Stochastic Process Model with Basketball Data

A Thesis
Presented to
Department of Statistical Science
Duke University

Sonia Xu

October 2017

Approved for the
Bachelor of Science in Statistical Science

_____

Dr. Alexander Volfovsky

_____

Committeemember O. Name

_____

Committeemember T. Name

_____

Dus X. Name, DUS

Acknowledgements

I want to thank a few people.

Preface

This is an example of a thesis setup to use the reed thesis document class (for LaTeX) and the R bookdown package, in general.

Table of Contents

Introduction **1**

Chapter 1: Boxscore **3**

Chapter 2: Literature Review **5**

Chapter 3: Dataset **7**

Chapter 4: Model Replication **9**
4.1 Motivation 9
4.2 Microtransition Model 9
4.3 Macrotransition Entrance Model 9
4.4 Macrotransition Exit Model 9

Chapter 5: Macrotransition Exit Model **11**

Chapter 6: Implementation of this Model **13**
6.1 in the works... 13
6.2 Next Steps (have yet to get this far) 13
6.3 Exploratory Data Analysis 13
6.4 Line breaks 14
6.5 R chunks 14
6.6 Inline code 14
6.7 Including plots 15
6.8 Loading and exploring data 16
6.9 Additional resources 19

Chapter 7: Math typesetting **21**
7.1 Math 21

Chapter 8: Tables, Graphics, References, and Labels **23**
8.1 Tables 23
8.2 Figures 24
8.3 Footnotes and Endnotes 26
8.4 Bibliographies 26
8.5 Anything else? 27

Chapter 9: Organization **29**

Conclusion **31**

Appendix A: The First Appendix **33**

Appendix B: The Second Appendix, for Fun **35**

References **37**

List of Tables

6.1 Max Delays by Airline 18

8.1 Correlation of Inheritance Factors for Parents and Child 23

List of Figures

8.1 Duke logo 24
8.2 Mean Delays by Airline 25
8.3 Subdiv. graph 26
8.4 A Larger Figure, Flipped Upside Down 26

Abstract

The preface pretty much says it all.

Second paragraph of abstract starts here.

Dedication

You can have a dedication here if you wish.

Introduction

Welcome to the *R Markdown* thesis template. This template is based on (and in many places copied directly from) the Reed College LaTeX template, but hopefully it will provide a nicer interface for those that have never used TeX or LaTeX before. Using *R Markdown* will also allow you to easily keep track of your analyses in **R** chunks of code, with the resulting plots and output included as well. The hope is this *R Markdown* template gets you in the habit of doing reproducible research, which benefits you long-term as a researcher, but also will greatly help anyone that is trying to reproduce or build onto your results down the road.

Hopefully, you won't have much of a learning period to go through and you will reap the benefits of a nicely formatted thesis. The use of LaTeX in combination with *Markdown* is more consistent than the output of a word processor, much less prone to corruption or crashing, and the resulting file is smaller than a Word file. While you may have never had problems using Word in the past, your thesis is likely going to be about twice as large and complex as anything you've written before, taxing Word's capabilities. After working with *Markdown* and **R** together for a few weeks, we are confident this will be your reporting style of choice going forward.

Why use it?

R Markdown creates a simple and straightforward way to interface with the beauty of LaTeX. Packages have been written in **R** to work directly with LaTeX to produce nicely formatting tables and paragraphs. In addition to creating a user friendly interface to LaTeX, *R Markdown* also allows you to read in your data, to analyze it and to visualize it using **R** functions, and also to provide the documentation and commentary on the results of your project. Further, it allows for **R** results to be passed inline to the commentary of your results. You'll see more on this later.

Having your code and commentary all together in one place has a plethora of benefits!

Who should use it?

Anyone who needs to use data analysis, math, tables, a lot of figures, complex cross-references, or who just cares about the final appearance of their document should use *R Markdown*. Of particular use should be anyone in the sciences, but the user-friendly nature of *Markdown* and its ability to keep track of and easily include figures, automatically generate a table of contents, index, references, table of figures, etc. should make it of great benefit to nearly anyone writing a thesis project.

Chapter 1

Boxscore

In basketball, a boxscore provides the statistical summary of the game via defensive, offensive, and overall success metrics. Popular metrics include rebounds per game (RBG), player efficiency rating (PER), free throw attempts (FTA), and 3 field goals made (3FGM). However, these metrics cannot capture the entirety of the game because they do not take into account the opposing team's defense/offense, nor previous plays that significantly influenced the flow of the game.

Chapter 2

Literature Review

Previous works have sought to capture the game more robustly. Below describes a summary of a few:

"Flow Motifs in Soccer: What can passing behavior tell us?" by Joris Bekkers and Shaunak Dabadghao was released in the 2017 MIT Sloan Sports Analytics Conference, and focused on the static passing networks of "the last 4 seasons of 6 big European leagues with 8219 matches, 3532 unique players and 155 unique teams." Passing sequences were denoted as a sequence of all players involved five seconds before an attempted score. This paper created radar graphs that illustrated the most popular passing sequences by player, and compared radar graphs to identify similar players. Passing sequences within teams were also compared between teams by clustering the different passing styles of the different teams. Key players were determined by the frequency that they were included in the passing sequences.

"Exploring Team Passing Networks and Player Movement Dynamics in Youth Association Football (Soccer)" by Bruno Goncalves, Diogo Coutinho, Sara Santos, Carlos Lago-Penas, Sergio Jimenez, and Jamie Sampaio compared the passing sequences of two games played by two groups that differ in age range, which showed that regardless of age, network centrality was distinctive in both groups, and affirmed the long-held belief that more passes lead to better game outcomes. Similar to the first paper, key players were the ones most frequently involved in the passing sequences. This paper created weighted graphs of the passing sequences, which better visualized the passing structure of the team, and made it easier to identify important players.

"Basketball Teams as Strategic Networks" by Jennifer H. Fewell, Dieter Armbruster, John Ingraham, Alexander Petersen, and James S. Waters provided measurements to assess team entropy. First recording the complete 30 seconds of a possession as a passing sequence, they discovered that recording the last three nodes (players) before a shot attempt was a better way to record passing sequences to avoid "noisy" passing data. Although they were able to recognize various aspects of team dynamics through weighted graphs like the second paper, they did not find a consistent predictor of positive game outcomes. This paper also identified that in general, teams typically range between two playing styles: always passing to the best player or having no distinct patterns in passing. These patterns can be noted by distinct betweenness scores and uniform betweenness scores, respectively. Weighted graphs clearly illustrated the

two different playing styles. Also, the paper found that the positions most involved with successful shots were: 1. PG 2. SG 3. SF 4. PF 5. CN.

Joachim Gudmundsson and Michael Horton summarised a variety of methods that utilize object tracking data to analyze team and player performances in "Spatio-Temporal Analysis of Team Sports – A Survey." Their research survey spanned modeling passing networks via graph theory to calculating rebound probability with spatial coordinates. In particular, work conducted by Daniel Cervone, Alex D'Amour, Luke Bornn, and Kirk Goldsberry attempted to capture the game wholelistically via a new measure called Expected Possession Value (EPV) in the paper "A Multiresolution Stochastic Process Model for Predicting Basketball Possession Outcomes." This new metric uses three models–a Microtransition Model, Macrotransition Entrance Model, and a Macrotransition Exit Model–to capture the spatial biases of each player and the in-game effects of pressure, so that it can measure the likelihood of a successful play (made shot) given the previous sequence of events. To compare players against the league-average scores, they also calculated Expected Possession Value -Adjusted as an application for teams.

Chapter 3

Dataset

The dataset is from the Duke University Men's Basketball SportsVu tracking data. Features were created by taking snapshots of the game every 1/25th of a second and recording the player's location, action, team, etc. Data was collected for each season from 2013-2016; the dataset totals about 2 million observations and 72 features.

Chapter 4

Model Replication

4.1 Motivation

This paper is particularly interesting because EPV utilizes the spatio-temporal elements of the game, so it models the NBA game dynamically. Given Duke Basketball data, the motivation is to replicate "A Multiresolution Stochastic Process Model for Predicting Basketball Possession Outcomes," to better understand the Duke Men's team, as well as to compare professional basketball to collegiate basketball individual and team playing styles. Below is a brief overview of each model used in the paper to calculate EPV.

4.2 Microtransition Model

$x^l(t+\epsilon) = x^l(t) + \alpha_x^l[x^l(t) - x^l(t-\epsilon)] + \eta_x^l(t)$ where $\eta_x^l(t) \sim N(\mu_x^l(z^l(t)), (\sigma_x^l)^2)$

The microtransition model models the defensive conditions of the game based on the (x, y) coordinates of a player and their acceleration effects $(\alpha_x^l(t))$. It is also assumed that a player's spatial location is normally distributed. Since players play differently, each microtransition model is specifically fitted to the player.

4.3 Macrotransition Entrance Model

$P(M(t)|F_t^{(Z)}$ The macrotransition entrance model predicts whether the next move will be a pass (4 options), shot attempt, or turnover. The model is disjoint.

4.4 Macrotransition Exit Model

$P(C_{\delta_t}|M(t), F_t^{(Z)})$ Given the Macrotransition Entrance Model predicts a shot attempt, it indexes to a logistic regression model to calculate player l's successful shot probability. Given the Macrotransition Entrance Model predicts a pass,it indexes to a model that predicts where the pass will take place. Otherwise, a turnover is assumed.

Chapter 5

Macrotransition Exit Model

equation

Chapter 6

Implementation of this Model

6.1 in the works...

6.2 Next Steps (have yet to get this far)

Both metrics calculated via a semi-Markov process, EPV fails to capture the full nature of the possession because it only uses the last posession as a prior. The model would be more robust if it captured the entirety of the posession in its prior–however, the computational time of such an ordeal would prevent any real-time analyses. Thus, this paper proposes that a simpler model may perform more quickly and potentially just as robustly to allow for game-time analyses.

Here is a brief introduction into using *R Markdown*. *Markdown* is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. *R Markdown* provides the flexibility of *Markdown* with the implementation of **R** input and output. For more details on using *R Markdown* see `http://rmarkdown.rstudio.com`.

Be careful with your spacing in *Markdown* documents. While whitespace largely is ignored, it does at times give *Markdown* signals as to how to proceed. As a habit, try to keep everything left aligned whenever possible, especially as you type a new paragraph. In other words, there is no need to indent basic text in the Rmd document (in fact, it might cause your text to do funny things if you do).

6.3 Exploratory Data Analysis

-eda, write up literature

It's easy to create a list. It can be unordered like

- Item 1
- Item 2

or it can be ordered like

1. Item 1
2. Item 2

Notice that I intentionally mislabeled Item 2 as number 4. *Markdown* automatically figures this out! You can put any numbers in the list and it will create the list. Check it out below.

To create a sublist, just indent the values a bit (at least four spaces or a tab). (Here's one case where indentation is key!)

1. Item 1
2. Item 2
3. Item 3
    - Item 3a
    - Item 3b

6.4 Line breaks

Make sure to add white space between lines if you'd like to start a new paragraph. Look at what happens below in the outputted document if you don't:

Here is the first sentence. Here is another sentence. Here is the last sentence to end the paragraph. This should be a new paragraph.

Now for the correct way:

Here is the first sentence. Here is another sentence. Here is the last sentence to end the paragraph.

This should be a new paragraph.

6.5 R chunks

When you click the **Knit** button above a document will be generated that includes both content as well as the output of any embedded **R** code chunks within the document. You can embed an **R** code chunk like this (`cars` is a built-in **R** dataset):

```
summary(cars)
```

```
   speed           dist       
Min.   : 4.0   Min.   :  2.00  
1st Qu.:12.0   1st Qu.: 26.00  
Median :15.0   Median : 36.00  
Mean   :15.4   Mean   : 42.98  
3rd Qu.:19.0   3rd Qu.: 56.00  
Max.   :25.0   Max.   :120.00  
```

6.6 Inline code

If you'd like to put the results of your analysis directly into your discussion, add inline code like this:

The `cos` of 2π is 1.

Another example would be the direct calculation of the standard deviation:

The standard deviation of `speed` in `cars` is 5.2876444.

One last neat feature is the use of the `ifelse` conditional statement which can be used to output text depending on the result of an **R** calculation:

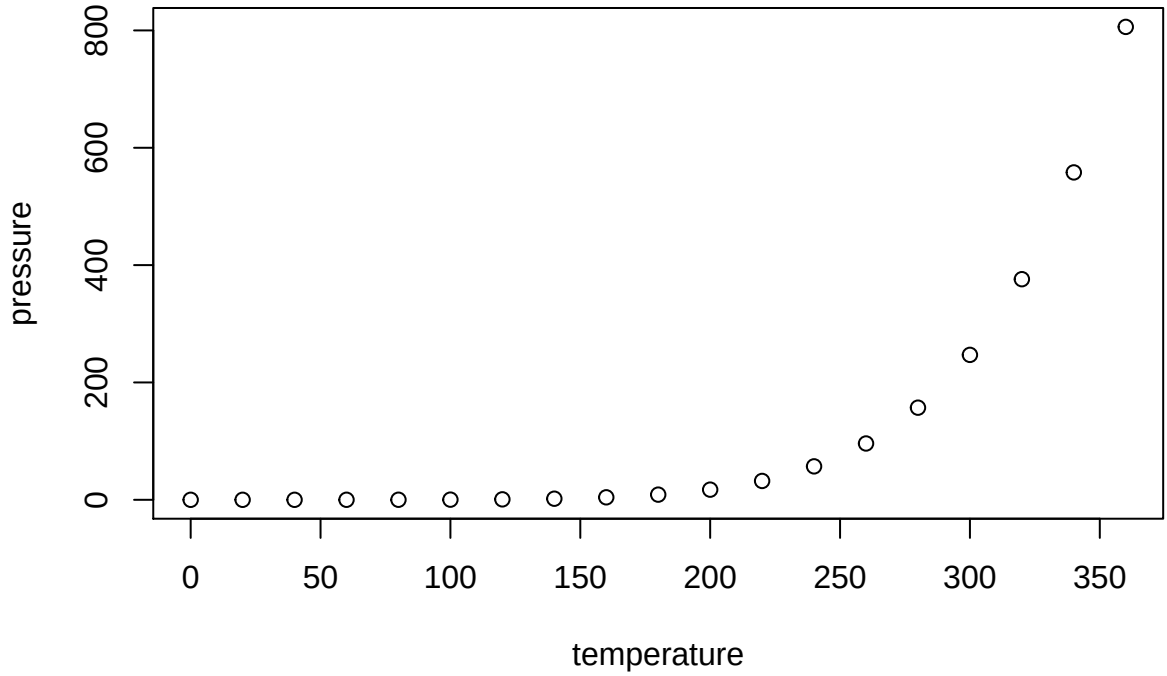
The standard deviation is less than 6.

Note the use of `>` here, which signifies a quotation environment that will be indented.

As you see with `$2 \pi$` above, mathematics can be added by surrounding the mathematical text with dollar signs. More examples of this are in [Mathematics and Science] if you uncomment the code in Math.

6.7 Including plots

You can also embed plots. For example, here is a way to use the base **R** graphics package to produce a plot using the built-in `pressure` dataset:



Note that the `echo=FALSE` parameter was added to the code chunk to prevent printing of the **R** code that generated the plot. There are plenty of other ways to add chunk options. More information is available at `http://yihui.name/knitr/options/`.

Another useful chunk option is the setting of `cache=TRUE` as you see here. If document rendering becomes time consuming due to long computations or plots that are expensive to generate you can use knitr caching to improve performance. Later in this file, you'll see a way to reference plots created in **R** or external figures.

6.8 Loading and exploring data

Included in this template is a file called `flights.csv`. This file includes a subset of the larger dataset of information about all flights that departed from Seattle and Portland in 2014. More information about this dataset and its **R** package is available at `http://github.com/ismayc/pnwflights14`. This subset includes only Portland flights and only rows that were complete with no missing values. Merges were also done with the `airports` and `airlines` data sets in the `pnwflights14` package to get more descriptive airport and airline names.

We can load in this data set using the following command:

```
flights <- read.csv("data/flights.csv")
```

The data is now stored in the data frame called `flights` in **R**. To get a better feel for the variables included in this dataset we can use a variety of functions. Here we can see the dimensions (rows by columns) and also the names of the columns.

```
dim(flights)
```

```
[1] 52808    16
```

```
names(flights)
```

```
 [1] "month"        "day"          "dep_time"     "dep_delay"
 [5] "arr_time"     "arr_delay"    "carrier"      "tailnum"
 [9] "flight"       "dest"         "air_time"     "distance"
[13] "hour"         "minute"       "carrier_name" "dest_name"
```

Another good idea is to take a look at the dataset in table form. With this dataset having more than 50,000 rows, we won't explicitly show the results of the command here. I recommend you enter the command into the Console ***after*** you have run the **R** chunks above to load the data into **R**.

```
View(flights)
```

While not required, it is highly recommended you use the `dplyr` package to manipulate and summarize your data set as needed. It uses a syntax that is easy to understand using chaining operations. Below I've created a few examples of using

`dplyr` to get information about the Portland flights in 2014. You will also see the use of the `ggplot2` package, which produces beautiful, high-quality academic visuals.

We begin by checking to ensure that needed packages are installed and then we load them into our current working environment:

```
# List of packages required for this analysis
pkg <- c("dplyr", "ggplot2", "knitr", "bookdown", "devtools")
# Check if packages are not installed and assign the
# names of the packages not installed to the variable new.pkg
new.pkg <- pkg[!(pkg %in% installed.packages())]
# If there are any packages in the list that aren't installed,
# install them
if (length(new.pkg))
  install.packages(new.pkg, repos = "http://cran.rstudio.com")
# Load packages (thesisdowndss will load all of the packages as well)
library(thesisdowndss)
```

The example we show here does the following:

- Selects only the `carrier_name` and `arr_delay` from the `flights` dataset and then assigns this subset to a new variable called `flights2`.
- Using `flights2`, we determine the largest arrival delay for each of the carriers.

```
flights2 <- flights %>%
  select(carrier_name, arr_delay)
max_delays <- flights2 %>%
  group_by(carrier_name) %>%
  summarize(max_arr_delay = max(arr_delay, na.rm = TRUE))
```

A useful function in the `knitr` package for making nice tables in *R Markdown* is called `kable`. It is much easier to use than manually entering values into a table by copying and pasting values into Excel or LaTeX. This again goes to show how nice reproducible documents can be! (Note the use of `results="asis"`, which will produce the table instead of the code to create the table.) The `caption.short` argument is used to include a shorter title to appear in the List of Tables.

```
kable(max_delays,
      col.names = c("Airline", "Max Arrival Delay"),
      caption = "Maximum Delays by Airline",
      caption.short = "Max Delays by Airline",
      longtable = TRUE,
      booktabs = TRUE)
```

Table 6.1: Maximum Delays by Airline

Airline	Max Arrival Delay
Alaska Airlines Inc.	338
American Airlines Inc.	1539
Delta Air Lines Inc.	651
Frontier Airlines Inc.	575
Hawaiian Airlines Inc.	407
JetBlue Airways	273
SkyWest Airlines Inc.	421
Southwest Airlines Co.	694
United Air Lines Inc.	472
US Airways Inc.	347
Virgin America	366

The last two options make the table a little easier-to-read.
We can further look into the properties of the largest value here for American

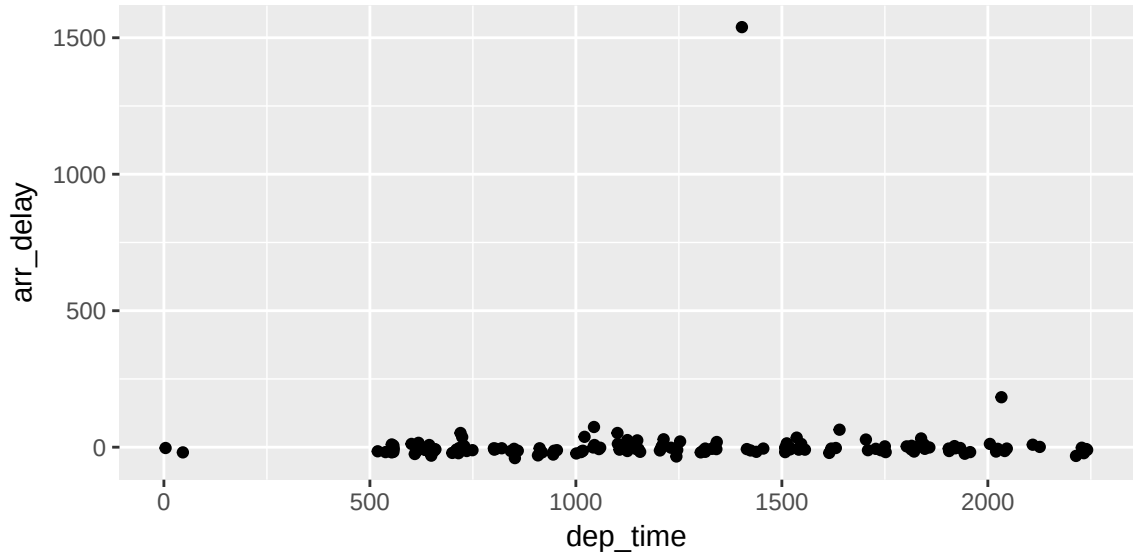
Airlines Inc. To do so, we can isolate the row corresponding to the arrival delay of 1539 minutes for American in our original `flights` dataset.

```
flights %>% filter(arr_delay == 1539,
                   carrier_name == "American Airlines Inc.") %>%
  select(-c(month, day, carrier, dest_name, hour,
            minute, carrier_name, arr_delay))
```

```
  dep_time dep_delay arr_time tailnum flight dest air_time distance
1     1403       1553     1934  N595AA   1568  DFW      182     1616
```

We see that the flight occurred on March 3rd and departed a little after 2 PM on its way to Dallas/Fort Worth. Lastly, we show how we can visualize the arrival delay of all departing flights from Portland on March 3rd against time of departure.

```
flights %>% filter(month == 3, day == 3) %>%
  ggplot(aes(x = dep_time, y = arr_delay)) + geom_point()
```



6.9 Additional resources

- *Markdown* Cheatsheet - `https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet`
- *R Markdown* Reference Guide - `https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf`
- `dplyr` Documentation - `http://dplyr.tidyverse.org/`
- `ggplot2` Documentation - `http://ggplot2.tidyverse.org/`

Chapter 7

Math typesetting

7.1 Math

TEX is the best way to typeset mathematics. Donald Knuth designed TEX when he got frustrated at how long it was taking the typesetters to finish his book, which contained a lot of mathematics. One nice feature of *R Markdown* is its ability to read LaTeX code directly.

If you are doing a thesis that will involve lots of math, you will want to read the following section.

$$\sum_{j=1}^{n}(\delta\theta_j)^2 \leq \frac{\beta_i^2}{\delta_i^2+\rho_i^2}\left[2\rho_i^2+\frac{\delta_i^2\beta_i^2}{\delta_i^2+\rho_i^2}\right] \equiv \omega_i^2$$

From Informational Dynamics, we have the following (Dave Braden):
After n such encounters the posterior density for θ is

$$\pi(\theta|X_1 < y_1, \ldots, X_n < y_n) \propto \pi(\theta)\prod_{i=1}^{n}\int_{-\infty}^{y_i}\exp\left(-\frac{(x-\theta)^2}{2\sigma^2}\right)\,dx$$

Another equation:

$$\det\begin{vmatrix} c_0 & c_1 & c_2 & \ldots & c_n \\ c_1 & c_2 & c_3 & \ldots & c_{n+1} \\ c_2 & c_3 & c_4 & \ldots & c_{n+2} \\ \vdots & \vdots & \vdots & & \vdots \\ c_n & c_{n+1} & c_{n+2} & \ldots & c_{2n} \end{vmatrix} > 0$$

Lapidus and Pindar, Numerical Solution of Partial Differential Equations in Science and Engineering. Page 54

$$\int_t\left\{\sum_{j=1}^{3}T_j\left(\frac{d\phi_j}{dt}+k\phi_j\right)-kT_e\right\}w_i(t)\,dt=0, \qquad i=1,2,3.$$

L&P Galerkin method weighting functions. Page 55

$$\sum_{j=1}^{3} T_j \int_0^1 \left\{ \frac{d\phi_j}{dt} + k\phi_j \right\} \phi_i \; dt = \int_0^1 k\, T_e \phi_i dt, \qquad i = 1, 2, 3$$

Another L&P (p145)

$$\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} f\big(\xi, \eta, \zeta\big) = \sum_{k=1}^{n}\sum_{j=1}^{n}\sum_{i=1}^{n} w_i w_j w_k f\big(\xi, \eta, \zeta\big).$$

Another L&P (p126)

$$\int_{A_e} (\,\cdot\,) dx dy = \int_{-1}^{1}\int_{-1}^{1} (\,\cdot\,) \det[J] d\xi d\eta.$$

Chapter 8

Tables, Graphics, References, and Labels

8.1 Tables

In addition to the tables that can be automatically generated from a data frame in **R** that you saw in [R Markdown Basics] using the `kable` function, you can also create tables using *pandoc*. (More information is available at `http://pandoc.org/README.html#tables`.) This might be useful if you don't have values specifically stored in **R**, but you'd like to display them in table form. Below is an example. Pay careful attention to the alignment in the table and hyphens to create the rows and columns.

Table 8.1: Correlation of Inheritance Factors for Parents and Child

Factors	Correlation between Parents & Child	Inherited
Education	-0.49	Yes
Socio-Economic Status	0.28	Slight
Income	0.08	No
Family Size	0.18	Slight
Occupational Prestige	0.21	Slight

We can also create a link to the table by doing the following: Table 8.1. If you go back to Loading and exploring data and look at the `kable` table, we can create a reference to this max delays table too: Table 6.1. The addition of the `(\#tab:inher)` option to the end of the table caption allows us to then make a reference to Table `\@ref(tab:label)`. Note that this reference could appear anywhere throughout the document after the table has appeared.

8.2 Figures

If your thesis has a lot of figures, *R Markdown* might behave better for you than that other word processor. One perk is that it will automatically number the figures accordingly in each chapter. You'll also be able to create a label for each figure, add a caption, and then reference the figure in a way similar to what we saw with tables earlier. If you label your figures, you can move the figures around and *R Markdown* will automatically adjust the numbering for you. No need for you to remember! So that you don't have to get too far into LaTeX to do this, a couple **R** functions have been created for you to assist. You'll see their use below.

In the **R** chunk below, we will load in a picture stored as `duke.png` in our main directory. We then give it the caption of "Duke logo", the label of "dukelogo", and specify that this is a figure. Make note of the different **R** chunk options that are given in the R Markdown file (not shown in the knitted document).

```
include_graphics(path = "figure/duke.png")
```

Figure 8.1: Duke logo

Here is a reference to the Duke logo: Figure 8.1. Note the use of the `fig:` code here. By naming the **R** chunk that contains the figure, we can then reference that figure later as done in the first sentence here. We can also specify the caption for the figure via the R chunk option `fig.cap`.

Below we will investigate how to save the output of an **R** plot and label it in a way similar to that done above. Recall the `flights` dataset from Chapter 2. (Note that we've shown a different way to reference a section or chapter here.) We will next explore a bar graph with the mean flight departure delays by airline from Portland for 2014. Note also the use of the `scale` parameter which is discussed on the next page.

```
flights %>% group_by(carrier) %>%
  summarize(mean_dep_delay = mean(dep_delay)) %>%
  ggplot(aes(x = carrier, y = mean_dep_delay)) +
  geom_bar(position = "identity", stat = "identity", fill = "red")
```

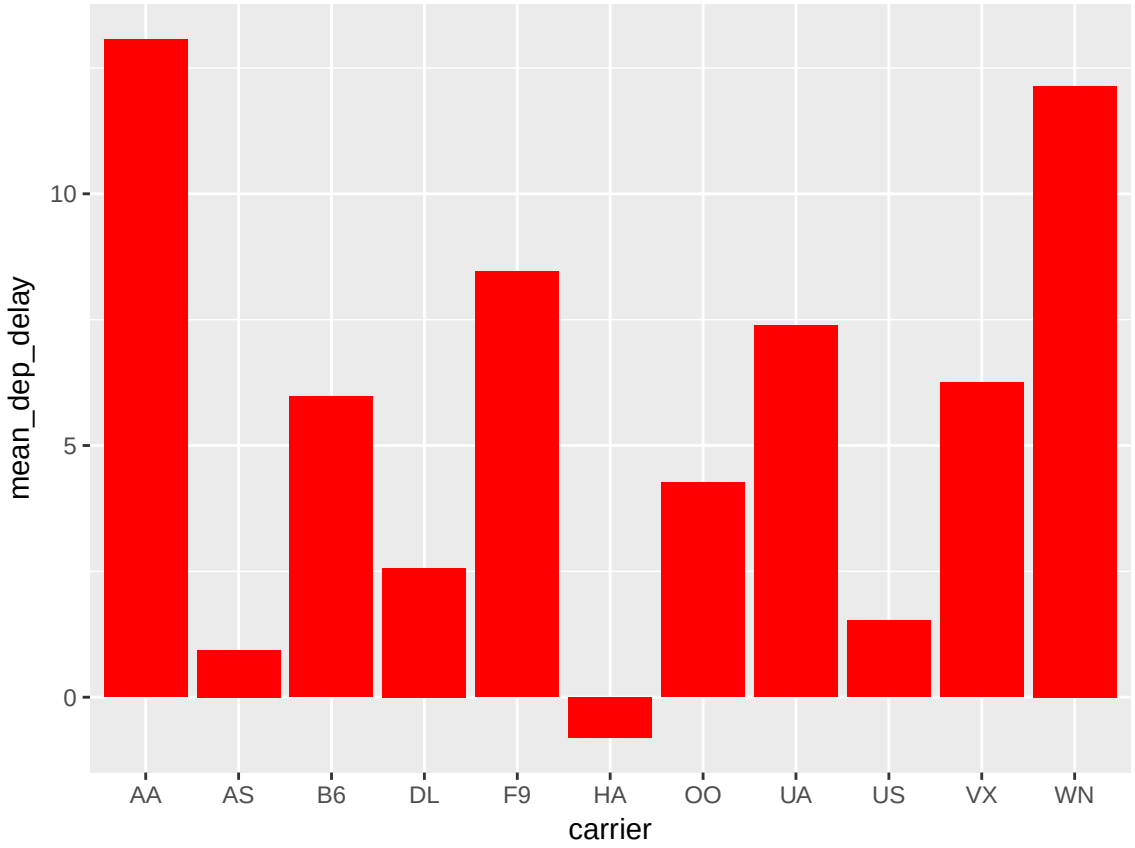


Figure 8.2: Mean Delays by Airline

Here is a reference to this image: Figure 8.2.

A table linking these carrier codes to airline names is available at `https://github.com/ismayc/pnwflights14/blob/master/data/airlines.csv`.

Next, we will explore the use of the `out.extra` chunk option, which can be used to shrink or expand an image loaded from a file by specifying `"scale= "`. Here we use the mathematical graph stored in the "subdivision.pdf" file.

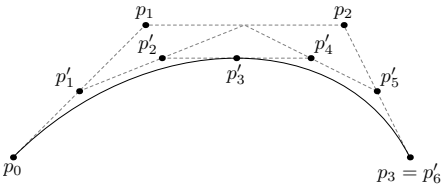


Figure 8.3: Subdiv. graph

Here is a reference to this image: Figure 8.3. Note that `echo=FALSE` is specified so that the **R** code is hidden in the document.

More Figure Stuff

Lastly, we will explore how to rotate and enlarge figures using the `out.extra` chunk option. (Currently this only works in the PDF version of the book.)

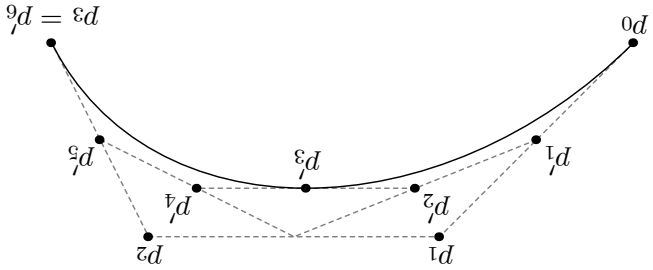


Figure 8.4: A Larger Figure, Flipped Upside Down

As another example, here is a reference: Figure 8.4.

8.3 Footnotes and Endnotes

You might want to footnote something.[1] The footnote will be in a smaller font and placed appropriately. Endnotes work in much the same way.

8.4 Bibliographies

Of course you will need to cite things, and you will probably accumulate an armful of sources. There are a variety of tools available for creating a bibliography database (stored with the .bib extension). In addition to BibTeX suggested below, you may want to consider using the free and easy-to-use tool called Zotero. The Duke librarians have created Zotero documentation at `https://library.duke.edu/`

[1] footnote text

`research/zotero`. In addition, a tutorial is available from Middlebury College at `http://sites.middlebury.edu/zoteromiddlebury/`.

R Markdown uses *pandoc* (`http://pandoc.org/`) to build its bibliographies. One nice caveat of this is that you won't have to do a second compile to load in references as standard LaTeX requires. To cite references in your thesis (after creating your bibliography database), place the reference name inside square brackets and precede it by the "at" symbol. For example, here's a reference to a book about worrying: (Molina & Borkovec, 1994). This `Molina1994` entry appears in a file called `thesis.bib` in the `bib` folder. This bibliography database file was created by a program called BibTeX. You can call this file something else if you like (look at the YAML header in the main .Rmd file) and, by default, is to placed in the `bib` folder.

For more information about BibTeX and bibliographies, see the following documentation from Reed College at (`http://web.reed.edu/cis/help/latex/index.html`)[2]. There are three pages on this topic: *bibtex* (which talks about using BibTeX, at `http://web.reed.edu/cis/help/latex/bibtex.html`), *bibtexstyles* (about how to find and use the bibliography style that best suits your needs, at `http://web.reed.edu/cis/help/latex/bibtexstyles.html`) and *bibman* (which covers how to make and maintain a bibliography by hand, without BibTeX, at `http://web.reed.edu/cis/help/latex/bibman.html`). The last page will not be useful unless you have only a few sources.

If you look at the YAML header at the top of the main .Rmd file you can see that we can specify the style of the bibliography by referencing the appropriate csl file. You can download a variety of different style files at `https://www.zotero.org/styles`. Make sure to download the file into the csl folder.

Tips for Bibliographies

- Like with thesis formatting, the sooner you start compiling your bibliography for something as large as thesis, the better. Typing in source after source is mind-numbing enough; do you really want to do it for hours on end in late April? Think of it as procrastination.
- The cite key (a citation's label) needs to be unique from the other entries.
- When you have more than one author or editor, you need to separate each author's name by the word "and" e.g. `Author = {Noble, Sam and Youngberg, Jessica},`.
- Bibliographies made using BibTeX (whether manually or using a manager) accept LaTeX markup, so you can italicize and add symbols as necessary.
- To force capitalization in an article title or where all lowercase is generally used, bracket the capital letter in curly braces.

8.5 Anything else?

If you'd like to see examples of other things in this template, please contact Mine Cetinkaya-Rundel (email `mine@stat.duke.edu`) with your suggestions. We love to

[2]Reed College (2007)

see people using *R Markdown* for their theses, and are happy to help.

Chapter 9

Organization

Your paper should be an evolving report on the project in all aspects developed so far, in the form of a draft scientific paper. It must be written in RMarkdown or LaTeX with figures included.

Make sure all figures have font sizes and line widths set so that the final pdf versions are properly legible. Presentation (including correctness of mathematical equations, graphics, tables, citations and bibligraphy, as well as prose) should be pristine.

All details of developments of models, code and examples/analyses must be clearly described – sufficient to that a knowledgeable reader will be able to follow the logic and replicate the analysis.

By the end of the first (Fall) semester, you need to develop a readable interim report. In the second (Spring) semester your task is to evolve this paper into a complete write-up of your work, as if intending to consider submitting to a scientific journal.

However primitive the content may seem to be at the start, start writing.

A fairly standard outline is as follows:

- **Title**
- **Abstract**
- **Chapter 1. Introduction** (setting, problem description, citations, etc.)
- **Chapter 2. Literature review**
- **A Next Chapter:** Some papers have one or two chapters, some papers have several. Keep chapters relatively short: Each section should have one focus. For example,
  - **Chapter 3. New Statistical Models** (theory, ideas)
  - **Chapter 4. Some Computational Issues**
  - **Chapter 5. Simulated Data** (evaluation of models)
  - **Chapter 6. Application** (real motivating problem and data)
  - ...
- **Chapter X. Conclusion** (what was done, what was learned, what was good/bad, where research might or could go next)
- **Appendix** (maybe some extra math, details of code)

- **Bibliography** (use bibtex, per the example bib files)

Conclusion

If we don't want Conclusion to have a chapter number next to it, we can add the `{-}` attribute.

More info

And here's some other random info: the first paragraph after a chapter title or section head *shouldn't be* indented, because indents are to tell the reader that you're starting a new paragraph. Since that's obvious after a chapter or section title, proper typesetting doesn't add an indent there.

Appendix A

The First Appendix

This first appendix includes all of the R chunks of code that were hidden throughout the document (using the `include = FALSE` chunk tag) to help with readibility and/or setup.

In the main Rmd file

```
# This chunk ensures that the thesisdowndss package is
# installed and loaded. This thesisdowndss package includes
# the template files for the thesis.
if(!require(devtools))
  install.packages("devtools", repos = "http://cran.rstudio.com")
if(!require(thesisdowndss))
  devtools::install_github("mine-cetinkaya-rundel/thesisdowndss")
library(thesisdowndss)
```

In Chapter 8:

```
# This chunk ensures that the thesisdowndss package is
# installed and loaded. This thesisdowndss package includes
# the template files for the thesis and also two functions
# used for labeling and referencing
if(!require(devtools))
  install.packages("devtools", repos = "http://cran.rstudio.com")
if(!require(dplyr))
    install.packages("dplyr", repos = "http://cran.rstudio.com")
if(!require(ggplot2))
    install.packages("ggplot2", repos = "http://cran.rstudio.com")
if(!require(ggplot2))
    install.packages("bookdown", repos = "http://cran.rstudio.com")
if(!require(thesisdowndss)){
  library(devtools)
  devtools::install_github("mine-cetinkaya-rundel/thesisdowndss")
  }
```

```
library(thesisdowndss)
flights <- read.csv("data/flights.csv")
```

Appendix B

The Second Appendix, for Fun

References

Angel, E. (2000). *Interactive computer graphics : A top-down approach with opengl*. Boston, MA: Addison Wesley Longman.

Angel, E. (2001a). *Batch-file computer graphics : A bottom-up approach with quicktime*. Boston, MA: Wesley Addison Longman.

Angel, E. (2001b). *Test second book by angel*. Boston, MA: Wesley Addison Longman.

Molina, S. T., & Borkovec, T. D. (1994). The Penn State worry questionnaire: Psychometric properties and associated characteristics. In G. C. L. Davey & F. Tallis (Eds.), *Worrying: Perspectives on theory, assessment and treatment* (pp. 265–283). New York: Wiley.

Reed College. (2007, March). LaTeX your document. Retrieved from `http://web.reed.edu/cis/help/LaTeX/index.html`