STA5092Z – Exploratory Data Analysis

Masters in Science – Data Science Specialisation Program

Mr Stefan Britz

Dr. Şebnem Er

University of Cape Town

Department of Statistical Sciences

Stefan.Britz@uct.ac.za

Sebnem.Er@uct.ac.za – https://sebnemer.github.io/

Introduction

Introduction - Lecturers

Mr Stefan Britz
Stefan.Britz@uct.ac.za
Github page
UCT Research Profile

Dr. Şebnem Er
Sebnem.Er@uct.ac.za
Github page
Google Scholar page

Introduction - Structure of the Course

- First 4 weeks of the first semester
- 24 hours of lectures - in classroom environment Mon/Wed/Fri 4-6pm.
- 8 lectures by Dr Şebnem Er, and 16 lectures by Mr Stefan Britz

Introduction - Grading

There are 3 assignments during the 4 weeks of lectures.
Your final mark will be a weighted average of these assignments.

There will be a late hand-in penalty.

Introduction - Lecture Philosophy

- The idea is for the course to be very applied and hands-on, and based around worked examples and case studies, with a focus on the code used to generate an analysis.
- The course is entirely in R, and the goal of the course is as much to introduce students to R and develop R skills as to cover the theory of EDA – which is after all relatively simple.

Introduction - Resources

- All material available on VULA
- Exploratory Data Analysis, 1977 Addison-Wesley Publishing Company (John Wilder Tukey)
- Exploratory Data Analysis: Past, Present and Future, 1993 Technical Report (John W Tukey) https://apps.dtic.mil/dtic/tr/fulltext/u2/a266775.pdf
- How to look at data: A review of John W. Tukey's Exploratory Data Analysis, 1979 (Russell M. Church) https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1332871/pdf/jeab0140.pdf

Introduction - Resources cont.

- RP: Exploratory Data Analysis with R (Roger Peng): https://bookdown.org/rdpeng/exdata/
- JB: STA545: Data wrangling, exploration, and analysis with R (Jenny Bryan): https://stat545.com/index.html
- HW: R for data science (Hadley Wickham): https://r4ds.had.co.nz/

Introduction - Lecture Outline

Lecture	Material	Possible resources
1	R installation, basics, workflows	JB2
2	visualizing raw data with ggplot	HW3, RP6
3	managing data frames with dplyr	JB5-7, HW5, RP3
4	(filter, select, arrange, mutate, summarize)	Some from DSFI
5	EDA checklist	RP4, HW7
6	(right questions, correlation, missing values, outliers)	
7	Reshaping, tidying, joining dataframes	JB8, JB14-16, HW12-13
8	(tidyr, more dplyr)	
9-10	Principles of good graphics	JB24-27, RP5, RP14-15
11-12	Exploring time series data	JB10-13, HW14-16
13	Exploring spatial data	
14	(mapview, leaflet, sf)	
15	Functional programming with R	
16	(writing functions, purrr)	
17	Visualising data using animations	RP11-12
18	(gganimate)	
19-20	Version control with Git and GitHub	JB3, some from DSFI
21	Dashboards	JB42
22	(flexdashboard, shiny)	
23-24	Dimensionality reduction and Clustering	RP13

Lecture 1 - 2

Software - R

- Download R, RStudio, later on Git, account on Github
- Version check

Software - Why R?

- Both R and Python are free.
- R already has all of the statistics support because it was developed by statisticians for statisticians. A lot of statistical modelling research is conducted in R.
- Python was originally developed as a programming language for software development, DS tools (`scikit-learn`, `pandas`, `numpy`) were added on. Though the majority of DL research is done in Python, such as `keras`, `PyTorch`
- R has `Tidyverse`, a set of packages that makes it easy to import, manipulate, visualise and report data.
- Very easy to generate dashboards using R Shiny.

Ref: Python vs. R for Data Science: What's the Difference (by Richie Cotton)

Ref: R vs Python for Data Science: The Winner is... (by Martijn Theuwissen)

Software - Why R? cont.

- It is the language I know the best, I know very little Python.
- Python and R programmers get inspired from each other, ie. Python's `plotnine` inspired by R's `ggplot2`, and R's `rvest` by Python's `BeautifulSoup`.
- You can also use functions written in Python with `source_python()` function in R.
- You can run R code from Python with `rp2` package, and you can run Python code from R using `reticulate`. R version of DL package `Keras` calls Python.
- Though, I do encourage you to learn Python as well. No harm in two languages.

Take away: There is no winner, you are here to learn the skills, your focus should be on skills. If you can program in R, you can do it in any other language.

Function	Description
`version$version.string`	which version of R
`RStudio.Version()$version`	which version of RStudio
`objects()`	list of objects in your environment
`ls()`	list of objects in your environment
`rm(list = ls())`	remove everything from your enviro
`getwd()`	what is your working directory
`setwd(C:/EDA)`	

Dataset resources

Before you can work with data you have to get some data:

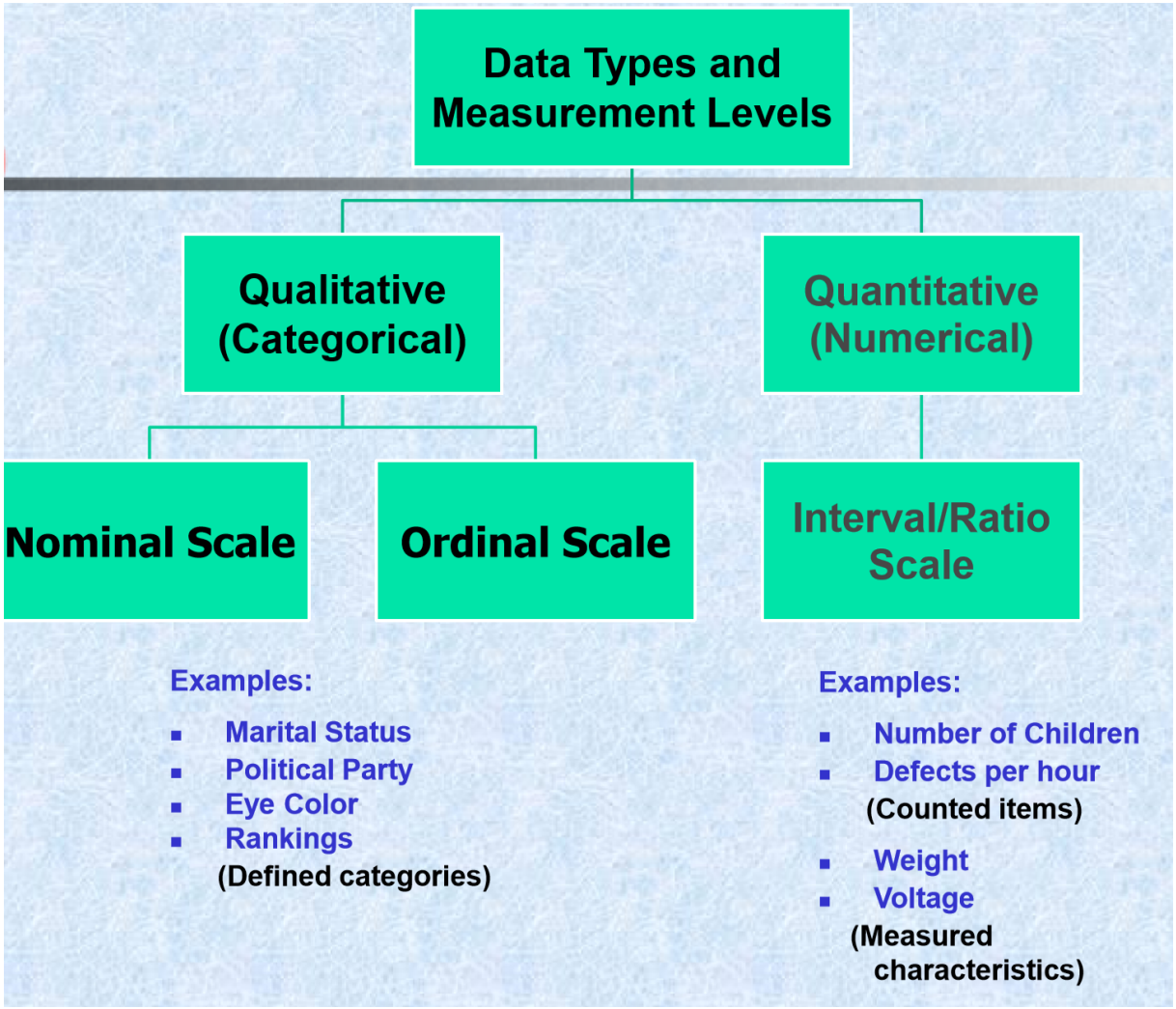
Zindi Competitions: https://zindi.africa/competitions

Zindi is the first data science competition platform in Africa.

Zindi hosts an entire data science ecosystem of scientists, engineers, academics, companies, NGOs, governments and institutions focused on solving Africa's most pressing problems.

Kaggle competitions: https://www.kaggle.com/competitions

Dataset types - Measurement levels



Dataset file formats

- Raw files (.csv,.txt, .xlsx, .sav, etc.)
- Databases (mySQL, MongoDB)
- APIs (Twitter)

How do we clean data, how do we tidy data, how do we share data? This course will cover all these as well as exploring the data.

Tukey's EDA book and terminology

About the book:

- The book provides techniques and advice about how to explore data.
- The approach of EDA is detective in character, it is a search for clues. Some of the clues may be misleading, but some will lead to discoveries.

Tukey's EDA book and terminology

About the author, Tukey:

- Tukey favors simplicity because simple statements are clear.
- Tukey favors clear visual displays of quantitative facts.
- Tukey likes precision, it is far better to be able to say some response measure is a linear function of a particular stimulus variable than to say it increases with the stimulus variable.
- Tukey favors depth of analysis. It is always good to look at the residuals.
- Tukey values accuracy. A misplaced decimal vs a misplaced digit.
- Tukey values replicability of summary observations in situations containing aberrant observations.

Tukey's EDA book and terminology

The methods describes in the book:

- Frequency distributions
- Measures of central tendency and variability
- Scale transformations
- Graphical displays of single variable and of relationships
- Smoothing techniques
- Analysis of tables

Graphs

Tukey's approach to data analysis is highly visual and he has numerous suggestions for graphical displays. Tukey emphasizes the value of graphs for the following:

- Graphs can be used to store quantitative data,
- Graphs can be used to communicate conclusions,
- Graphs can be used to discover new information. Tukey particularly emphasizes the value of graphs for discovery.

Some types of plots are better for one purpose, others are better for another.

Distribution of a Single Quantitative Variable

Tukey's novel distribution tools:

- Stem and Leaf: The measures Tukey proposes involve no arithmetic, only counting. - not widely used.
- Histogram: One should note,
  - its height,
  - where it is centered,
  - how spread out it is,
  - whether it is asymmetric,
  - whether there are any discontinuities.

- Box-Whisker Plots: These plots show medians, quartiles, and two extreme values in a format that is easy to grasp quickly. Very powerful when comparing several frequency distributions.
- Q-Q plots

Visual Display of a Single Qualitative Variable

Relative Frequency Distribution - compare percentages and probability

- Frequency distribution
- Pie chart
- Bar chart

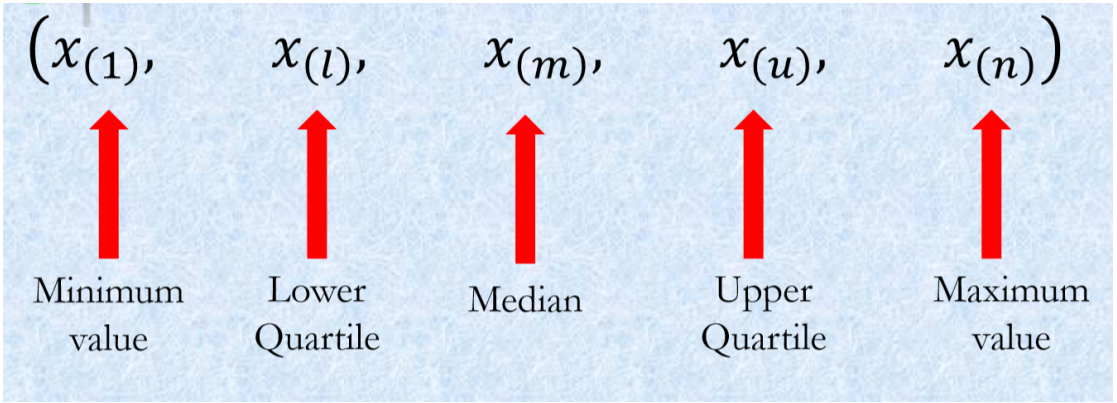
Summarizing data

In order to visually display data, sometimes we need to summarize our data with summary measures:

- Compute a few "key" numbers: 5 number summaries.
- Our aim is to reduce a large data set to a few numbers which will help us understand the important features of the data.
- Let's begin with the concept of ranked data.
- In a sample of size n, the smallest number has a rank of 1; the second smallest number has a rank of 2; .... ; the largest number has a rank of n.

  $x_1, x_2, x_3, ..., x_n$ and $x_{(r)}$ is the number with rank r.

Summarizing data - cont.



Which transformations and Why?

Very often it is more convenient to look at some transform of the original variable. If the distribution is far from symmetrical, one end of the distribution will be too crowded to permit careful inspection.
Tukey deals extensively with scale transformations. He gives three main reasons for transformations:

- A transformation may be selected to produce a symmetrical distribution,
- A transformation may increase the similarity of the spread of the different sets of numbers,
- A transformation may straighten out a line.

Types of transformations

The transformations discussed range on:

x^n

x^{n-1}

x^2

$\log x$

$-\frac{1}{x}$

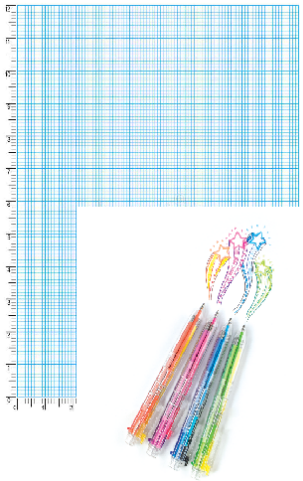
$-\frac{1}{x^2}$

$-\frac{1}{x^n}$

Other dependent variables, e.g. counts and latencies, are occasionally transformed by taking the square root, the logarithm, or the reciprocal.

Data Science Toolbox

Visualising plays an important role in exploring your data, and you would know that Tukey favours analysis of data with four-color pen, graph paper, few tables etc.:



Though we will use R and its functions for this purpose:

- ggplot2
- tidyr
- dplyr

Example 1 - Female headed households in SA

Let us look at the "Womxn in Big Data South Africa: Female-Headed Households in South Africa" competition at https://zindi.africa/competitions/womxn-in-big-data-south-africa-female-headed-households-in-south-africa

The datasets are provided in a .csv file format, test.csv, train.csv, variable_descriptions.csv.

The target variable of interest is the percentage of households per ward that are both female-headed and earn an annual income that is below R19,600 (approximately $2,300 USD in 2011).

```
> Train <- read.csv("C:/Users/01438475/Google Drive/
    UCTcourses/DataScience/LectureNotes/EDA/Datasets/Woman/
    Train.csv", header = TRUE)
> str(Train)
> summary(Train)
```

Now we will explore this dataset.

tidyverse ecosystem

The majority of the packages that you will use are part of the so-called tidyverse package:

```
> install.packages("tidyverse")
> library(tidyverse)
```

```
> Attaching packages tidyverse 1.2.1
> ggplot2 3.2.1     purrr   0.3.3
> tibble  2.1.3     dplyr   0.8.3
> tidyr   1.0.0     stringr 1.4.0
> readr   1.3.1     forcats 0.4.0
> Conflicts tidyverse_conflicts()
> dplyr::filter() masks stats::filter()
> dplyr::lag()    masks stats::lag()
```

This tells you that tidyverse is loading the ggplot2, tibble, tidyr, readr, purrr, and dplyr packages.

ggplot2 functions1

ggplot2

part of the tidyverse

3.2.1

`geom_abline()` `geom_hline()` `geom_vline()`	Reference lines: horizontal, vertical, and diagonal
`geom_bar()` `geom_col()` `stat_count()`	Bar charts
`geom_bin2d()` `stat_bin_2d()`	Heatmap of 2d bin counts
`geom_blank()`	Draw nothing
`geom_boxplot()` `stat_boxplot()`	A box and whiskers plot (in the style of Tukey)
`geom_contour()` `stat_contour()`	2d contours of a 3d surface
`geom_count()` `stat_sum()`	Count overlapping points
`geom_density()` `stat_density()`	Smoothed density estimates
`geom_density_2d()` `stat_density_2d()`	Contours of a 2d density estimate

ggplot2 functions2

ggplot2

part of the tidyverse

3.2.1

`geom_dotplot()`	Dot plot
`geom_errorbarh()`	Horizontal error bars
`geom_hex()` `stat_bin_hex()`	Hexagonal heatmap of 2d bin counts
`geom_freqpoly()` `geom_histogram()` `stat_bin()`	Histograms and frequency polygons
`geom_jitter()`	Jittered points
`geom_crossbar()` `geom_errorbar()` `geom_linerange()` `geom_pointrange()`	Vertical intervals: lines, crossbars & errorbars
`geom_map()`	Polygons from a reference map
`geom_path()` `geom_line()` `geom_step()`	Connect observations

ggplot2 functions3

ggplot2

part of the tidyverse

3.2.1

Refere

`geom_point()`	Points
`geom_polygon()`	Polygons
`geom_qq_line()` `stat_qq_line()` `geom_qq()` `stat_qq()`	A quantile-quantile plot
`geom_quantile()` `stat_quantile()`	Quantile regression
`geom_ribbon()` `geom_area()`	Ribbons and area plots
`geom_rug()`	Rug plots in the margins
`geom_segment()` `geom_curve()`	Line segments and curves
`geom_smooth()` `stat_smooth()`	Smoothed conditional means
`geom_spoke()`	Line segments parameterised by location, direction and distance
`geom_label()` `geom_text()`	Text

ggplot2 functions4

ggplot2

part of the tidyverse

3.2.1

`geom_raster()` `geom_rect()` `geom_tile()` — Rectangles

`geom_violin()` `stat_ydensity()` — Violin plot

`coord_sf()` `geom_sf()` `geom_sf_label()` `geom_sf_text()` `stat_sf()` — Visualise sf objects

Example 1 with ggplot2


```
> library(ggplot2)
```

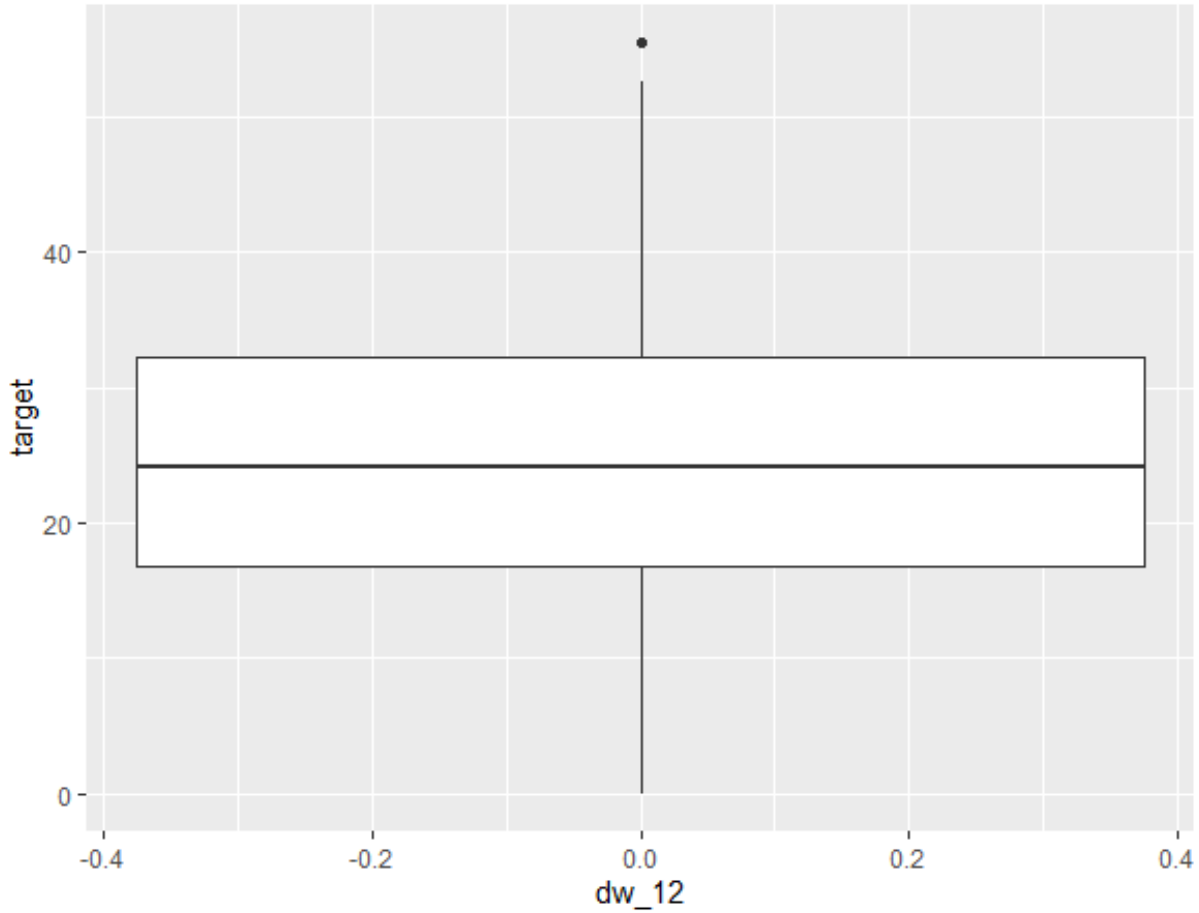
```
Warning message:
package 'ggplot2' was built under R version 3.6.1
```

```
> R.version.string
```

```
[1] "R version 3.6.0 (2019-04-26)"
```

Example 1 with ggplot2 - boxplot

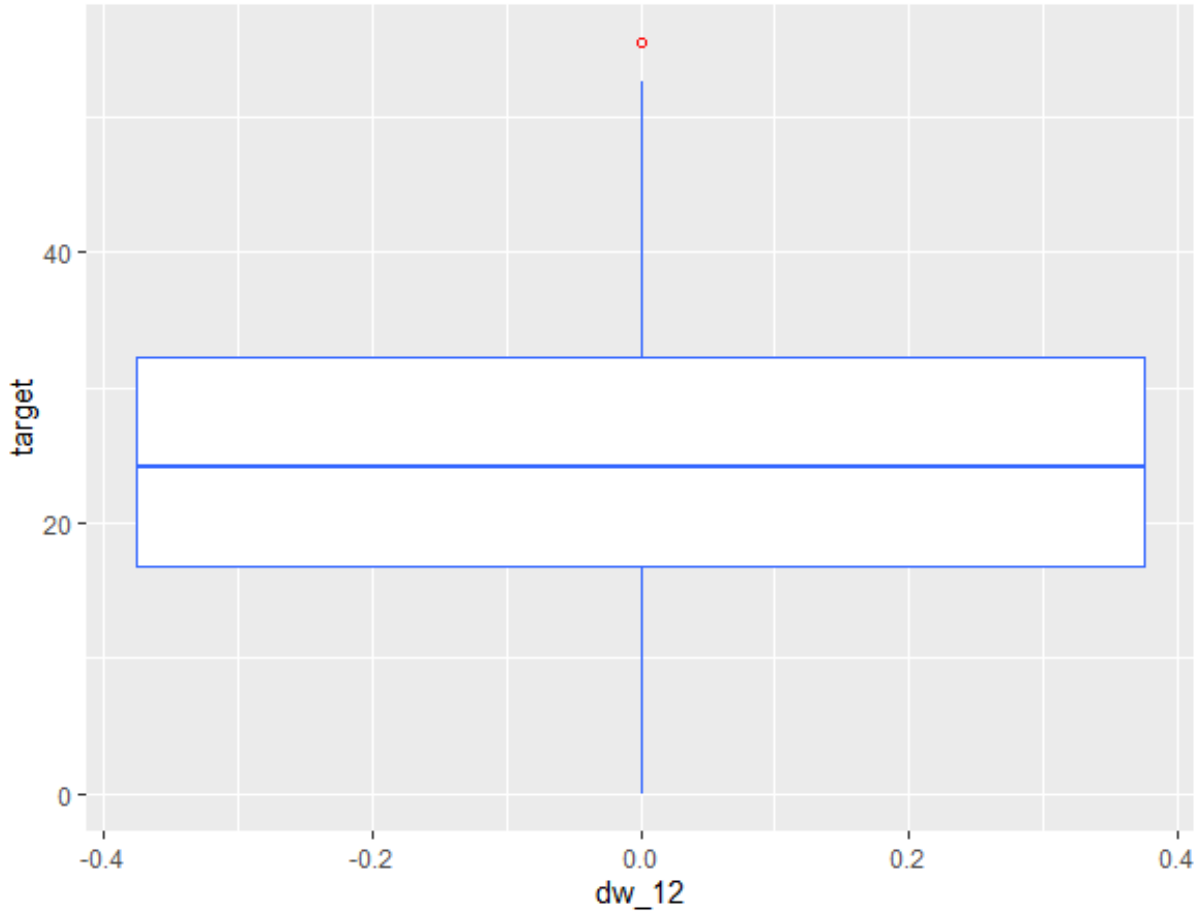
```
> plot1 = ggplot(data = Train, aes(dw_12,target))
> plot1 + geom_boxplot()
```



How to interpret this: IntroStat p:21-26

Example 1 with ggplot2 - boxplot options

```
> plot1 + geom_boxplot(fill = "white", colour = "#3366FF",
    outlier.colour = "red", outlier.shape = 1)
```



Example 1 with ggplot2 - boxplot options

A cheat sheet:

https://rstudio.com/wp-content/uploads/2016/11/ggplot2-cheatsheet-2.1.pdf

https://ggplot2.tidyverse.org/reference/

ggplot2: Elegant Graphics for Data Analysis (Use R!) by (Hadley Wickham)

Example 2

Rmd file... Reproducible results, don't save output!

Lecture 2 - 3

Getting and Cleaning Data

```
> install.packages("swirl")
> packageVersion("swirl") # make sure it is at least 2.2.21
> library(swirl)
> install_from_swirl("Getting and Cleaning Data")
> swirl
```

```
[1] "R version 3.6.0 (2019-04-26)"
```