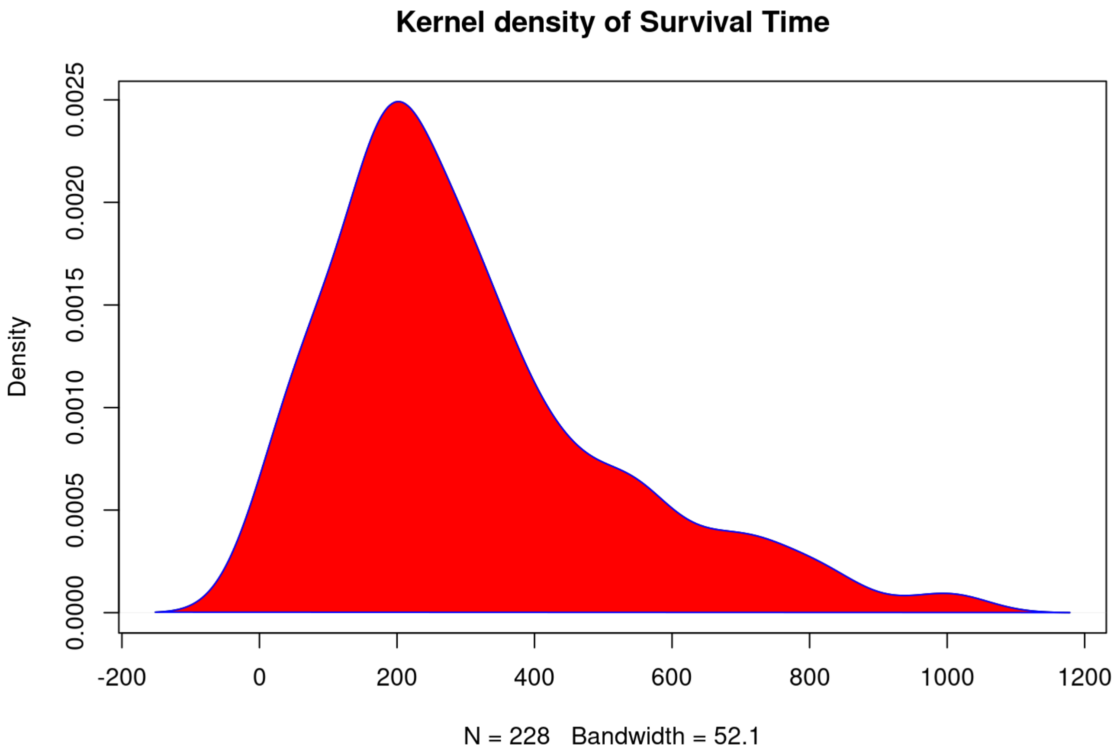
```
hist(lung$time/365, breaks=6, col="orange",
        main="Histogram of survival time in Lung Cancer Dataset",
        xlab="Survival Time (years)")
```

Univariate visualizations: box plots



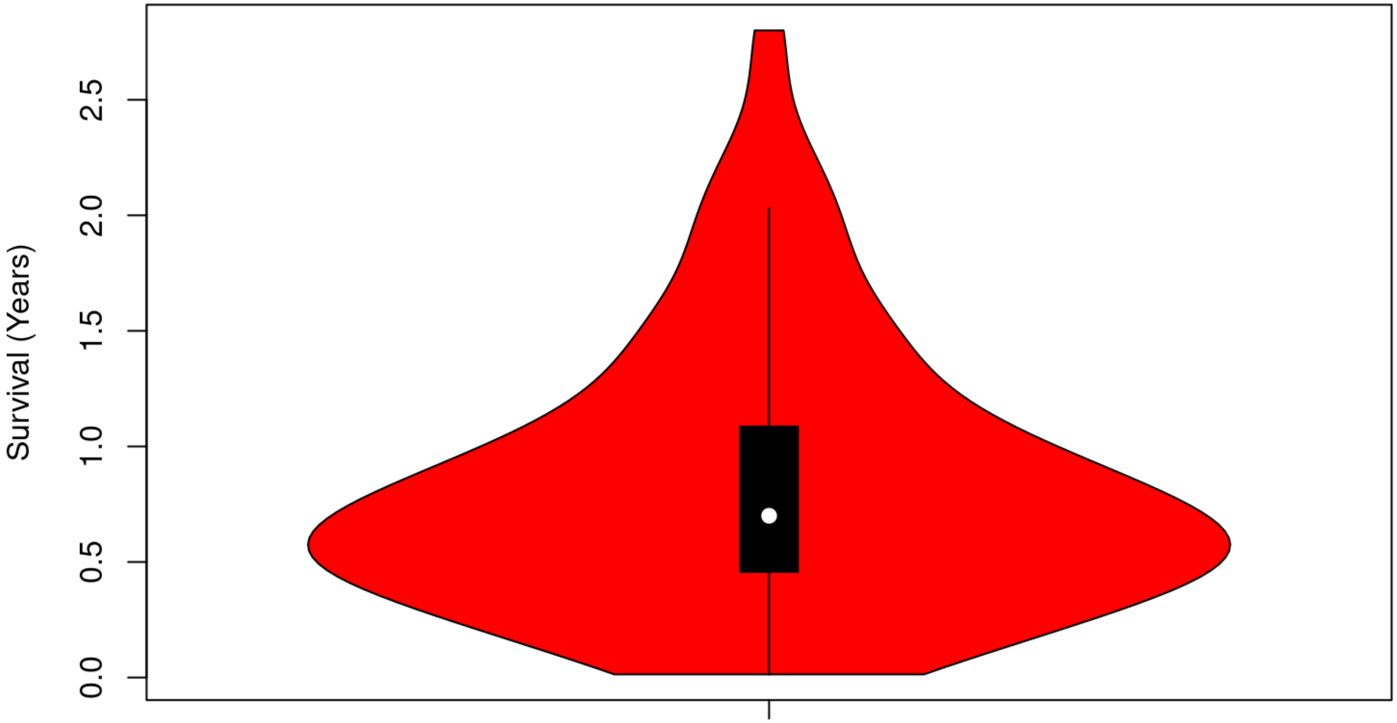
```
boxplot(lung$time/365, ylab="Survival (Years)",
        col="orange", pch=19, cex=2)
# Source: https://www.kdnuggets.com/2019/11/understanding-boxplots.html
```

Univariate visualizations: kernel density



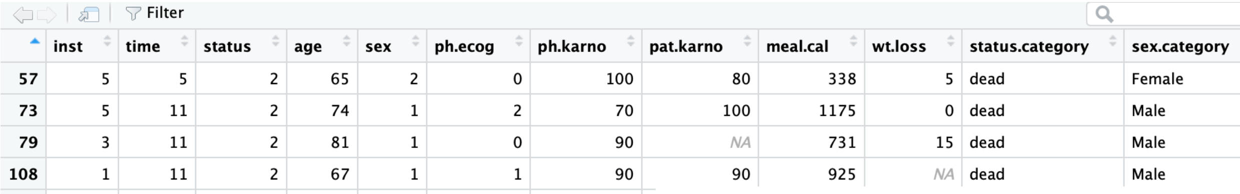
```
# kernel smoothing for probability density estimation
d <- density(lung$time)
plot(d, main="Kernel density of Survival Time")
polygon(d, col="red", border="blue")
```

Univariate visualizations: violin plots



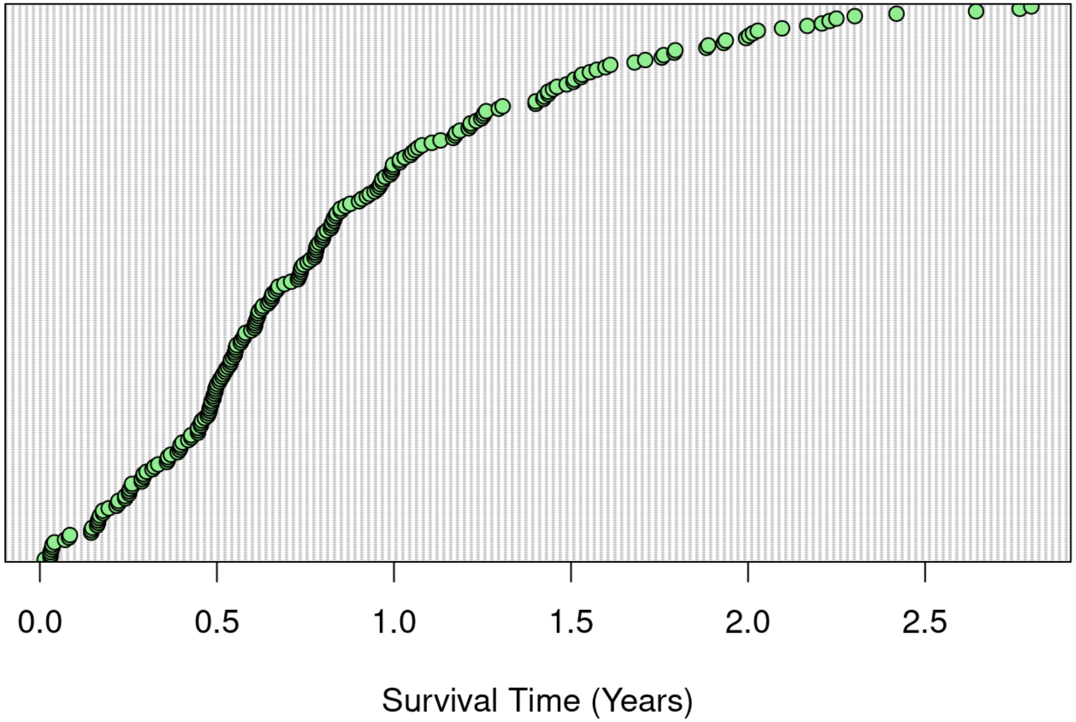
```
# a box plot with a rotated kernel density plot on either side
vioplot(lung$time/365, ylab="Survival (Years)", col="red")
```

Univariate visualizations: dot plots



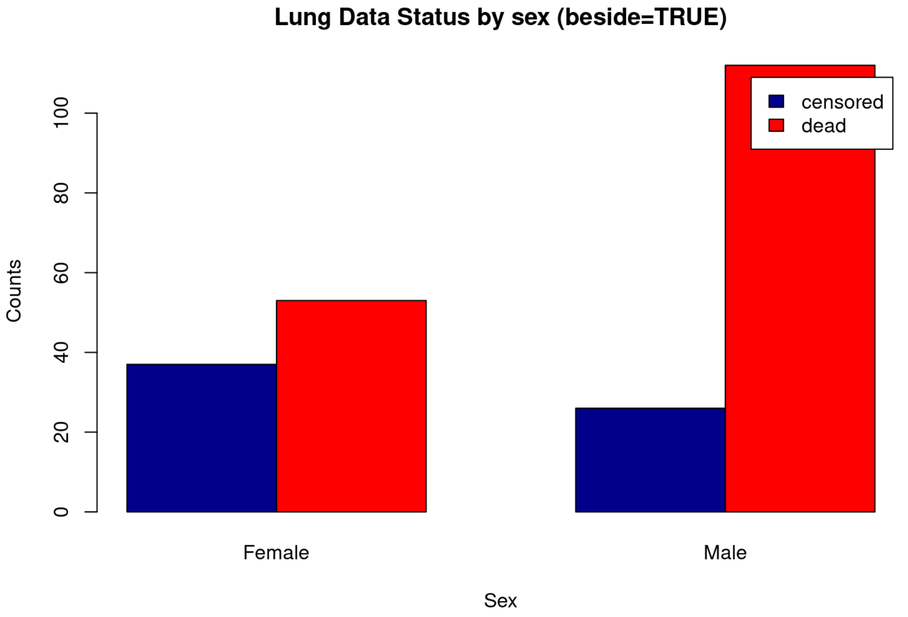
	inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss	status.category	sex.category
57	5	5	2	65	2	0	100	80	338	5	dead	Female
73	5	11	2	74	1	2	70	100	1175	0	dead	Male
79	3	11	2	81	1	0	90	NA	731	15	dead	Male
108	1	11	2	67	1	1	90	90	925	NA	dead	Male
30	1	12	2	74	1	2	70					
116	1	13	2	76	1	2	70					
215	11	13	2	65	1	1	80					
111	13	15	2	69	1	0	90					
32	1	26	2	73	1	2	60					
96	12	30	2	72	1	2	80					

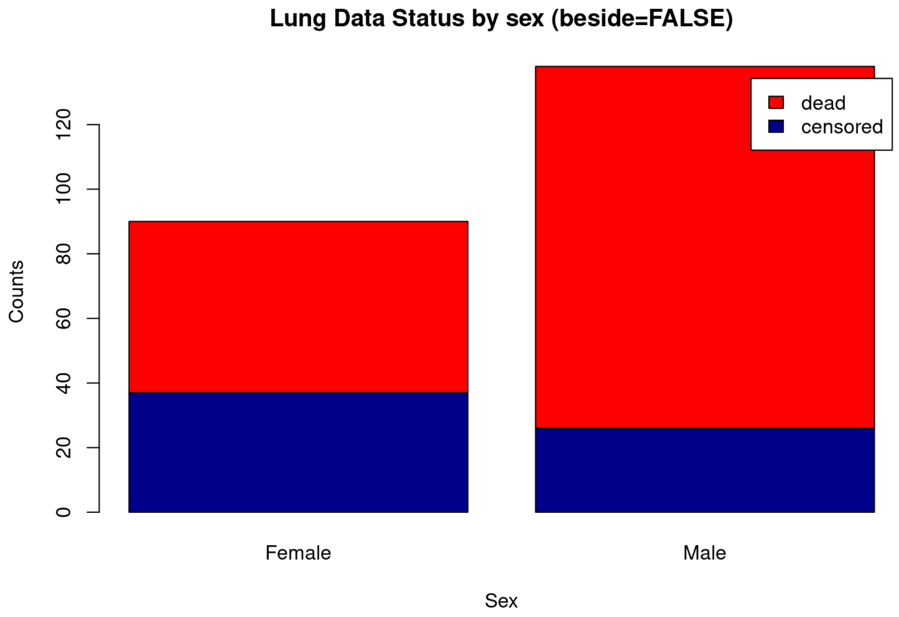
Showing 1 to 10 of 228 entries, 12 total columns



```
tmpData <- lung[order(lung$time), ]
dotchart(tmpData$time/365, label=NULL, cex=1.2,
       pch=21, bg="lightgreen", xlab="Survival Time (Years)")
View(tmpData) # can also use head()
```

Bivariate visualization: frequency per group (stacked vs groups)

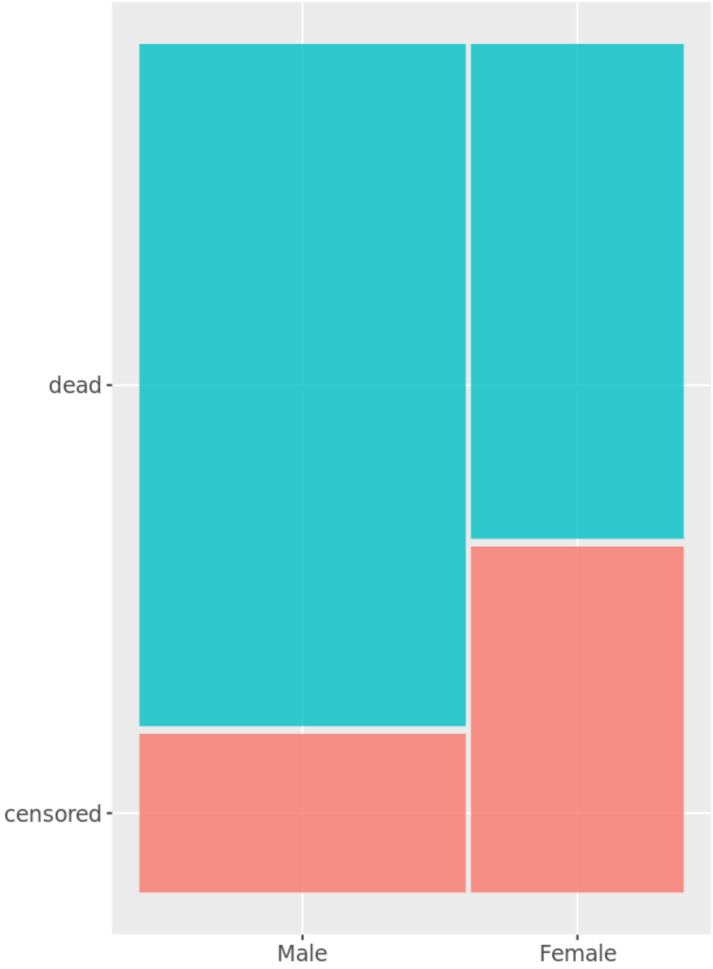

Lung Data Status by sex (beside=TRUE)


Lung Data Status by sex (beside=FALSE)

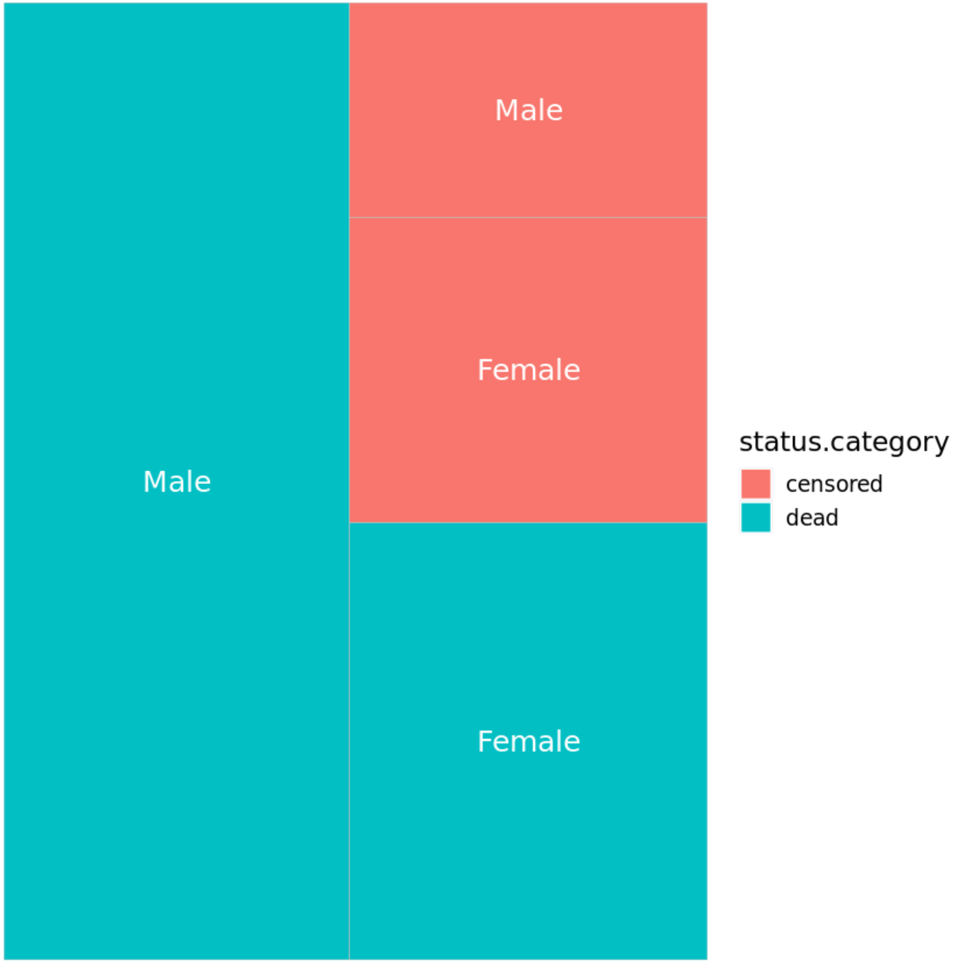
```
counts <- table(lung$status.category, lung$sex.category)
barplot(counts, main="Lung Data Status by sex", xlab="Sex",
        ylab="Counts", col=c("darkblue","red"),
        legend = rownames(counts), beside=TRUE)
```

Bivariate visualization

mosaic plot

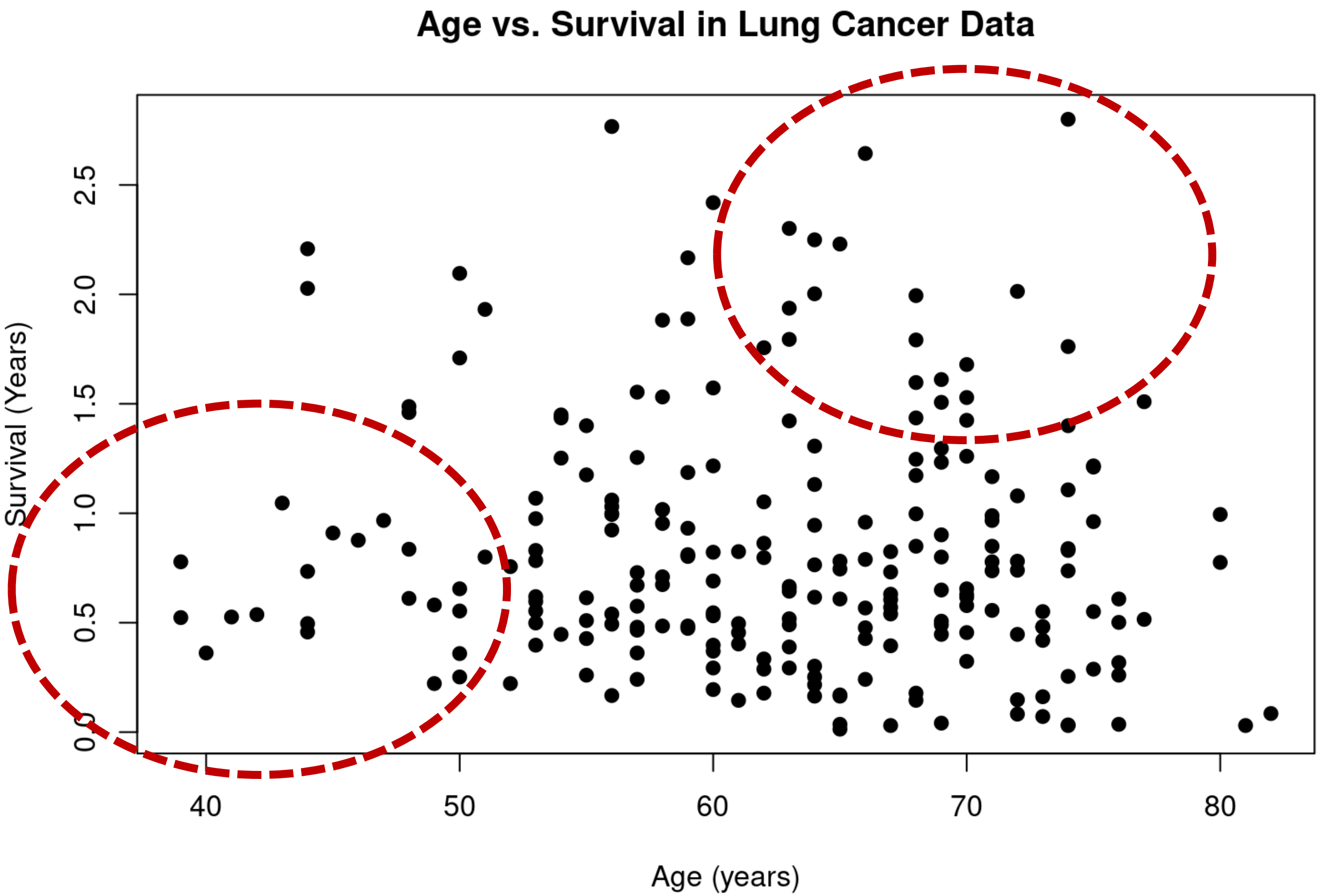


treemap



https://clauswilke.com/dataviz/nested-proportions.html

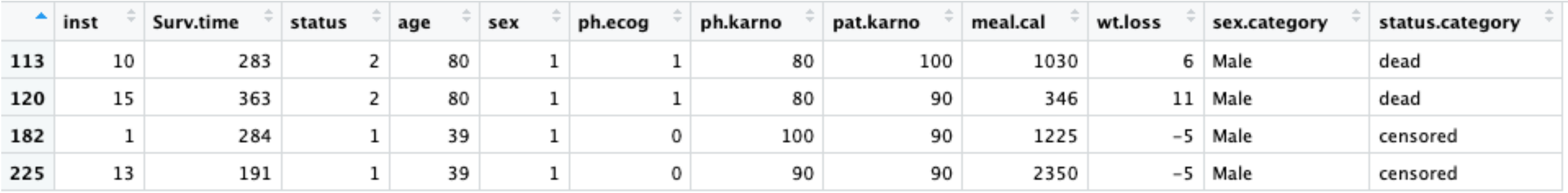
Bivariate visualization: scatter plots



```
plot(lung$age, lung$time/365, pch=19,
        main="Age vs. Survival in Lung Cancer Data",
        xlab="Age (years)", ylab="Survival (Years)")
lung[lung$time > 100 & lung$age >= 80,]
lung[lung$time > 100 & lung$age < 40,]
```

Filtering for extreme cases

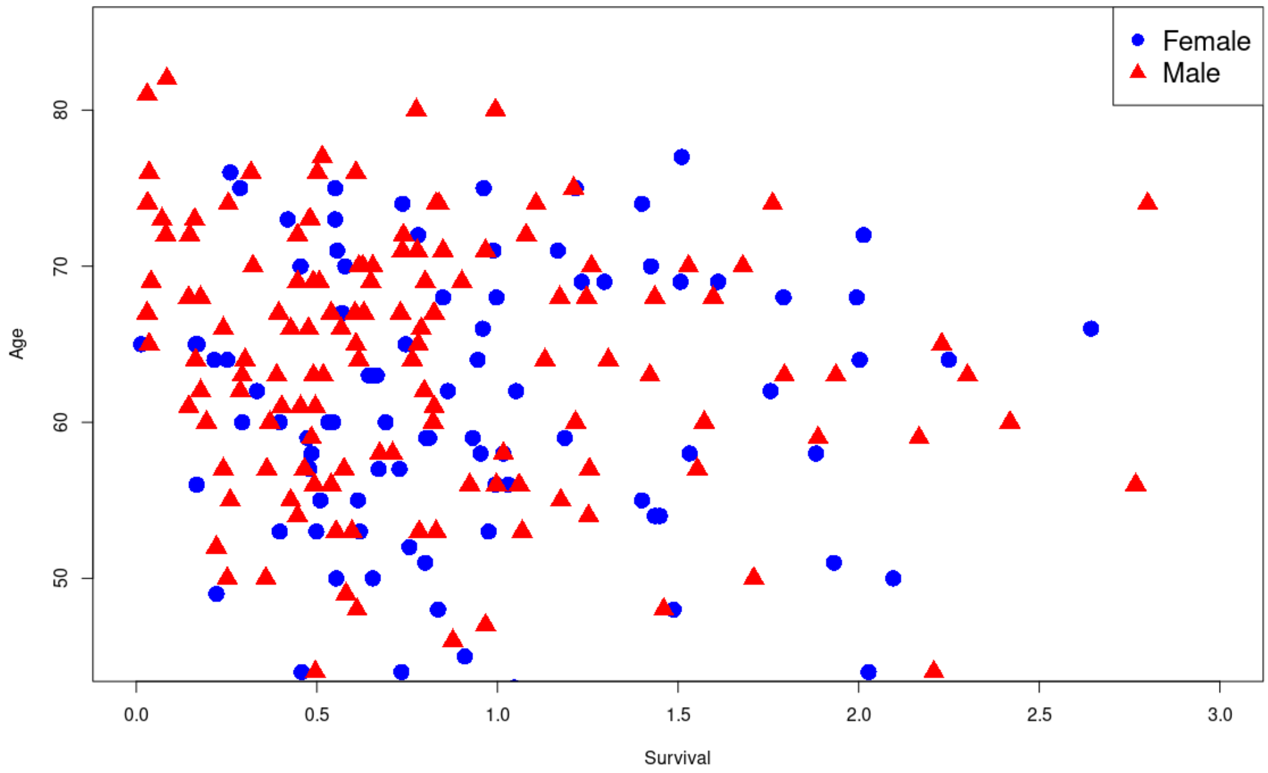
Tmp



	inst	Surv.time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss	sex.category	status.category
113	10	283	2	80	1	1	80	100	1030	6	Male	dead
120	15	363	2	80	1	1	80	90	346	11	Male	dead
182	1	284	1	39	1	0	100	90	1225	-5	Male	censored
225	13	191	1	39	1	0	90	90	2350	-5	Male	censored

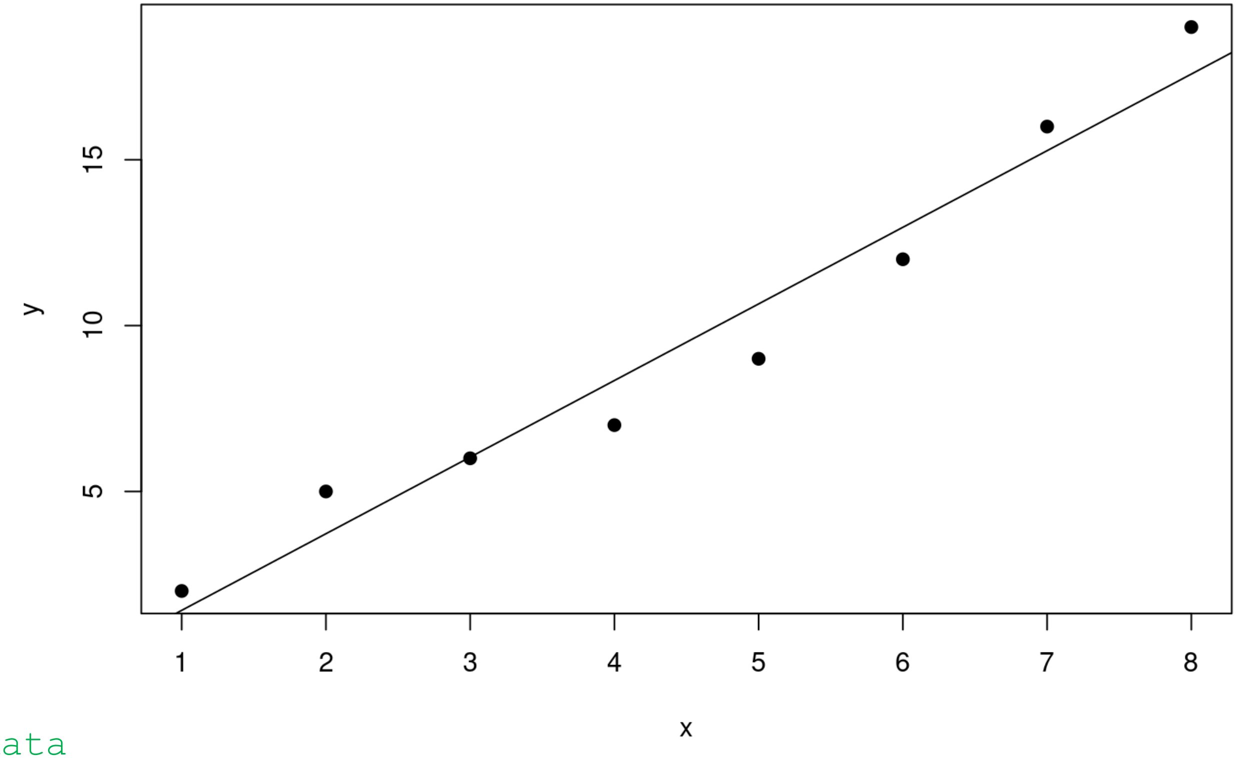
```
# Patients over 80 years old who survived more than 100 days
lung[lung$time > 100 & lung$age >= 80,]

# Patients younger than 40 years old who survived more than 100 days
lung[lung$time > 100 & lung$age < 40,]

# Filter both in one statement
Tmp <- lung[lung$time > 100 & (lung$age >= 80 | lung$age < 40),]
```

Bivariate visualization: scatter plots / group



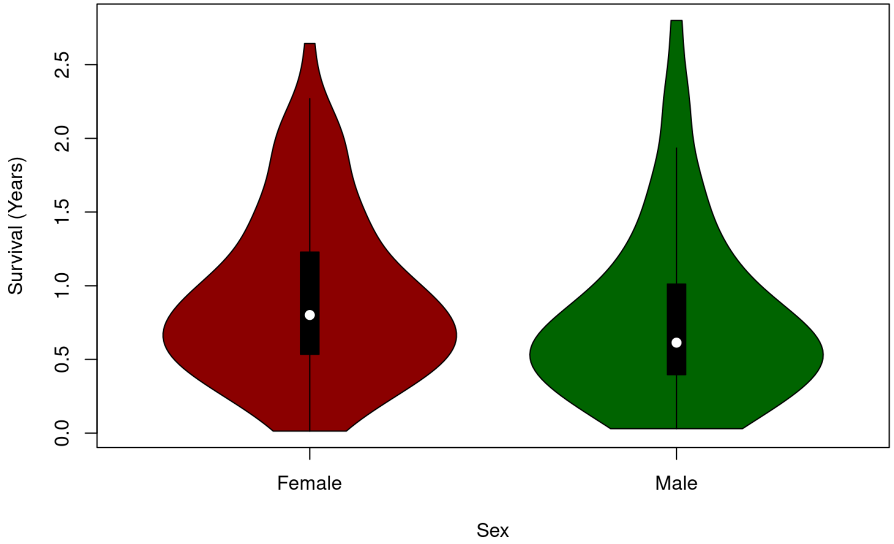
```
plot(lung[lung$sex.category == "Female", ]$time/365,
  xlim=c(0,3), ylim=c(45,85),
  lung[lung$sex.category == "Female", ]$age, pch=16, cex=2,
  col="blue", ylab="Age", xlab="Survival")
points(lung[lung$sex.category == "Male", ]$time/365,
  lung[lung$sex.category == "Male", ]$age, pch=17, col="red", cex=2)
legend("topright",c("Female", "Male"),
  col=c("blue","red"),pch=c(16,17), cex=1.5)
```

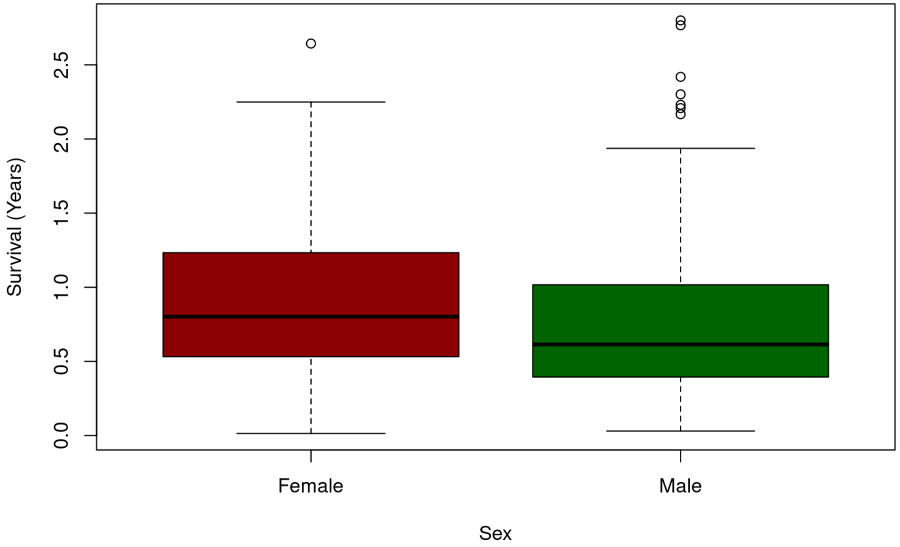
Bivariate visualization: best fit line



```
# makeup data
x <- c(1, 2, 3, 4, 5, 6, 7, 8)
y <- c(2, 5, 6, 7, 9, 12, 16, 19)

#create scatter plot of x vs. y
plot(x, y, pch=19)

#add line of best fit to scatter plot
abline(lm(y ~ x))
```

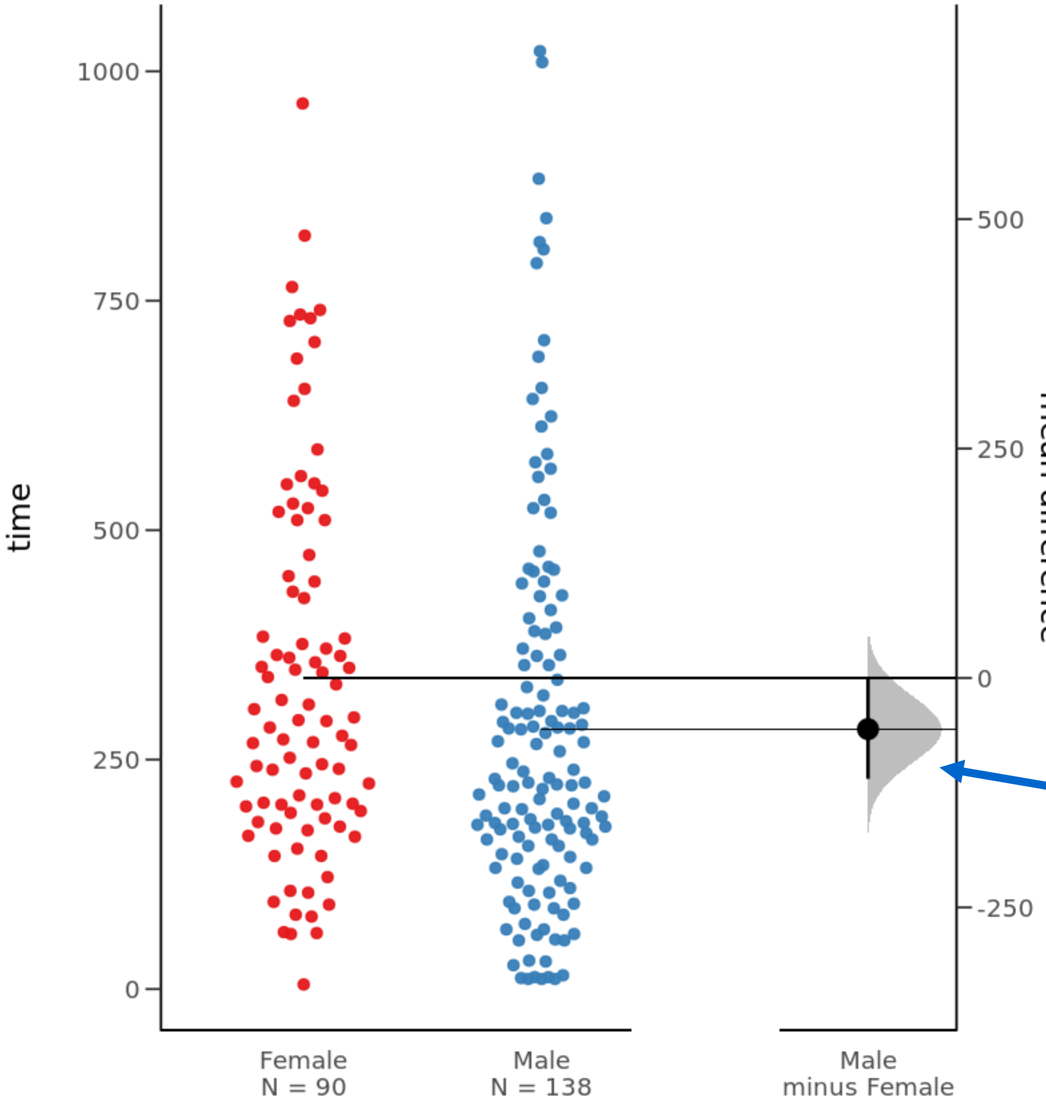
Bivariate visualization: survival / group (box plots & violin plots)





```
boxplot(lung$time/365 ~ lung$sex.category,
        col=c("darkred", "darkgreen"),
        main="Survivial Boxplot",
        xlab="Sex", ylab="Survival (Years)")

install.packages(“vioplot”)
library(vioplot)
vioplot(lung$time/365 ~ lung$sex.category,
        col=c("darkred", "darkgreen"),
        main="Survivial Density Plot",
        xlab="Sex", ylab="Survival (Years)”)
```

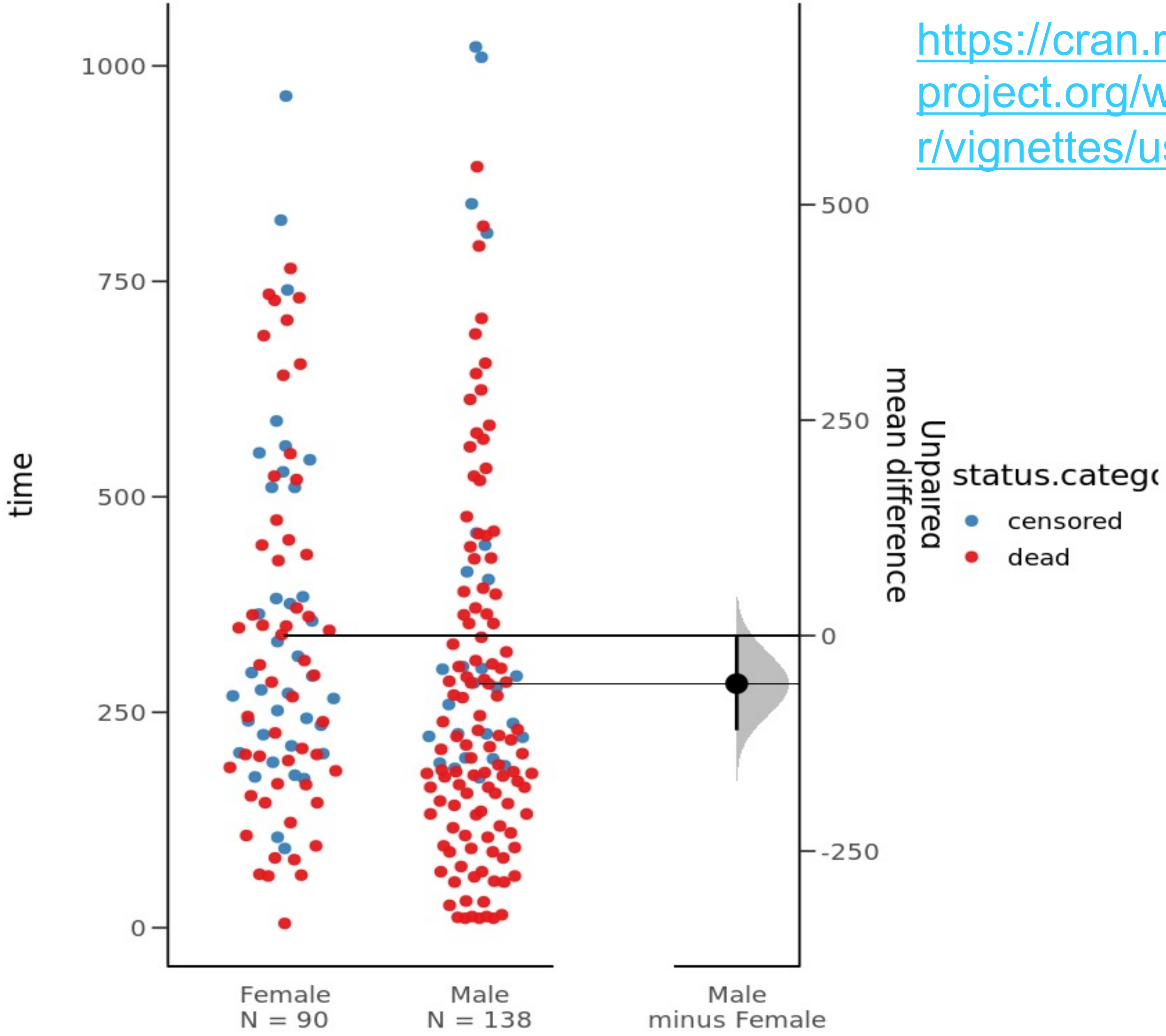
Bivariate visualization: survival / group (estimation plot)



https://cran.r-project.org/web/packages/dabestr/vignettes/using-dabestr.html

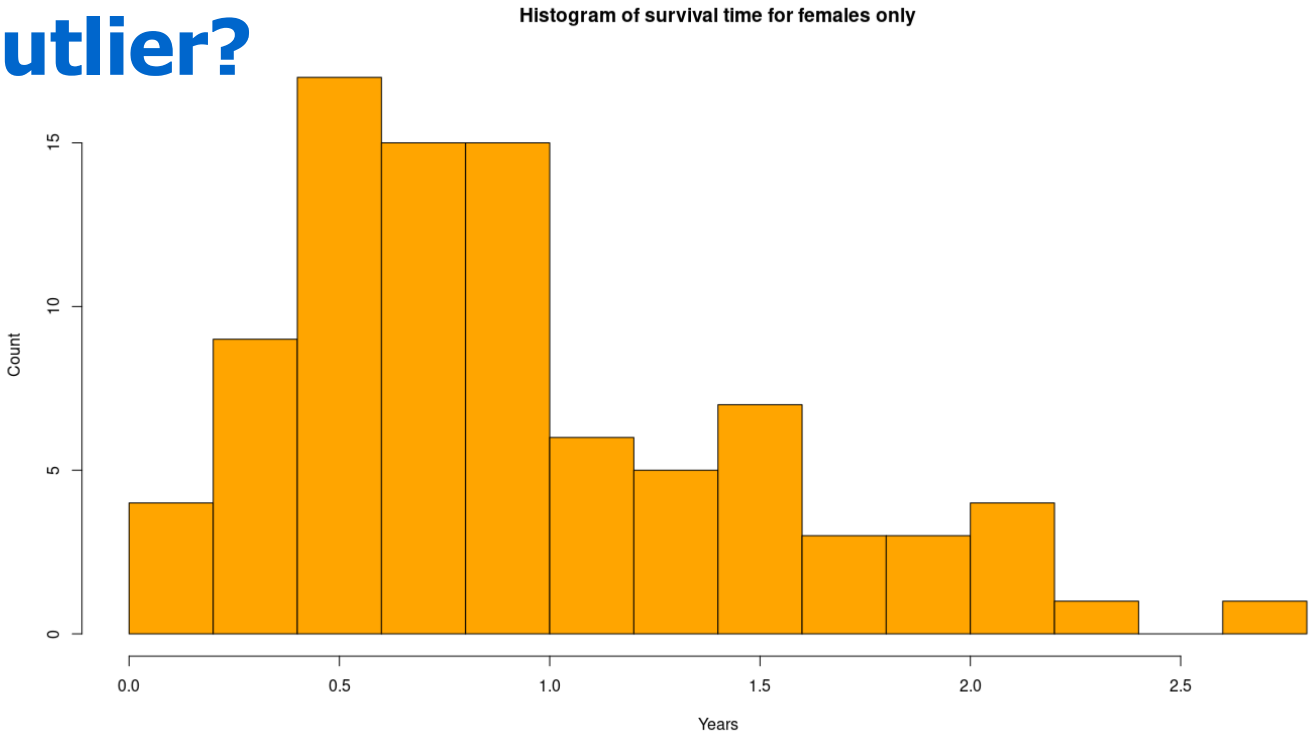
the 95% CI through nonparametric bootstrap resampling

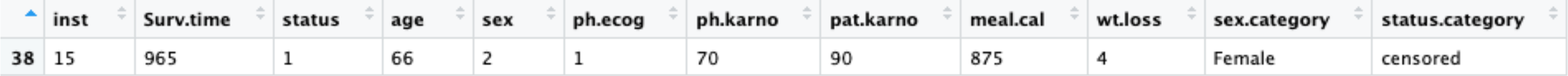
Bivariate visualization: survival / group (estimation plot)



https://cran.r-project.org/web/packages/dabestr/vignettes/using-dabestr.html

Outlier?

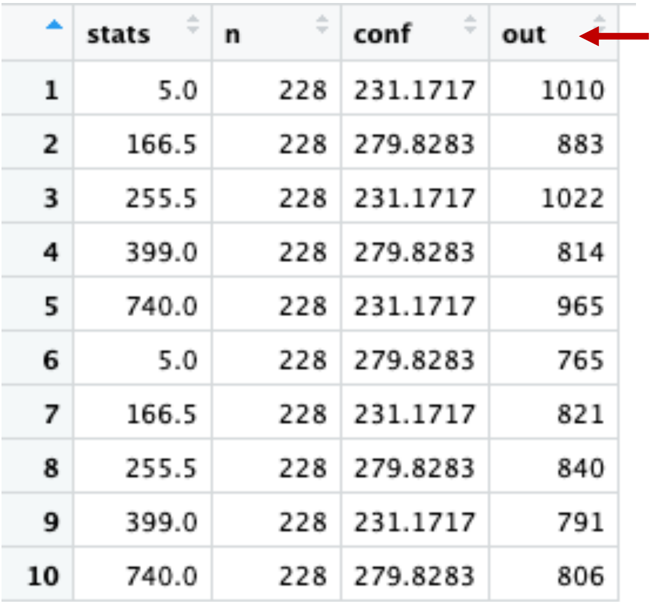




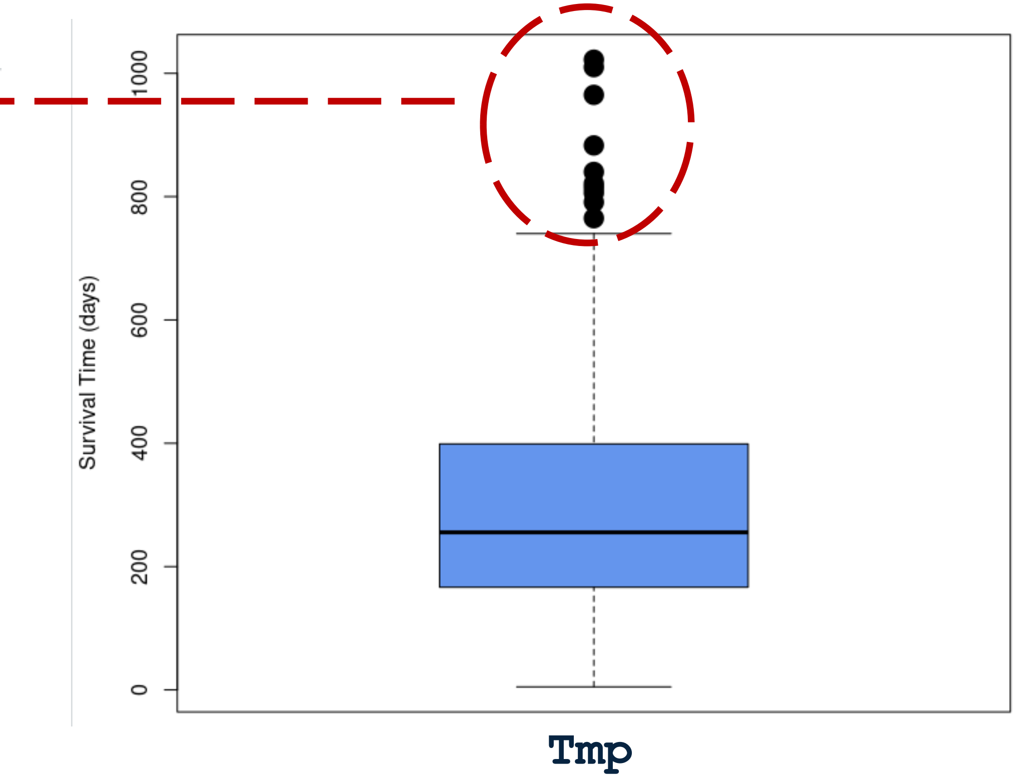
	inst	Surv.time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss	sex.category	status.category
38	15	965	1	66	2	1	70	90	875	4	Female	censored

```
# plot a histogram for survival only female patients
hist(lung[lung$sex.category == "Female",]$time/365,
     breaks=10, col="orange", xlab="Years", ylab="Count”,
     main="Histogram of survival time for females only")
# retrieve data for female patients who survived more than 2.5 years
lung[lung$sex.category == "Female"& lung$time/365 > 2.5,]
```

Outlier detection



	stats	n	conf	out
1	5.0	228	231.1717	1010
2	166.5	228	279.8283	883
3	255.5	228	231.1717	1022
4	399.0	228	279.8283	814
5	740.0	228	231.1717	965
6	5.0	228	279.8283	765
7	166.5	228	231.1717	821
8	255.5	228	279.8283	840
9	399.0	228	231.1717	791
10	740.0	228	279.8283	806

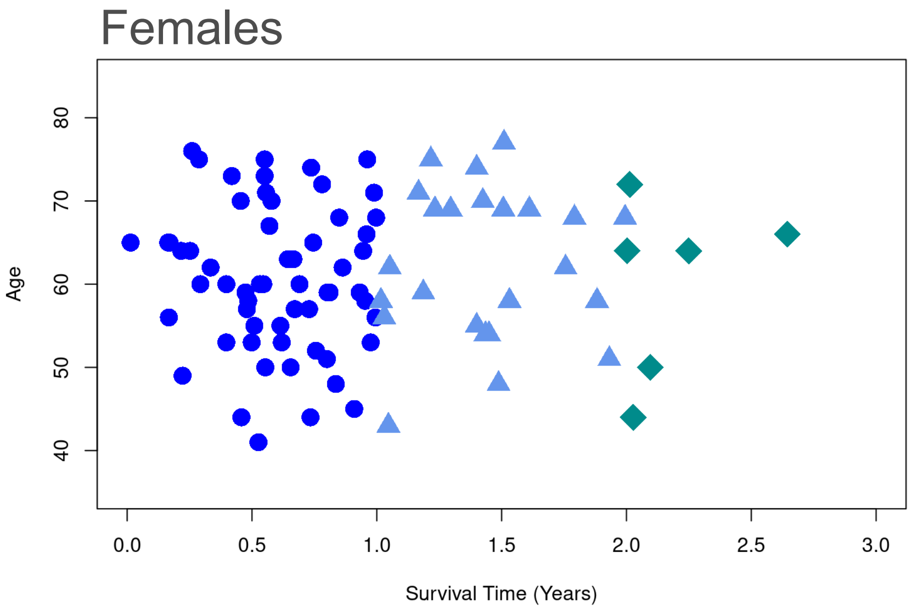


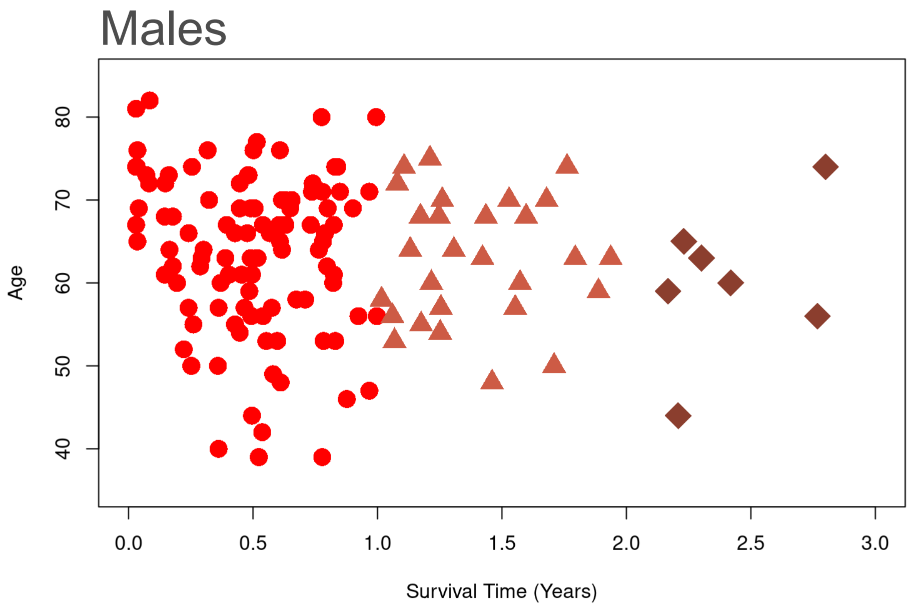
```
# plot the boxplot for survival time
boxplot(lung$time, col="cornflowerblue", pch=19, cex=2,
        ylab="Survival Time (days)")
Tmp <- boxplot.stats(lung$time) # retrieve stats from the boxplot
# list the outlier values from time
lung$time[lung$time %in% boxplot.stats(lung$time)$out]
# remove the outliers from data set lung
lung <- lung[! lung$time %in% boxplot.stats(lung$time)$out,]

# More reading: https://www.simplypsychology.org/boxplots.html
```

Multivariate visualization: grouping

```
# build the groups for every year of survival for either males or females
#
# males that survived  no more than 1 year
mg1 <- lung[lung$sex.category == "Male" & lung$time/365 <= 1,]

# males that survived more than 1 year but no more than 2 years
mg2 <- lung[lung$sex.category == "Male" & lung$time/365 > 1 & lung$time/365 <= 2,]

# males that survived more than 2 years but no more than 3 years
mg3 <- lung[lung$sex.category == "Male" & lung$time/365 > 2 & lung$time/365 <= 3,]


# females that survived no more than 1 year
fg1 <- lung[lung$sex.category == "Female" & lung$time/365 <= 1,]

# females that survived more than 1 year but no more than 2 years
fg2 <- lung[lung$sex.category == "Female" & lung$time/365 > 1 & lung$time/365 <= 2,]

# females that survived more than 2 years but no more than 3 years
fg3 <- lung[lung$sex.category == "Female" & lung$time/365 > 2 & lung$time/365 <= 3,]
```

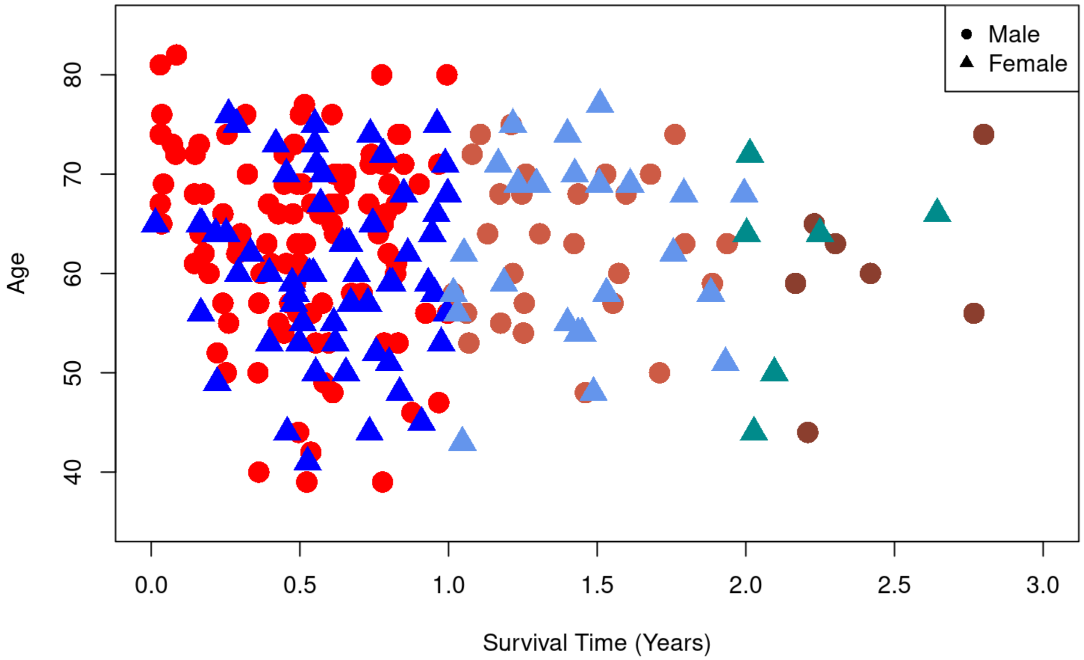
Multivariate visualization: scatter plot of age vs. survival for the groups


Females


Males

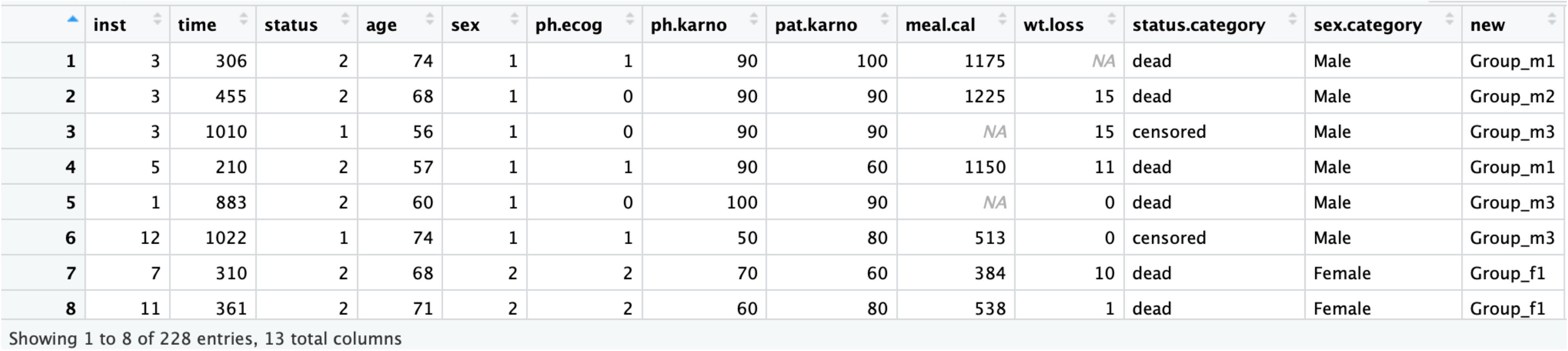
```
plot(mg1$time/365, mg1$age, xlim=c(0,3), ylim=c(35,85), pch=16,
   col = "red", cex=2, xlab="Survival Time (Years)", ylab="Age")
points(mg2$time/365, mg2$age, pch=17, col = "coral3", cex=2)
points(mg3$time/365, mg3$age, pch=18, col = "coral4", cex=3)

plot(fg1$time/365, fg1$age, xlim=c(0,3), ylim=c(35,85), pch=16,
   col = "blue", cex=2, xlab="Survival Time (Years)", ylab="Age")
points(fg2$time/365, fg2$age, pch=17, col = "cornflowerblue", cex=2)
points(fg3$time/365, fg3$age, pch=18, col = "cyan4", cex=3)
```

Multivariate visualization: scatter plot of age vs. survival for the groups -- together



```
# first group plot
plot(mg1$time/365, mg1$age, xlim=c(0,3), ylim=c(35,85), pch=16,
col = "red", cex=2, xlab="Survival Time (Years)", ylab="Age")

# subsequent additions to the plot
points(mg2$time/365, mg2$age, pch=16, col = "coral3", cex=2)
points(mg3$time/365, mg3$age, pch=16, col = "coral4", cex=2)
points(fg1$time/365, fg1$age, pch=17, col = "blue", cex=2)
points(fg2$time/365, fg2$age, pch=17, col = "cornflowerblue", cex=2)
points(fg3$time/365, fg3$age, pch=17, col = "cyan4", cex=2)
legend("topright", c("Male", "Female"), pch=c(16,17), cex=1)
```

Multivariate visualization: grouping

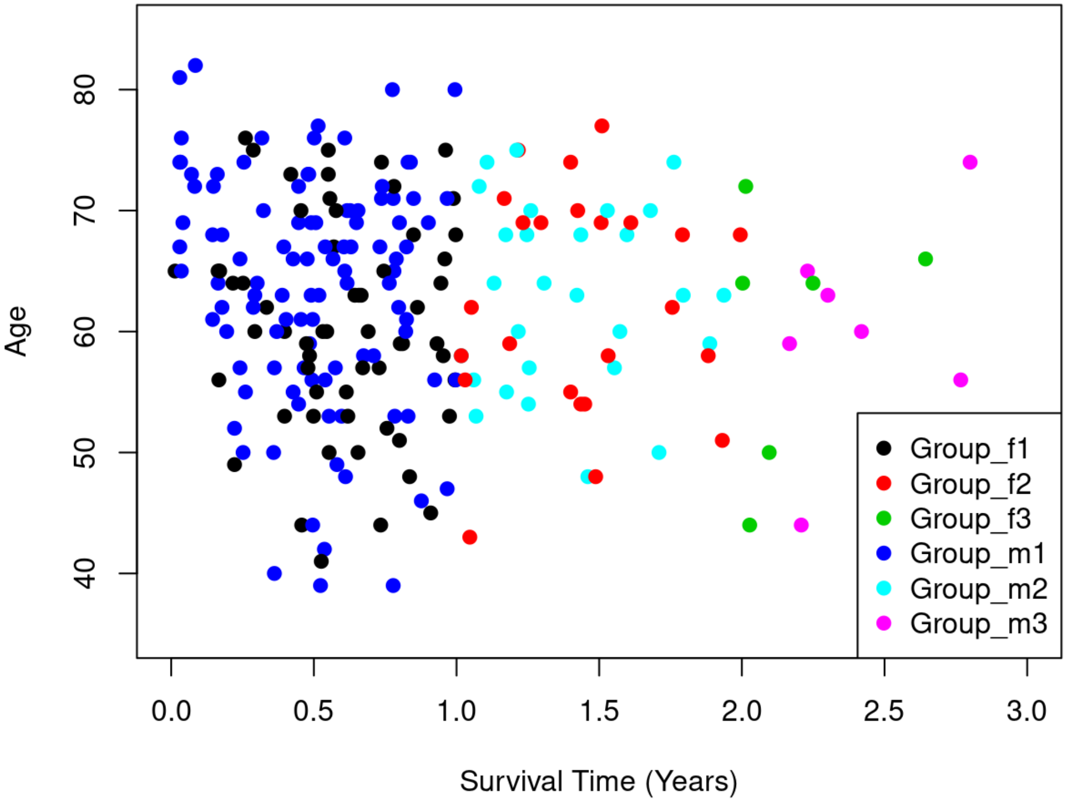


	inst	time	status	age	sex	ph.ecog	ph.karno	pat.karno	meal.cal	wt.loss	status.category	sex.category	new
1	3	306	2	74	1	1	90	100	1175	NA	dead	Male	Group_m1
2	3	455	2	68	1	0	90	90	1225	15	dead	Male	Group_m2
3	3	1010	1	56	1	0	90	90	NA	15	censored	Male	Group_m3
4	5	210	2	57	1	1	90	60	1150	11	dead	Male	Group_m1
5	1	883	2	60	1	0	100	90	NA	0	dead	Male	Group_m3
6	12	1022	1	74	1	1	50	80	513	0	censored	Male	Group_m3
7	7	310	2	68	2	2	70	60	384	10	dead	Female	Group_f1
8	11	361	2	71	2	2	60	80	538	1	dead	Female	Group_f1

Showing 1 to 8 of 228 entries, 13 total columns

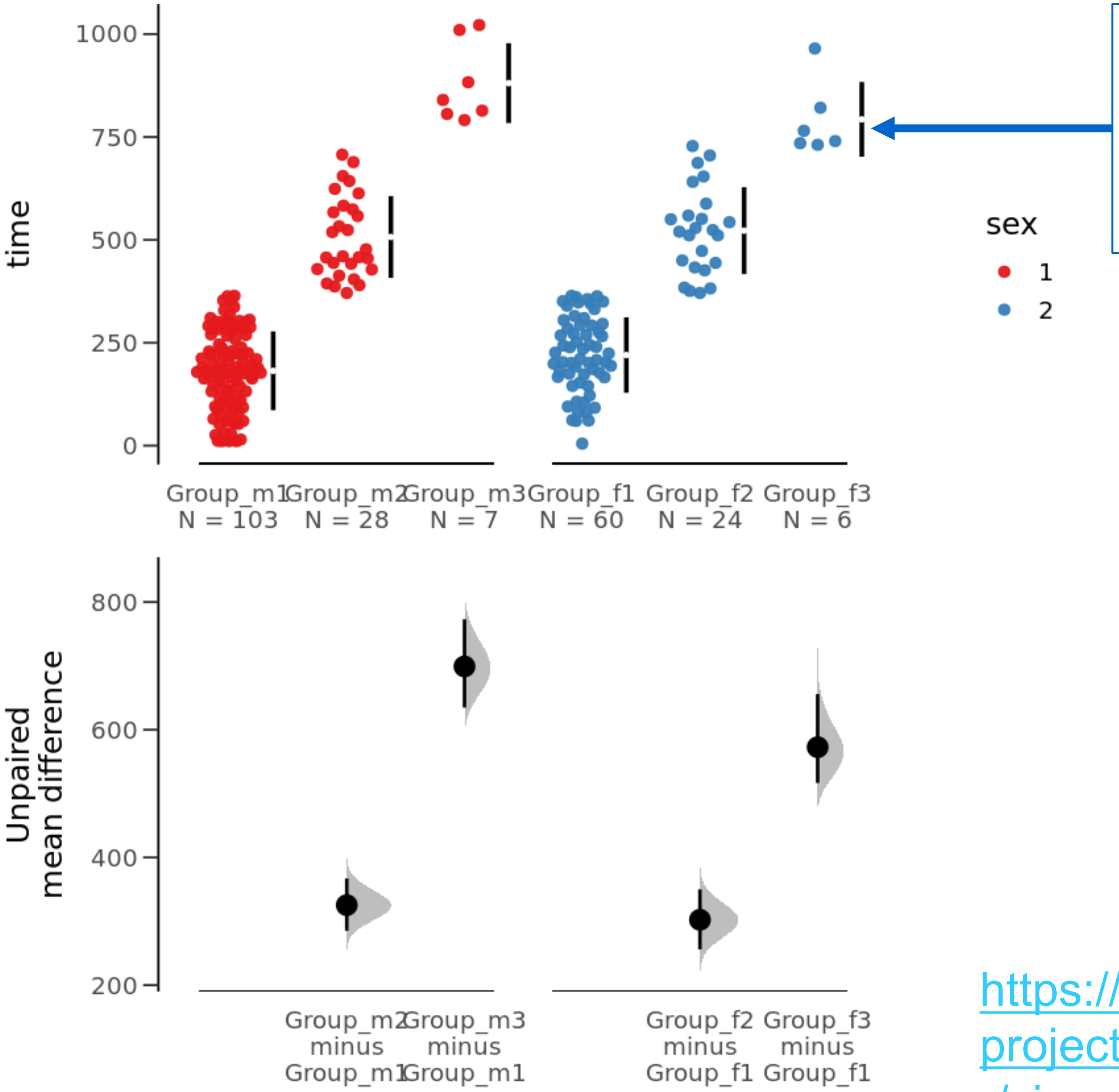
```
# build the groups for every year of survival for either males or females
# by adding a new column into lung and creating new group labels
lung$new <- NA; # new column filled with NAs

# males that survived  no more than 1 year
lung[lung$sex.category=="Male" & lung$time/365<=1,]$new <- "Group_m1 "
# males that survived more than 1 year but no more than 2 years
lung[lung$sex.category=="Male" & lung$time/365>1 & lung$time/365<=2,]$new <- "Group_m2 "
# males that survived more than 2 years but no more than 3 years
lung[lung$sex.category=="Male" & lung$time/365>2 & lung$time/365<=3,]$new <- "Group_m3 "

# females that survived no more than 1 year
lung[lung$sex.category=="Female" & lung$time/365<=1,]$new <- "Group_f1"
# females that survived more than 1 year but no more than 2 years
lung[lung$sex.category=="Female" & lung$time/365>1 & lung$time/365<=2,]$new <- "Group_f2"
# females that survived more than 2 years but no more than 3 years
lung[lung$sex.category=="Female" & lung$time/365>2 & lung$time/365<=3,]$new <- "Group_f3"
```

Multivariate visualization: scatter plot of age vs. survival for the groups – together (easy)



```
plot(lung$time/365, lung$age, pch=19, col=factor(lung$new),
     xlim=c(0,3), ylim=c(35,85), xlab="Survival Time (Years)", ylab="Age")

legend("bottomright", legend = levels(factor(lung$new)), pch = 19,
       col = factor(levels(factor(x$new))))
# More: https://r-charts.com/correlation/scatter-plot-group/
```

Multivariate visualization: estimation plot for multiple comparison



These vertical lines are identical to conventional mean ± standard deviation error bars.

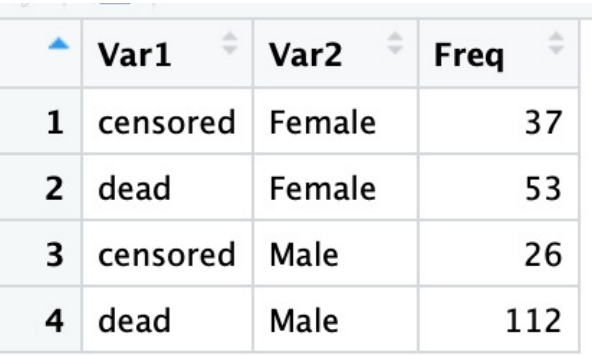
https://cran.r-project.org/web/packages/dabestr/vignettes/using-dabestr.html

Tabulation: frequencies

```
> table(lung$sex)
     1    2
    90  138

> table(lung$sex.category)
   Female           Male
       90            138

> table(lung$status)
         1         2
        63       165

> table(lung$status.category)
 Censored          dead
       63           165
```

Tabulation: contingency tables (2-way)

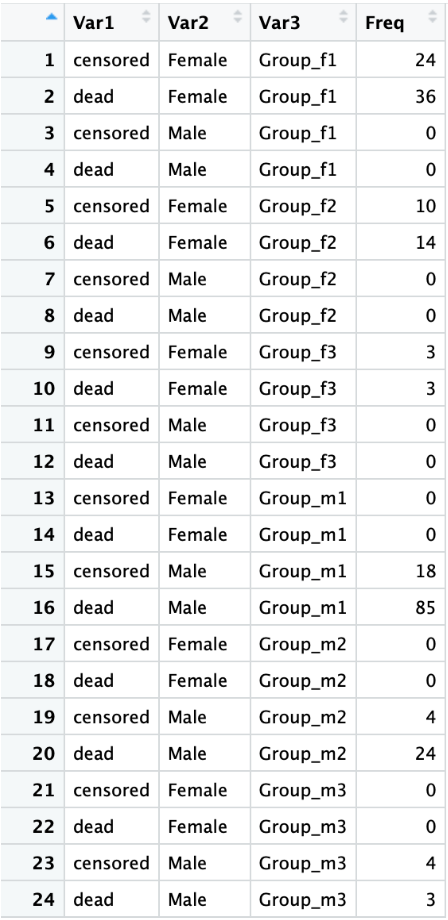
```
> mytable <- table(lung$status.category,lung$sex.category)
> mytable
                Female     Male
     censored       37       26
     dead           53      112

> margin.table(mytable, 1) # sum along rows
censored     dead
      63      165

> margin.table(mytable, 2) # sum along columns
Female Male
    90  138

> prop.table(mytable) # percentages of each cell
            Female      Male
censored 0.1622807 0.1140351
dead     0.2324561 0.4912281

> prop.table(mytable, 1) # row percentages
            Female      Male
censored 0.5873016 0.4126984
dead     0.3212121 0.6787879

> prop.table(mytable, 2) # column percentages
            Female      Male
censored 0.4111111 0.1884058
dead     0.5888889 0.8115942
```



	Var1	Var2	Freq
1	censored	Female	37
2	dead	Female	53
3	censored	Male	26
4	dead	Male	112

Tabulation: contingency tables (3-way)

mytable1



	Var1	Var2	Var3	Freq
1	censored	Female	Group_f1	24
2	dead	Female	Group_f1	36
3	censored	Male	Group_f1	0
4	dead	Male	Group_f1	0
5	censored	Female	Group_f2	10
6	dead	Female	Group_f2	14
7	censored	Male	Group_f2	0
8	dead	Male	Group_f2	0
9	censored	Female	Group_f3	3
10	dead	Female	Group_f3	3
11	censored	Male	Group_f3	0
12	dead	Male	Group_f3	0
13	censored	Female	Group_m1	0
14	dead	Female	Group_m1	0
15	censored	Male	Group_m1	18
16	dead	Male	Group_m1	85
17	censored	Female	Group_m2	0
18	dead	Female	Group_m2	0
19	censored	Male	Group_m2	4
20	dead	Male	Group_m2	24
21	censored	Female	Group_m3	0
22	dead	Female	Group_m3	0
23	censored	Male	Group_m3	4
24	dead	Male	Group_m3	3

```
> mytable1 <- table(lung$status.category,lung$sex.category,lung$new)
> ftable(mytable1)
                 Group_f1 Group_f2 Group_f3 Group_m1 Group_m2 Group_m3
Censored Female        24       10        3        0        0        0
         Male           0        0        0       18        4        4
Dead     Female        36       14        3        0        0        0
         Male           0        0        0       85       24        3
```

Independence test: Chi-square test

- the sample is large enough (in this case
- the p-value is an approximation (becomes exact with infinite sample size)

```
> mytable <- table(lung$status.category,lung$sex.category) # 2-way
> mytable
            Female    Male
 censored       37      26
     dead       53     112

> mytable <- table(lung$status.category,lung$sex.category)
> summary(mytable)
Number of cases in table: 228
Number of factors: 2
Test for independence of all factors:
        Chisq = 13.511, df = 1, p-value = 0.0002371

> mytable1 <- table(lung$status.category,lung$sex.category,lung$new)
> ftable(mytable1)
                   Group_f1 Group_f2 Group_f3 Group_m1 Group_m2 Group_m3
Censored  Female         24       10        3        0        0        0
          Male            0        0        0       18        4        4
Dead      Female         36       14        3        0        0        0
          Male            0        0        0       85       24        3

> summary(mytable)
Number of cases in table: 228
Number of factors: 2
Test for independence of all factors:
        Chisq = 13.511, df = 1, p-value = 0.0002371
```

Independence Test: Fisher’s exact test

- For small sample size
- The p-value is exact, not an approximation.

```
> mytable <- table(lung$status.category,lung$sex.category) # 2-way
> mytable
             Female     Male
 censored        37       26
     dead        53      112

> fisher.test(mytable)
               Fisher's Exact Test for Count Data

data: mytable
p-value = 0.0004349
alternative hypothesis: true
odds ratio is not equal to 1
95 percent confidence interval:
1.583762 5.727861
sample estimates:
odds ratio
2.991585
```

Further reading:
https://statsandr.com/blog/fisher-s-exact-test-in-r-independence-test-for-a-small-sample/