

Introduction to \mathcal{R}

Session 5. Basic Statistics

Dag Tanneberg¹

Potsdam Center for Quantitative Research
University of Potsdam, Germany
November 8/9, 2019

¹Quantitative Social Scientist, www.dtanneberg.de

Introduction

So far, you have learned. . .

- to navigate the \mathcal{R} environment,
- to create, index, and modify objects,
- to take advantage of control flow statements,
- generate decent looking graphs.

Looks like you are ready to unleash \mathcal{R} on real data.

What are we going to do?

We are going to plunge head first into data analysis, taking advantage of data on all *named* characters in George R.R. Martin's "A Song of Ice and Fire".² In the end, you will be able to offer scientific insight on the erstwhile mother of all cocktail party questions: **What's the chance that Jon Snow is going to die?**

To get started:

- 1 Quit & reopen \mathcal{R} .
- 2 Load `"./05/dta/asoiaf.csv"` from the course material.
 - **Note:** Uncheck the option "Strings as factors".
- 3 Open a new script file.
- 4 Load `"ggplot2"`.

²O'Neill, M. 2016. Game of Thrones. <https://bit.ly/2qjUfQ2> (last access: 10/08/2018).

Outline

- 1 Introduction
- 2 Many Values
- 3 Few Values
- 4 Adventures in Association
- 5 Summary

Few or Many Values?

We are interested in the variables `allegiances`, `age_in_chapters`, `gender`, and `nobility`.

- Proper tools for description change with data type
- Indicator: How many different values does a variable take?

Many Values

Central Tendency and Spread

Enter each of the following commands. Explain the output. Set `na.rm` to `FALSE`. What happens?

```
mean(asoiaf[, "age_in_chapters"], na.rm = TRUE)
sd(asoiaf[, "age_in_chapters"], na.rm = TRUE)
quantile(asoiaf[, "age_in_chapters",
  probs = c(0, .01, .05, .25, .5, .75, .9, .95, 1),
  na.rm = TRUE
)
summary(asoiaf[, "age_in_chapters"])
```

■ What have we learned?

Central Tendency and Spread

Enter each of the following commands. Explain the output. Set `na.rm` to `FALSE`. What happens?

```
mean(asoiaf[, "age_in_chapters"], na.rm = TRUE)
sd(asoiaf[, "age_in_chapters"], na.rm = TRUE)
quantile(asoiaf[, "age_in_chapters",
  probs = c(0, .01, .05, .25, .5, .75, .9, .95, 1),
  na.rm = TRUE
)
summary(asoiaf[, "age_in_chapters"])
```

- What have we learned?
- `mean()`, `sd()`, and `quantile()` return just that.

Central Tendency and Spread

Enter each of the following commands. Explain the output. Set `na.rm` to `FALSE`. What happens?

```
mean(asoiaf[, "age_in_chapters"], na.rm = TRUE)
sd(asoiaf[, "age_in_chapters"], na.rm = TRUE)
quantile(asoiaf[, "age_in_chapters",
  probs = c(0, .01, .05, .25, .5, .75, .9, .95, 1),
  na.rm = TRUE
)
summary(asoiaf[, "age_in_chapters"])
```

- **What have we learned?**
- `mean()`, `sd()`, and `quantile()` return just that.
- Each requires instructions on how to process NAs.

Central Tendency and Spread

Enter each of the following commands. Explain the output. Set `na.rm` to `FALSE`. What happens?

```
mean(asoiaf[, "age_in_chapters"], na.rm = TRUE)
sd(asoiaf[, "age_in_chapters"], na.rm = TRUE)
quantile(asoiaf[, "age_in_chapters"],
  probs = c(0, .01, .05, .25, .5, .75, .9, .95, 1),
  na.rm = TRUE
)
summary(asoiaf[, "age_in_chapters"])
```

- **What have we learned?**
- `mean()`, `sd()`, and `quantile()` return just that.
- Each requires instructions on how to process NAs.
- `summary()` returns the 5-point-summary plus mean and NAs.

Graphical EDA

ggplot2 offers numerous exploratory graphs.³ Create each of the graphs below. What do they return?

```
p <- ggplot(data = asoiaf, aes(x = age_in_chapters))  
p + geom_histogram()  
p + geom_density() + labs(y = "PDF")  
p + stat_ecdf() + labs(y = "CDF")  
p + geom_boxplot(aes(x = 0, y = age_in_chapters))  
ggplot(data = asoiaf, aes(sample = age_in_chapters)) +  
  geom_qq() + geom_qq_line()
```

³For an entire theory of graphical EDA using ggplot2 see Unwin, A. 2015. Graphical Data Analysis with R. Boca Raton: CRC Press.

Grouping Values

- **Goal:** Controlled loss of information for, e.g., tables

```
# Variant a. Aggregate data -----  
mu_age_by_allegiance <- aggregate(  
  x = asoiaf[, "age_in_chapters"],  
  by = list(allegiances = asoiaf[, "allegiances"]),  
  FUN = mean, na.rm = TRUE  
); mu_age_by_allegiance
```

Grouping Values, contd.

```
# Variant b. Recode the data -----
tmp <- cut(x = asoiaf[, "age_in_chapters"],
  breaks = 5
  # divides data into <breaks> pieces of equal length
); summary(tmp) # Note something weird?
tmp <- cut(x = asoiaf[, "age_in_chapters"],
  breaks = quantile(
    asoiaf[, "age_in_chapters"], na.rm = TRUE
  ), # vector of values at which to cut x.
  include.lowest = FALSE
); summary(tmp) # Notice something weird?
typeof(tmp); class(tmp) # Notice something weird?
```

Few Values

What are factors?

- Special instance (“class”) of atomic vectors
- Store nominal and ordinal data, e.g., eye color & letter grades
- Look like character strings, but may behave like integers
- Create factors only when needed

```
grades <- c("A", "B", "B", "C")
grades <- factor(grades,
  levels = c("C", "B", "A"), # state ALL values
  labels = c("C", "B", "A"), # name EACH value
  ordered = TRUE # defaults to FALSE (nominal data)
)
typeof(grades); attributes(grades) # Try these.
```


What are factors?, contd.

- Components of a factor: numeric value & character label
- labels (BUT NOT VALUES) can be used for logical indexing

```
grades; as.numeric(grades)
```

```
## [1] A B B C
```

```
## Levels: C < B < A
```

```
## [1] 3 2 2 1
```

```
grades > "C" # will work fine
```

```
## [1] TRUE TRUE TRUE FALSE
```

```
grades > 1 # will generally not work
```

```
## [1] NA NA NA NA
```

Simple N-way Contingency Tables

- `table()` creates N-way contingency tables

```
table(asoiaf[, "gender"]) # single 1way table
```

```
##
```

```
##    0    1
```

```
## 157 760
```

Explain the output of these statements. Explain the difference in your own words.

```
apply(
  asoiaf[, c("gender", "nobility")], 2, table
)
table(asoiaf[, "book_of_death"], asoiaf[, "nobility"])
```

Add Information to Contingency Tables

- You must state explicitly what information you require.
- Examples: Proportions & Totals

```
mytable <- table(  
  "gender" = asoiaf[, 'gender'],  
  "nobility" = asoiaf[, 'nobility']  
)  
prop.table(mytable)  
# Add argument margin = {1; 2}. What happens?  
addmargins(mytable)  
# Add argument margin = {1; 2}. What happens?
```

Adventures in Association

Two-way Contingency Tables

- Numerous methods provided
- Most defined by individual functions
- See packages **vcd** & **vcdExtra** for more options

```
mytable <- table(  
  "gender" = asoiaf[, 'gender'],  
  "nobility" = asoiaf[, 'nobility']  
)  
fit <- chisq.test(mytable); fit # Chi-Square Test  
fit <- fisher.test(mytable); fit # Fisher's Exact Test
```

Correlation Analysis

- Scatter plots are the starting point for any correlation analysis
- `base::pairs()` & `car::scatterplotMatrix()` return plot matrices
- Quantify associations using `cor()` and `cor.test()` functions

```
# a. Explain the code and plot. -----
ggplot(data = asoiaf,
  aes(x = chapter_of_intro, y = chapter_of_death)
) + geom_point() +
  geom_smooth(aes(col = "loess"), method = "loess") +
  geom_smooth(aes(col = "ols"), method = "lm")
# b. Correlation Analysis -----
cor(x = asoiaf[, c(6, 8)],
  use = "complete.obs", # What does <use> do?
  method = "pearson" # {pearson; kendall; spearman}
) # Now try cor.test() on your own.
```

Mean Comparison Tests

- Question: Do two groups come from the same population?
- In ASOIAF: Do nobles survive longer than other social strata?

```
t.test( # Alternative:  $\mu_0 < \mu_1$   
  age_in_chapters ~ nobility, data = asoiaf,  
  alternative = "less" # "greater" for  $\mu_0 > \mu_1$   
)
```

- Defaults to unequal variances & returns Welch's t-test
- Set paired = TRUE for paired t-tests
- For detailed instructions see Field, A. et al. 2012. Discovering Statistics Using R. Los Angeles: SAGE, Ch. 9.

Summary

R functions for basic statistics⁴

Method	\mathcal{R} Function(s) in base
Mean	mean()
Median	median()
Quantiles	quantile()
Extremes	range(), min(), max()
Variance	var()
Standard Deviation	sd()
Covariance Matrix	vcov()
Correlation	cor()
Analysis of Variance	aov(), lm(), anova()
Contingency Tables	table(), xtabs()
t-test for Means	prop.test(), binom.test()
χ^2 Test for Indep.	chisq.test()
Various Nonparametric Tests	friedman.test(), kruskal.test(), wilcox.test(), etc.

⁴Selectively adapted from Fox, J. and S. Weisberg. 2011. An R Companion to Applied Regression. 2nd ed. London: SAGE.