Documentation on using ioslides is available here: http://rmarkdown.rstudio.com/ioslides_presentation_format.html Some slides are adopted (or copied) from OpenIntro: https://www.openintro.org/

Announcements

- Slack Channel: https://data606fall2018.slack.com
  - Click here to join the group - You must click this link as it serves as the invitation to the channel.
- Completing labs - You may submit a PDF (even if created from the browser) or provide a link to Rpubs. Blackboard will not let you submit HTML files.
- Working Directories - See this page: http://data606.net/post/2018-08-28-getting-started-with-r/
- You can view the RMarkdown source for the Meetup slides here: https://github.com/jbryer/DATA606Fall2018/tree/master/Slides

Intro to Data

We will use the `lego` R package in this class which contains information about every Lego set manufactured from 1970 to 2014, a total of 5710 sets.

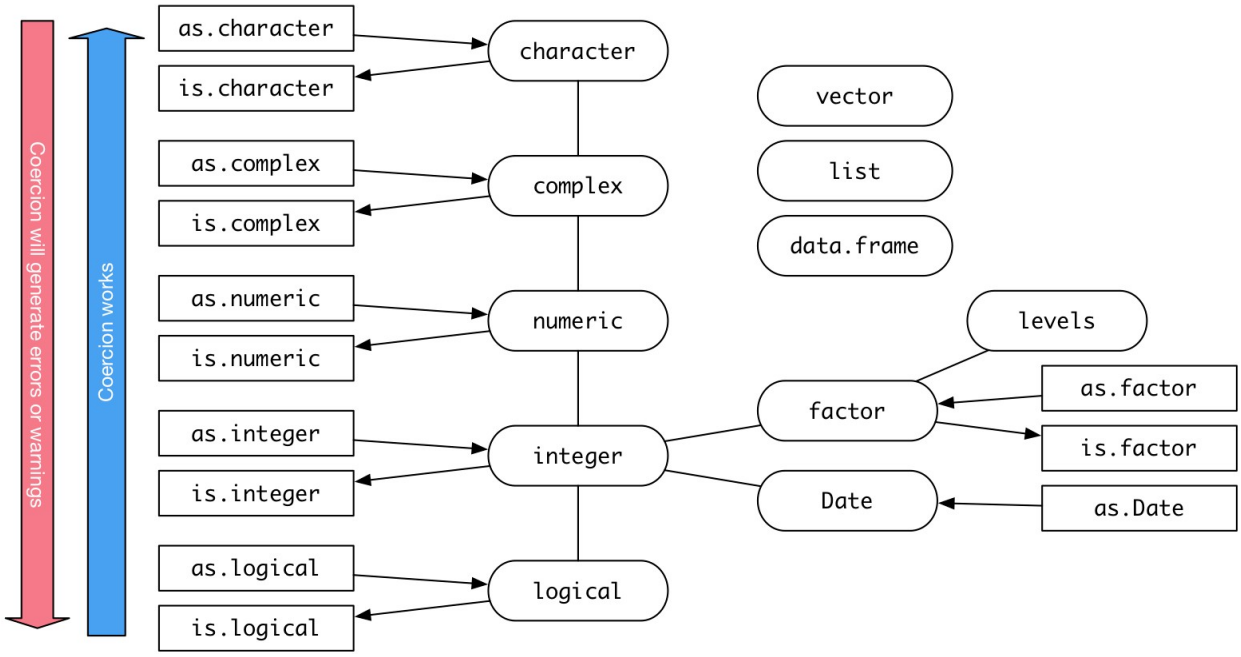
```
devtools::install_github("seankross/lego")
```

```
library(lego)
data(legosets)
```

Types of Variables

- Numerical (quantitative)
  - Continuous
  - Discrete
- Categorical (qualitative)
  - Regular categorical
  - Ordinal

Data Types in R



Types of Variables

```
str(legosets)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    6172 obs. of  14 variables:
##  $ Item_Number : chr  "10246" "10247" "10248" "10249" ...
##  $ Name        : chr  "Detective's Office" "Ferris Wheel" "Ferrari F40" "Toy Shop" ...
##  $ Year        : int  2015 2015 2015 2015 2015 2015 2015 2015 2015 2015 ...
##  $ Theme       : chr  "Advanced Models" "Advanced Models" "Advanced Models" "Advanced Models" ...
##  $ Subtheme    : chr  "Modular Buildings" "Fairground" "Vehicles" "Winter Village" ...
##  $ Pieces      : int  2262 2464 1158 898 13 39 32 105 13 11 ...
##  $ Minifigures : int  6 10 NA NA 1 2 2 3 2 2 ...
##  $ Image_URL   : chr  "http://images.brickset.com/sets/images/10246-1.jpg" "http://images.brickset.com/sets/image
##  $ GBP_MSRP    : num  132.99 149.99 69.99 59.99 9.99 ...
##  $ USD_MSRP    : num  159.99 199.99 99.99 79.99 9.99 ...
##  $ CAD_MSRP    : num  200 230 120 NA 13 ...
##  $ EUR_MSRP    : num  149.99 179.99 89.99 69.99 9.99 ...
##  $ Packaging   : chr  "Box" "Box" "Box" "Box" ...
##  $ Availability: chr  "Retail - limited" "Retail - limited" "LEGO exclusive" "LEGO exclusive" ...
```

Qualitative Variables

Descriptive statistics:

- Contingency Tables
- Proportional Tables

Plot types:

- Bar plot
- Mosaic plot

Contingency Tables

```
table(legosets$Availability, useNA='ifany')
```

```
##
##        LEGO exclusive    LEGOLAND exclusive         Not specified
##                   695                     2                  1795
##           Promotional Promotional (Airline)                Retail
##                   141                    12                  3120
##      Retail - limited               Unknown
##                   403                     4
```

```
table(legosets$Availability, legosets$Packaging, useNA='ifany')
```

```
##
##                         Blister pack  Box Box with backing card Bucket
##   LEGO exclusive                  45  147                     0      1
##   LEGOLAND exclusive               0    2                     0      0
##   Not specified                    0   20                     0      0
##   Promotional                      0   44                     0      0
##   Promotional (Airline)            0   11                     0      0
##   Retail                          53 2575                    16     30
##   Retail - limited                 2  302                     1      5
##   Unknown                          0    1                     0      0
##
##                         Canister Foil pack Loose Parts Not specified Other
##   LEGO exclusive               0         0          71             7     5
##   LEGOLAND exclusive           0         0           0             0     0
##   Not specified                0         5           0          1739     0
##   Promotional                  0         0           1             0     3
##   Promotional (Airline)        0         0           0             1     0
##   Retail                      78       285           0             0    28
##   Retail - limited             0         1           0             0     0
##   Unknown                      0         0           0             0     0
##
##                         Plastic box Polybag Shrink-wrapped  Tag  Tub
##   LEGO exclusive                  1     412              0    6    0
##   LEGOLAND exclusive              0       0              0    0    0
##   Not specified                   6      24              0    0    1
##   Promotional                     2      90              0    0    1
##   Promotional (Airline)           0       0              0    0    0
##   Retail                          0       4             18    0   33
##   Retail - limited                1      86              0    0    5
##   Unknown                         0       3              0    0    0
```

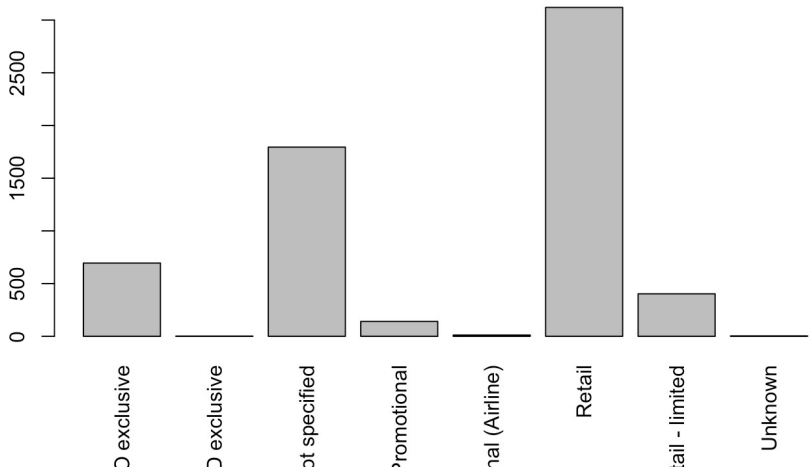
Proportional Tables

```
prop.table(table(legosets$Availability))
```

```
##
##         LEGO exclusive    LEGOLAND exclusive         Not specified
##           0.1126053143          0.0003240441          0.2908295528
##            Promotional Promotional (Airline)                Retail
##           0.0228451069          0.0019442644          0.5055087492
##      Retail - limited               Unknown
##           0.0652948801          0.0006480881
```

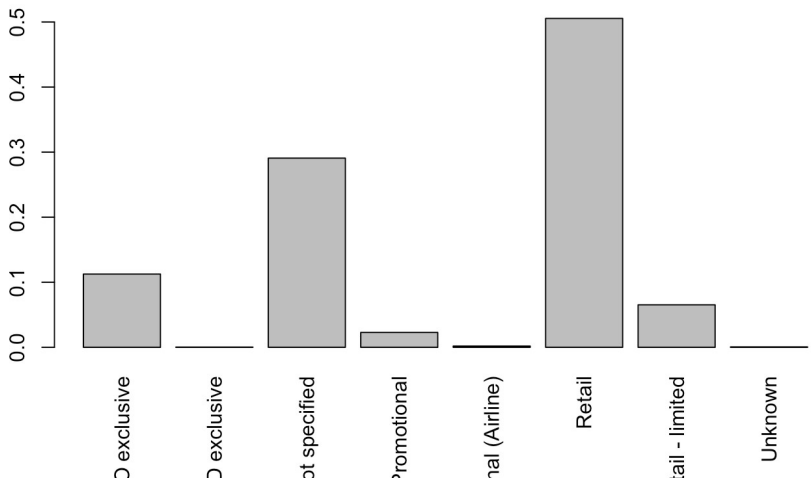
Bar Plots

```
barplot(table(legosets$Availability), las=3)
```



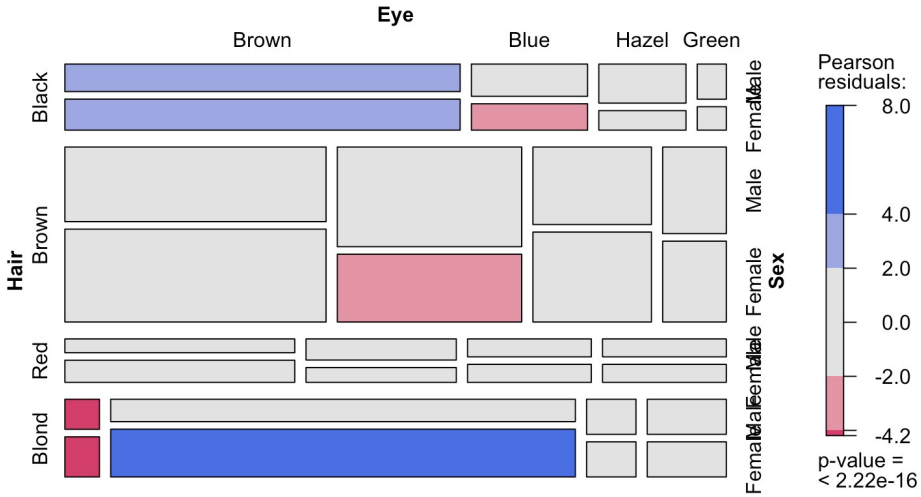
Bar Plots

```
barplot(prop.table(table(legosets$Availability)), las=3)
```



Mosaic Plot

```
library(vcd)
mosaic(HairEyeColor, shade=TRUE, legend=TRUE)
```



Quantitative Variables

Descriptive statistics:

- Mean
- Median
- Quartiles
- Variance: $s^2 = \sum_{i=1}^{n} \frac{\left(x_i - \bar{x}\right)^2}{n-1}$
- Standard deviation: $s = \sqrt{s^2}$

Plot types:

- Dot plots
- Histograms
- Density plots
- Box plots
- Scatterplots

Measures of Center

```
mean(legosets$Pieces, na.rm=TRUE)
```

```
## [1] 215.1686
```

```
median(legosets$Pieces, na.rm=TRUE)
```

```
## [1] 82
```

Measures of Spread

```
var(legosets$Pieces, na.rm=TRUE)
```

```
## [1] 126876.8
```

```
sqrt(var(legosets$Pieces, na.rm=TRUE))
```

```
## [1] 356.1976
```

```
sd(legosets$Pieces, na.rm=TRUE)
```

```
## [1] 356.1976
```

```
fivenum(legosets$Pieces, na.rm=TRUE)
```

```
## [1]    0.0   30.0   82.0  256.5 5922.0
```

```
IQR(legosets$Pieces, na.rm=TRUE)
```

```
## [1] 226.25
```

The `summary` Function

```
summary(legosets$Pieces)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##     0.0    30.0    82.0   215.2   256.2  5922.0     112
```

The psych Package

```
library(psych)
describe(legosets$Pieces, skew=FALSE)
```

```
##    vars    n   mean    sd min  max range   se
## X1    1 6060 215.17 356.2   0 5922  5922 4.58
```

```
describeBy(legosets$Pieces, group = legosets$Availability, skew=FALSE, mat=TRUE)
```

```
##     item               group1 vars    n      mean        sd min  max
## X11    1        LEGO exclusive    1  659 172.74203 442.96954   1 3428
## X12    2    LEGOLAND exclusive    1    2 211.00000 154.14928 102  320
## X13    3         Not specified    1 1747 145.87178 309.19929   1 5195
## X14    4           Promotional    1  140  53.97143 108.42721   1 1000
## X15    5 Promotional (Airline)    1   12 126.16667  47.01612  10  203
## X16    6               Retail    1 3094 245.78119 294.78052   0 3803
## X17    7      Retail - limited    1  402 410.94030 652.06435   1 5922
## X18    8               Unknown    1    4  27.50000  15.96872   6   44
##     range         se
## X11  3427  17.255643
## X12   218 109.000000
## X13  5194   7.397620
## X14   999   9.163772
## X15   193  13.572384
## X16  3803   5.299546
## X17  5921  32.522014
## X18    38   7.984360
```

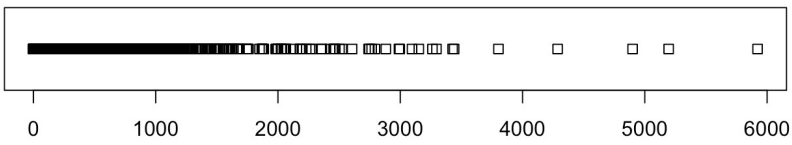
Robust Statistics

Median and IQR are more robust to skewness and outliers than mean and SD. Therefore,

- for skewed distributions it is often more helpful to use median and IQR to describe the center and spread
- for symmetric distributions it is often more helpful to use the mean and SD to describe the center and spread

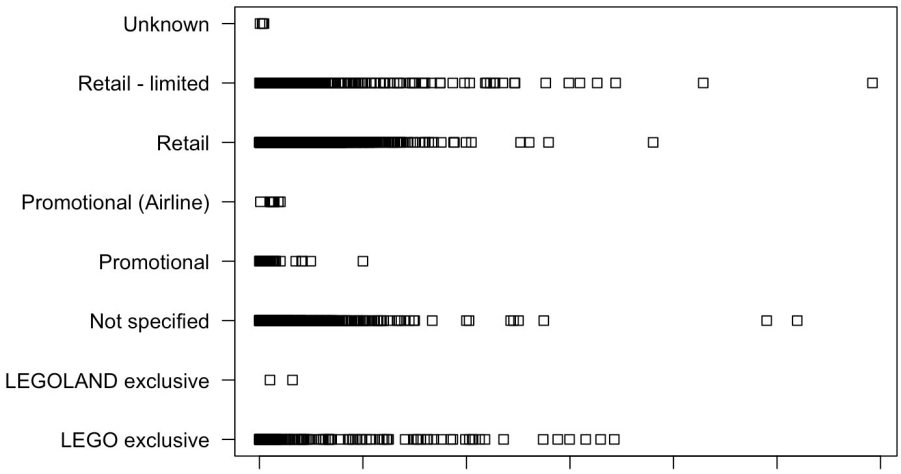
Dot Plot

```
stripchart(legosets$Pieces)
```



Dot Plot

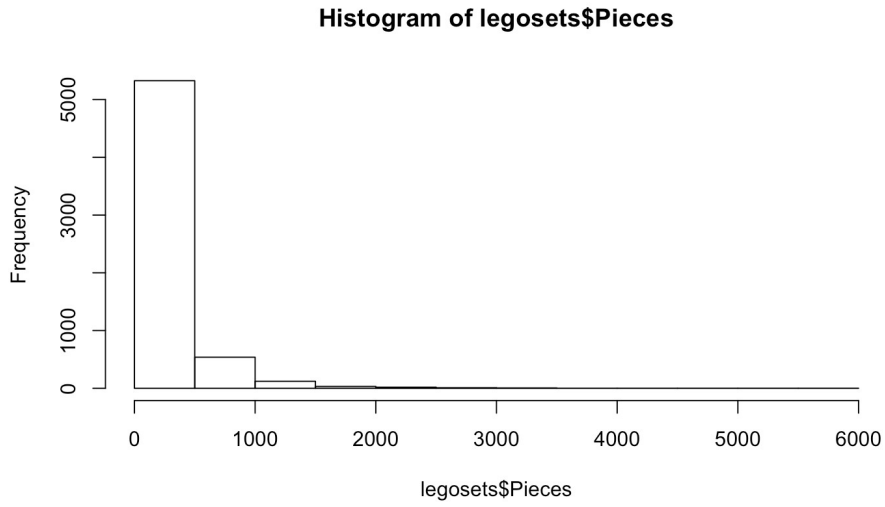
```
par.orig <- par(mar=c(1,10,1,1))
stripchart(legosets$Pieces ~ legosets$Availability, las=1)
```




```
par(par.orig)
```

Histograms

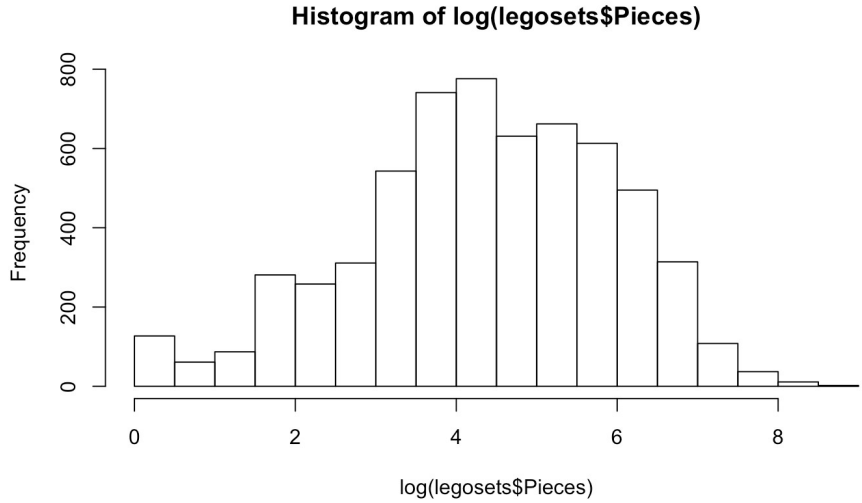
```
hist(legosets$Pieces)
```



Transformations

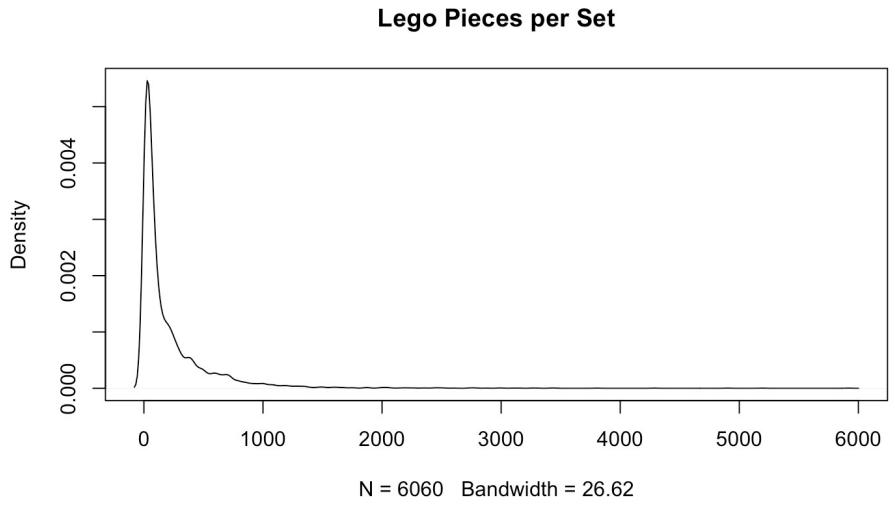
With highly skewed distributions, it is often helpful to transform the data. The log transformation is a common approach, especially when dealing with salary or similar data.

```
hist(log(legosets$Pieces))
```



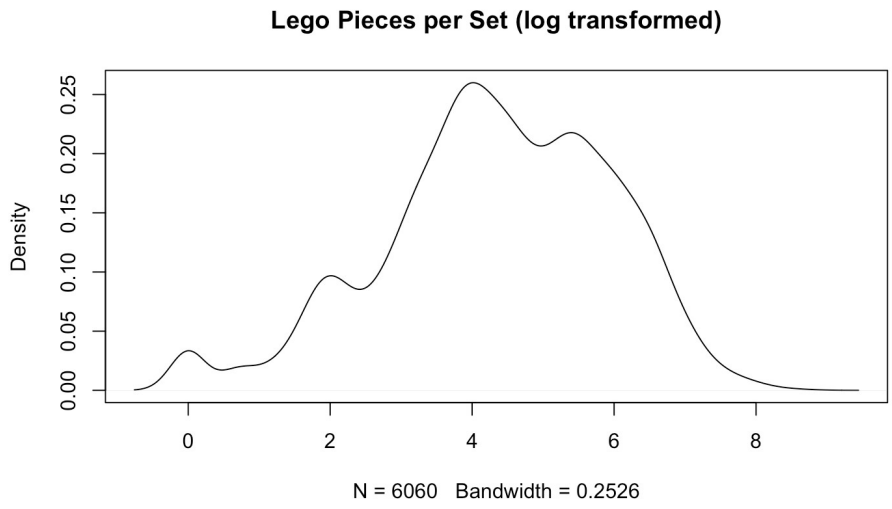
Density Plots

```
plot(density(legosets$Pieces, na.rm=TRUE), main='Lego Pieces per Set')
```



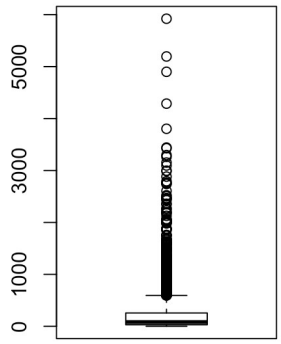
Density Plot (log tansformed)

```
plot(density(log(legosets$Pieces), na.rm=TRUE), main='Lego Pieces per Set (log transformed)')
```

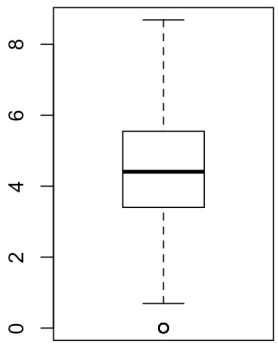


Box Plots

```
boxplot(legosets$Pieces)
```



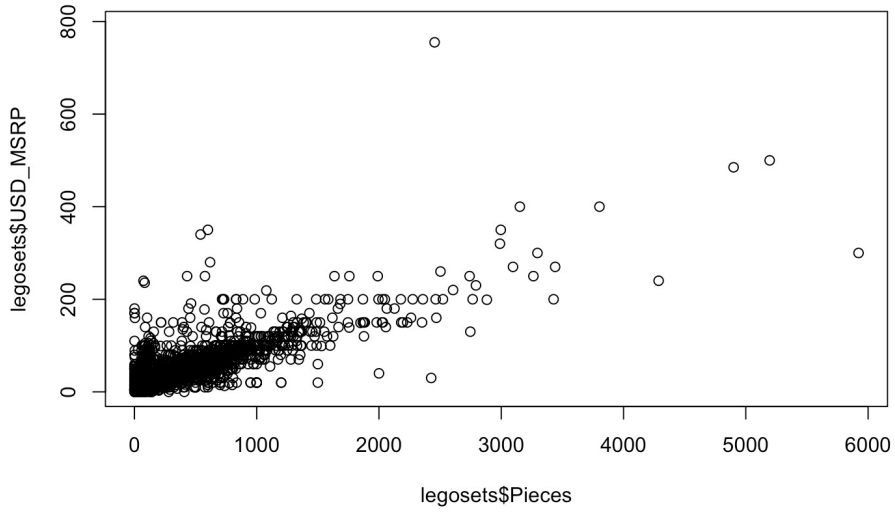

```
## Warning in bplt(at[i], wid = width[i], stats = z$stats[,
## z$out[z$group == : Outlier (-Inf) in boxplot 1 is not dra
```




```
boxplot(log(legosets$Pieces))
```

Scatter Plots

```
plot(legosets$Pieces, legosets$USD_MSRP)
```



Examining Possible Outliers (expensive sets)

```
legosets[which(legosets$USD_MSRP >= 400),]
```

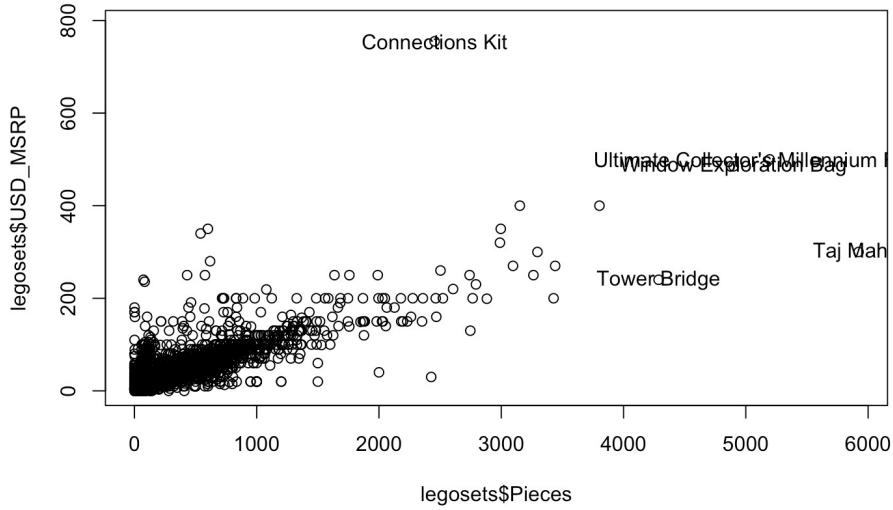
```
## # A tibble: 4 x 14
##   Item_Number Name   Year Theme Subtheme Pieces Minifigures Image_URL
##   <chr>       <chr> <int> <chr> <chr>     <int>       <int> <chr>
## 1 2000430     Iden…  2013 Seri… ""           NA           6 http://i…
## 2 2000431     Conn…  2013 Seri… ""         2455          NA http://i…
## 3 2000409     Wind…  2010 Seri… ""         4900          NA http://i…
## 4 10179       Ulti…  2007 Star… Ultimat…   5195           5 http://i…
## # ... with 6 more variables: GBP_MSRP <dbl>, USD_MSRP <dbl>,
## #   CAD_MSRP <dbl>, EUR_MSRP <dbl>, Packaging <chr>, Availability <chr>
```

Examining Possible Outliers (big sets)

```
legosets[which(legosets$Pieces >= 4000),]
```

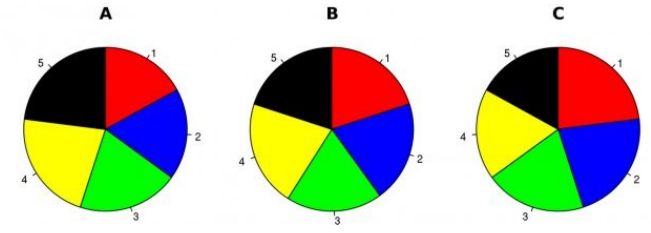
```
## # A tibble: 4 x 14
##   Item_Number Name   Year Theme Subtheme Pieces Minifigures Image_URL
##   <chr>       <chr> <int> <chr> <chr>     <int>       <int> <chr>
## 1 10214       Towe…  2010 Adva… Buildin…   4287          NA http://i…
## 2 2000409     Wind…  2010 Seri… ""         4900          NA http://i…
## 3 10189       Taj …  2008 Adva… Buildin…   5922          NA http://i…
## 4 10179       Ulti…  2007 Star… Ultimat…   5195           5 http://i…
## # ... with 6 more variables: GBP_MSRP <dbl>, USD_MSRP <dbl>,
## #   CAD_MSRP <dbl>, EUR_MSRP <dbl>, Packaging <chr>, Availability <chr>
```

```
plot(legosets$Pieces, legosets$USD_MSRP)
bigAndExpensive <- legosets[which(legosets$Pieces >= 4000 | legosets$USD_MSRP >= 400),]
text(bigAndExpensive$Pieces, bigAndExpensive$USD_MSRP, labels=bigAndExpensive$Name)
```



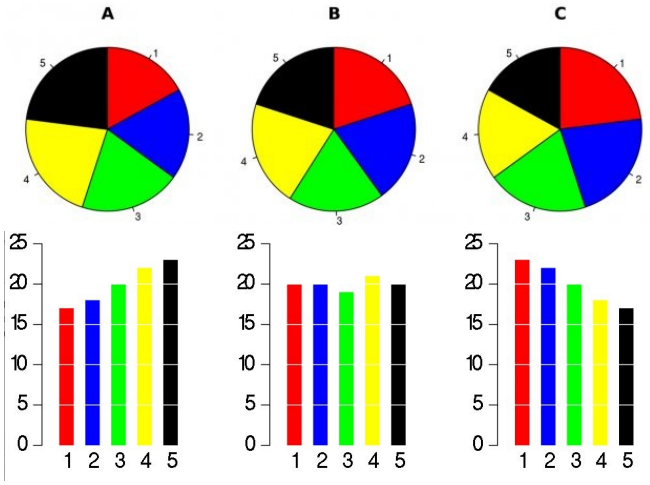
Pie Charts

There is only one pie chart in *OpenIntro Statistics* (Diez, Barr, & Çetinkaya-Rundel, 2015, p. 48). Consider the following three pie charts that represent the preference of five different colors. Is there a difference between the three pie charts? This is probably a difficult to answer.



Pie Charts

There is only one pie chart in *OpenIntro Statistics* (Diez, Barr, & Çetinkaya-Rundel, 2015, p. 48). Consider the following three pie charts that represent the preference of five different colors. Is there a difference between the three pie charts? This is probably a difficult to answer.



Source: https://en.wikipedia.org/wiki/Pie_chart.

Just say NO to pie charts!

"There is no data that can be displayed in a pie chart that cannot better be displayed in some other type of chart"

John Tukey

Sampling vs. Census

A census involves collecting data for the entire population of interest. This is problematic for several reasons, including:

- It can be difficult to complete a census: there always seem to be some individuals who are hard to locate or hard to measure. And these difficult-to-find people may have certain characteristics that distinguish them from the rest of the population.
- Populations rarely stand still. Even if you could take a census, the population changes constantly, so it's never possible to get a perfect measure.
- Taking a census may be more complex than sampling.

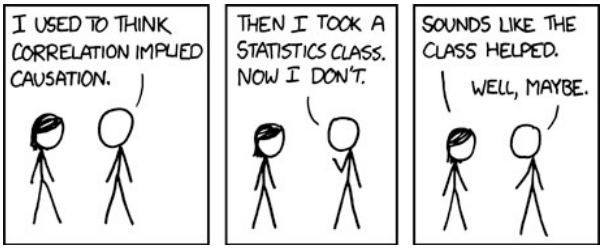
Sampling involves measuring a subset of the population of interest, usually randomly.

Sampling Bias

- **Non-response**: If only a small fraction of the randomly sampled people choose to respond to a survey, the sample may no longer be representative of the population.
- **Voluntary response**: Occurs when the sample consists of people who volunteer to respond because they have strong opinions on the issue. Such a sample will also not be representative of the population.
- **Convenience sample**: Individuals who are easily accessible are more likely to be included in the sample.

Observational Studies vs. Experiments

- **Observational study**: Researchers collect data in a way that does not directly interfere with how the data arise, i.e. they merely "observe", and can only establish an association between the explanatory and response variables.
- **Experiment**: Researchers randomly assign subjects to various treatments in order to establish causal connections between the explanatory and response variables.

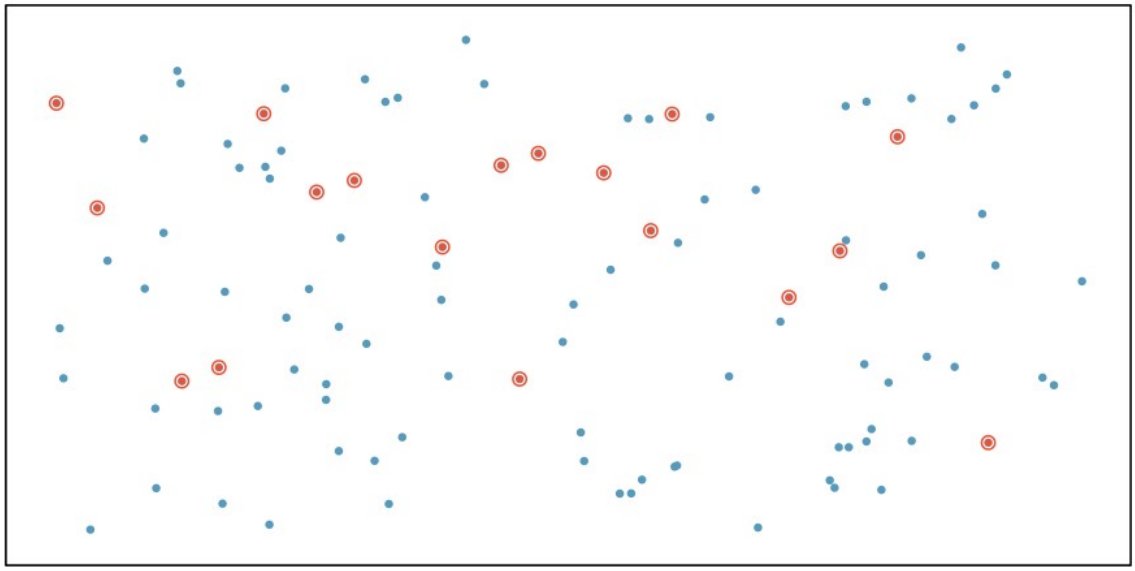


Source: XKCD 552 http://xkcd.com/552/

Correlation does not imply causation!

Simple Random Sampling

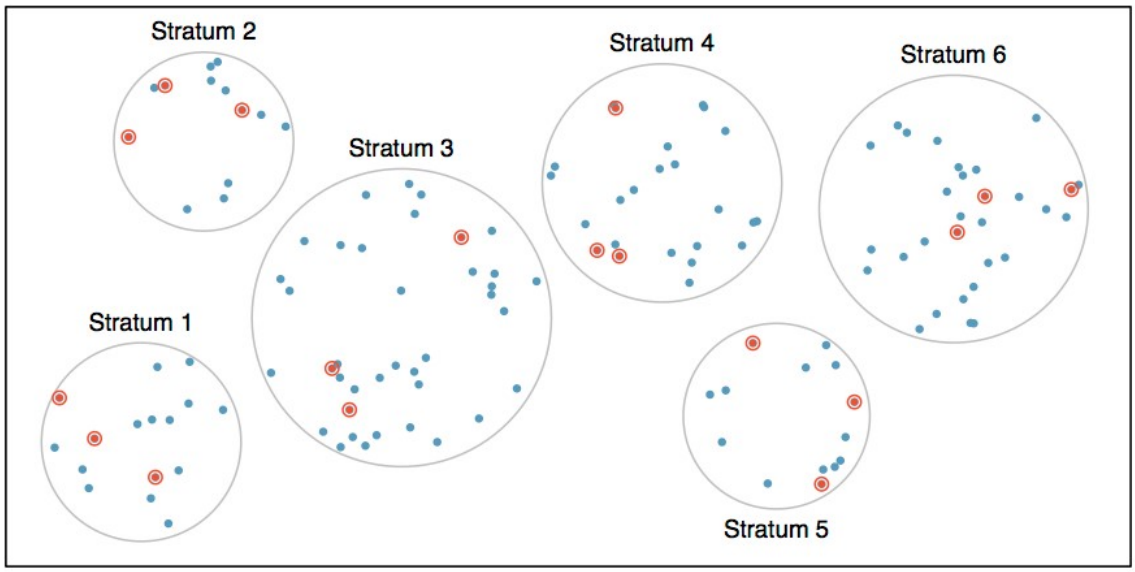
Randomly select cases from the population, where there is no implied connection between the points that are selected.



Simple Random Sample

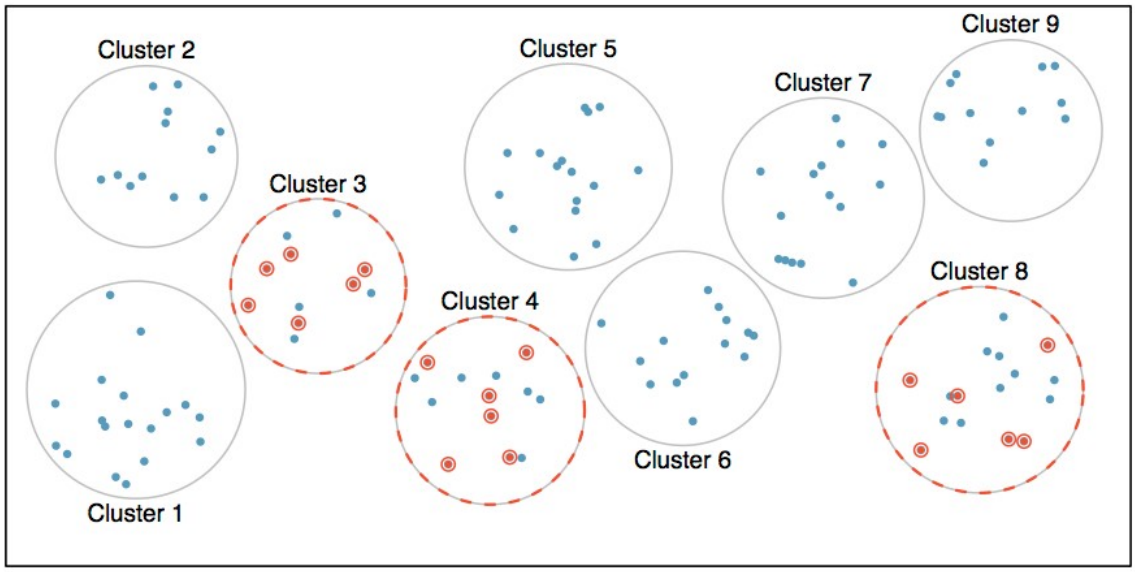
Stratified Sampling

Strata are made up of similar observations. We take a simple random sample from each stratum.



Cluster Sampling

Clusters are usually not made up of homogeneous observations so we take random samples from random samples of clusters.



Principles of experimental design

1. **Control**: Compare treatment of interest to a control group.
2. **Randomize**: Randomly assign subjects to treatments, and randomly sample from the population whenever possible.
3. **Replicate**: Within a study, replicate by collecting a sufficiently large sample. Or replicate the entire study.
4. **Block**: If there are variables that are known or suspected to affect the response variable, first group subjects into blocks based on these variables, and then randomize cases within each block to treatment groups.

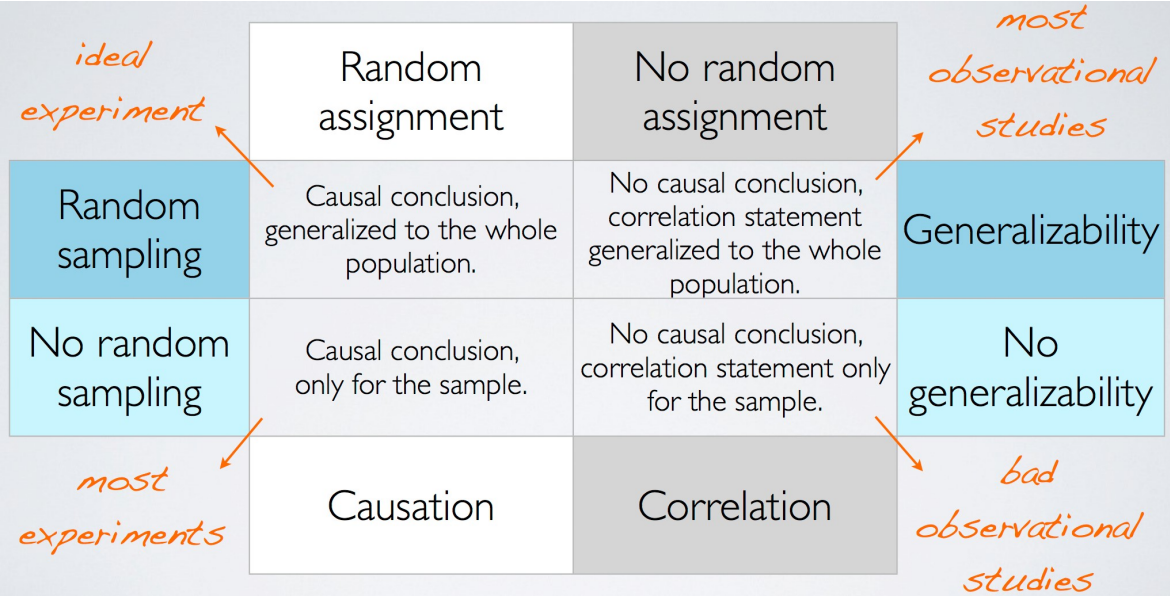
Difference between blocking and explanatory variables

- Factors are conditions we can impose on the experimental units.
- Blocking variables are characteristics that the experimental units come with, that we would like to control for.
- Blocking is like stratifying, except used in experimental settings when randomly assigning, as opposed to when sampling.

More experimental design terminology…

- **Placebo**: fake treatment, often used as the control group for medical studies
- **Placebo effect**: experimental units showing improvement simply because they believe they are receiving a special treatment
- **Blinding**: when experimental units do not know whether they are in the control or treatment group
- **Double-blind**: when both the experimental units and the researchers who interact with the patients do not know who is in the control and who is in the treatment group

Random assignment vs. random sampling

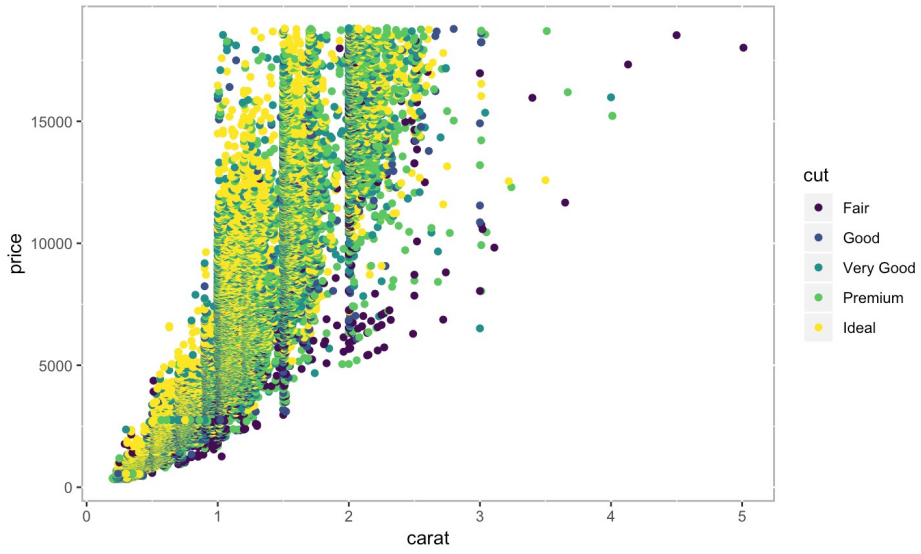


ggplot2

- `ggplot2` is an R package that provides an alternative framework based upon Wilkinson's (2005) Grammar of Graphics.
- `ggplot2` is, in general, more flexible for creating "prettier" and complex plots.
- Works by creating layers of different types of objects/geometries (i.e. bars, points, lines, polygons, etc.) `ggplot2` has at least three ways of creating plots:
  1. `qplot`
  2. `ggplot(...) + geom_XXX(...) + ...`
  3. `ggplot(...) + layer(...)`
- We will focus only on the second.

First Example

```
data(diamonds)
ggplot(diamonds, aes(x=carat, y=price, color=cut)) + geom_point()
```



Parts of a `ggplot2` Statement

- Data
  `ggplot(myDataFrame, aes(x=x, y=y)`
- Layers
  `geom_point(),geom_histogram()`
- Facets
  `facet_wrap(~ cut),facet_grid(~ cut)`
- Scales
  `scale_y_log10()`
- Other options
  `ggtitle('my title'),ylim(c(0, 10000)),xlab('x-axis label')`

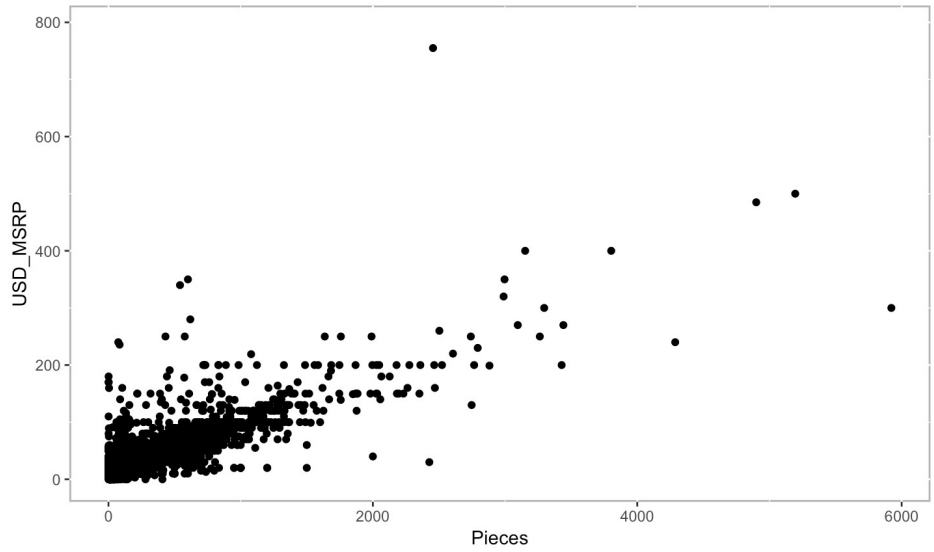
Lots of geoms

```
ls('package:ggplot2')[grep('geom_', ls('package:ggplot2'))]
```

```
##  [1] "geom_abline"          "geom_area"            "geom_bar"
##  [4] "geom_bin2d"           "geom_blank"           "geom_boxplot"
##  [7] "geom_col"             "geom_contour"         "geom_count"
## [10] "geom_crossbar"        "geom_curve"           "geom_density"
## [13] "geom_density_2d"      "geom_density2d"       "geom_dotplot"
## [16] "geom_errorbar"        "geom_errorbarh"       "geom_freqpoly"
## [19] "geom_hex"             "geom_histogram"       "geom_hline"
## [22] "geom_jitter"          "geom_label"           "geom_line"
## [25] "geom_linerange"       "geom_map"             "geom_path"
## [28] "geom_point"           "geom_pointrange"      "geom_polygon"
## [31] "geom_qq"              "geom_qq_line"         "geom_quantile"
## [34] "geom_raster"          "geom_rect"            "geom_ribbon"
## [37] "geom_rug"             "geom_segment"         "geom_sf"
## [40] "geom_smooth"          "geom_spoke"           "geom_step"
## [43] "geom_text"            "geom_tile"            "geom_violin"
## [46] "geom_vline"           "update_geom_defaults"
```

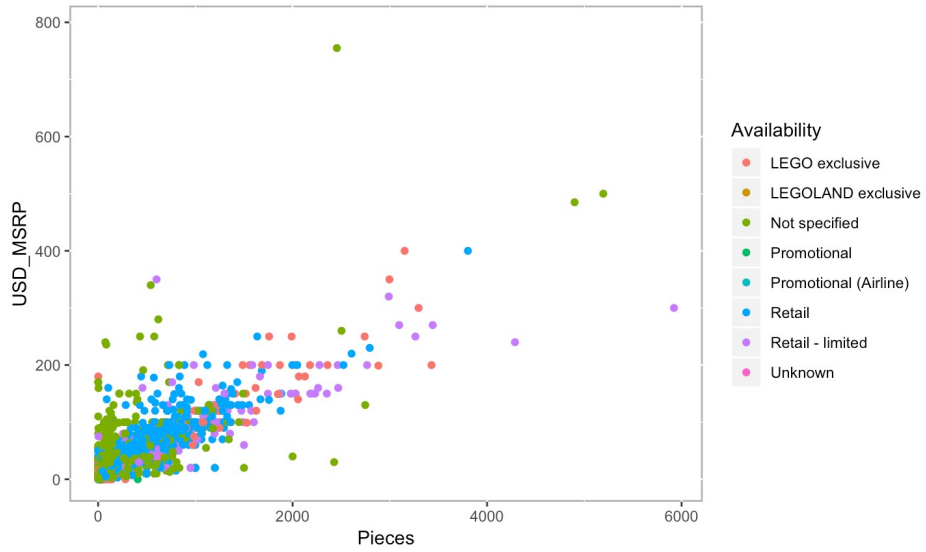
Scatterplot Revisited

```
ggplot(legosets, aes(x=Pieces, y=USD_MSRP)) + geom_point()
```



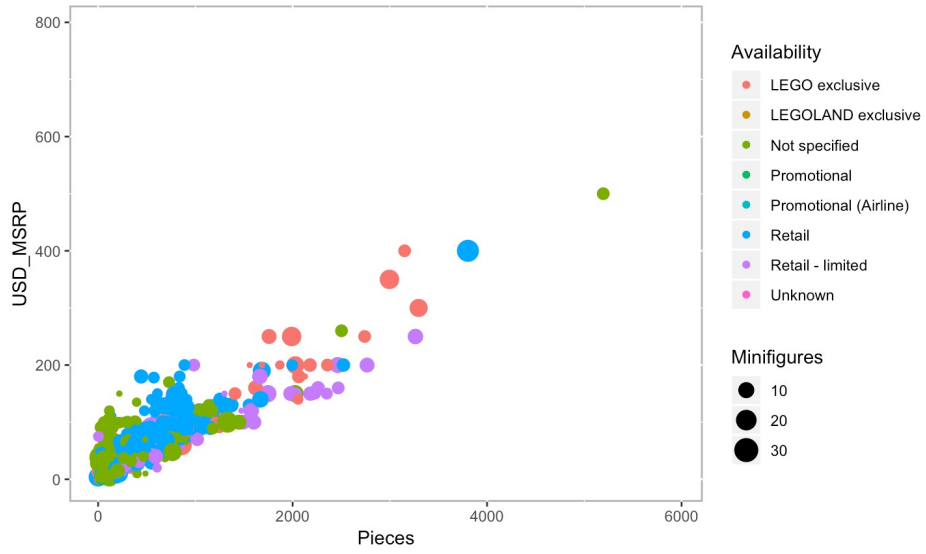
Scatterplot Revisited (cont.)

```
ggplot(legosets, aes(x=Pieces, y=USD_MSRP, color=Availability)) + geom_point()
```



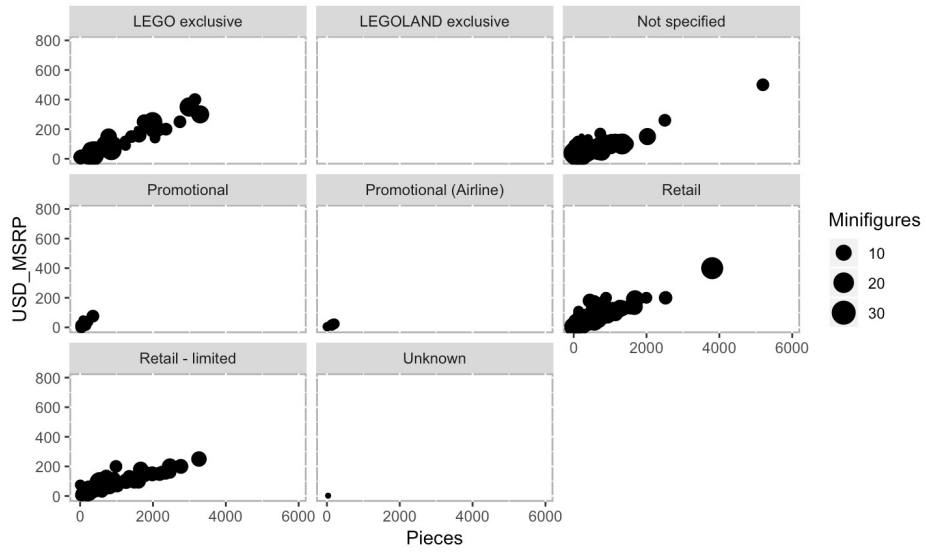
Scatterplot Revisited (cont.)

```
ggplot(legosets, aes(x=Pieces, y=USD_MSRP, size=Minifigures, color=Availability)) + geom_point()
```



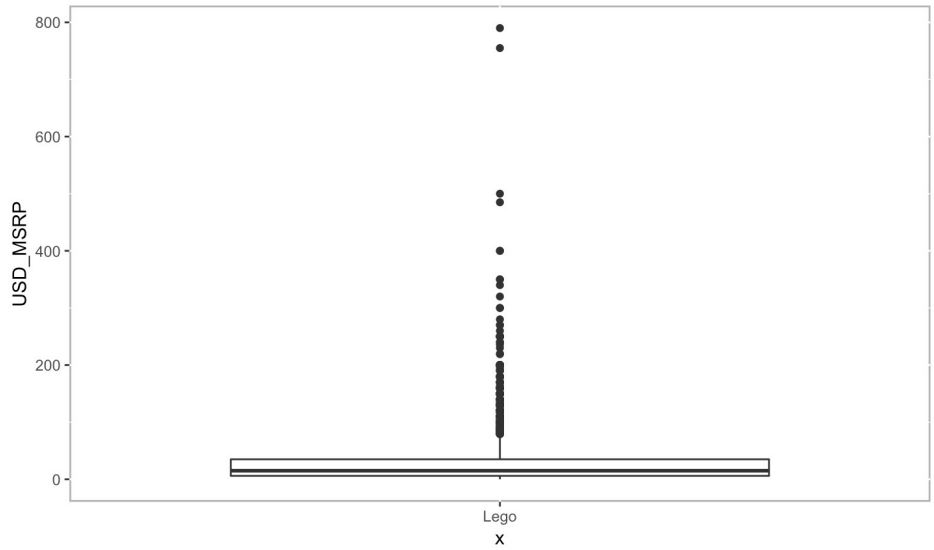
Scatterplot Revisited (cont.)

```
ggplot(legosets, aes(x=Pieces, y=USD_MSRP, size=Minifigures)) + geom_point() + facet_wrap(~ Availability)
```



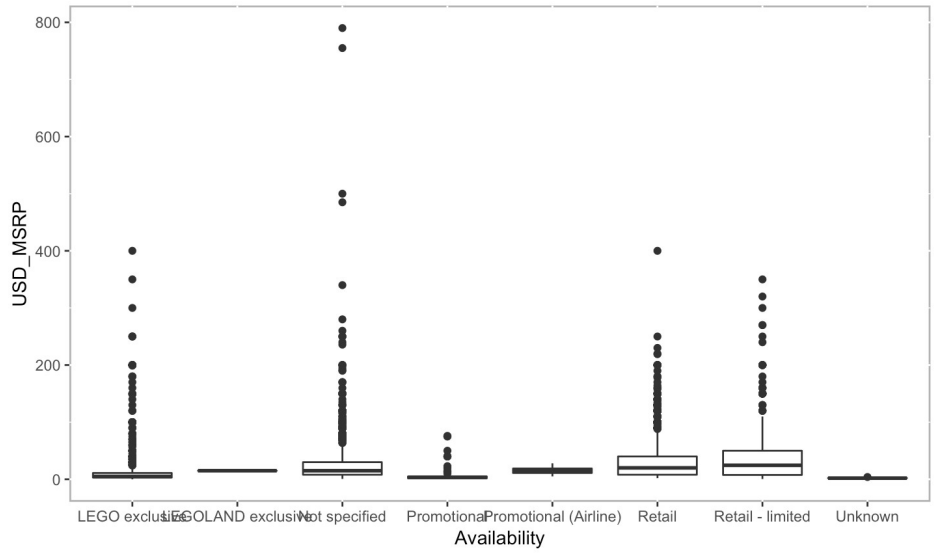
Boxplots Revisited

```
ggplot(legosets, aes(x='Lego', y=USD_MSRP)) + geom_boxplot()
```



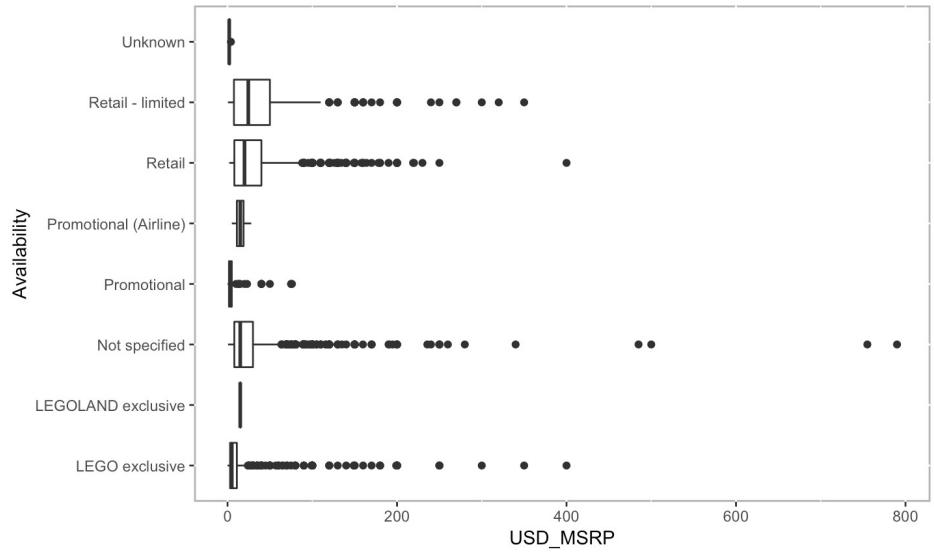
Boxplots Revisited (cont.)

```
ggplot(legosets, aes(x=Availability, y=USD_MSRP)) + geom_boxplot()
```



Boxplots Revisited (cont.)

```
ggplot(legosets, aes(x=Availability, y=USD_MSRP)) + geom_boxplot() + coord_flip()
```



Likert Scales

Likert scales are a type of questionaire where respondents are asked to rate items on scales usually ranging from four to seven levels (e.g. strongly disagree to strongly agree).

```
library(likert)
library(reshape)
data(pisaitems)
items24 <- pisaitems[,substr(names(pisaitems), 1,5) == 'ST24Q']
items24 <- rename(items24, c(
            ST24Q01="I read only if I have to.",
            ST24Q02="Reading is one of my favorite hobbies.",
            ST24Q03="I like talking about books with other people.",
            ST24Q04="I find it hard to finish books.",
            ST24Q05="I feel happy if I receive a book as a present.",
            ST24Q06="For me, reading is a waste of time.",
            ST24Q07="I enjoy going to a bookstore or a library.",
            ST24Q08="I read only to get information that I need.",
            ST24Q09="I cannot sit still and read for more than a few minutes.",
            ST24Q10="I like to express my opinions about books I have read.",
            ST24Q11="I like to exchange books with my friends."))
```

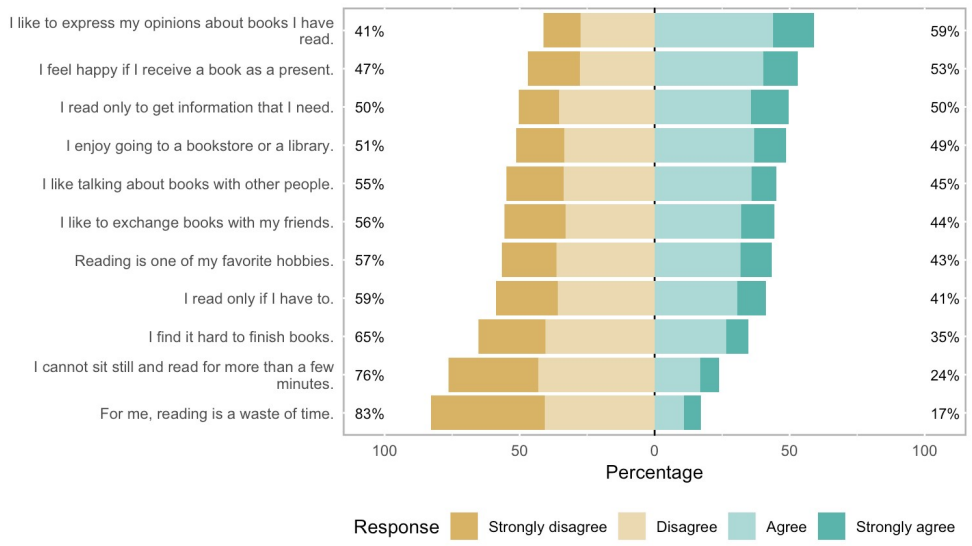
`likert` R Package

```
l24 <- likert(items24)
summary(l24)
```

```
##                                                          Item      low
## 10   I like to express my opinions about books I have read. 41.07516
## 5            I feel happy if I receive a book as a present. 46.93475
## 8               I read only to get information that I need. 50.39874
## 7                I enjoy going to a bookstore or a library. 51.21231
## 3             I like talking about books with other people. 54.99129
## 11               I like to exchange books with my friends. 55.54115
## 2                   Reading is one of my favorite hobbies. 56.64470
## 1                                I read only if I have to. 58.72868
## 4                          I find it hard to finish books. 65.35125
## 9  I cannot sit still and read for more than a few minutes. 76.24524
## 6                       For me, reading is a waste of time. 82.88729
##    neutral     high     mean        sd
## 10       0 58.92484 2.604913 0.9009968
## 5        0 53.06525 2.466751 0.9446590
## 8        0 49.60126 2.484616 0.9089688
## 7        0 48.78769 2.428508 0.9164136
## 3        0 45.00871 2.328049 0.9090326
## 11       0 44.45885 2.343193 0.9609234
## 2        0 43.35530 2.344530 0.9277495
## 1        0 41.27132 2.291811 0.9369023
## 4        0 34.64875 2.178299 0.8991628
## 9        0 23.75476 1.974736 0.8793028
## 6        0 17.11271 1.810093 0.8611554
```

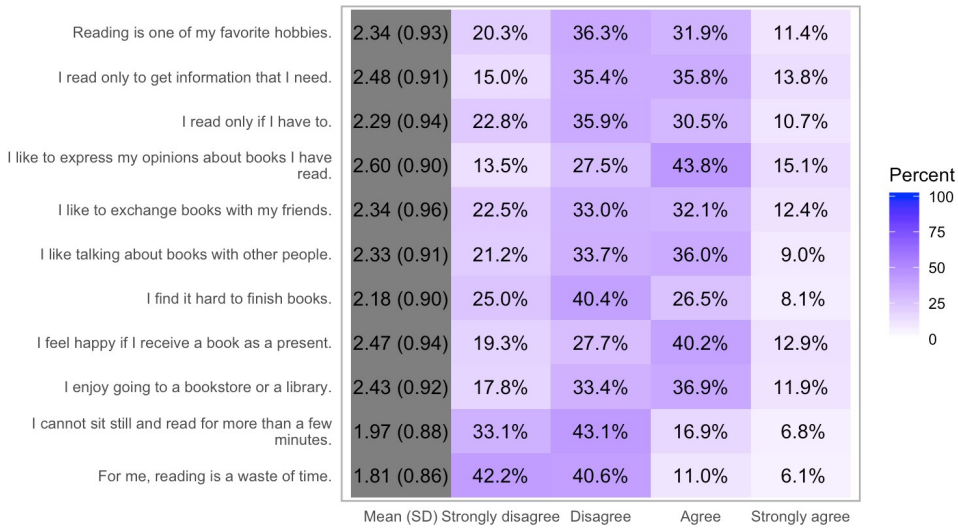
`likert` Plots

```
plot(l24)
```



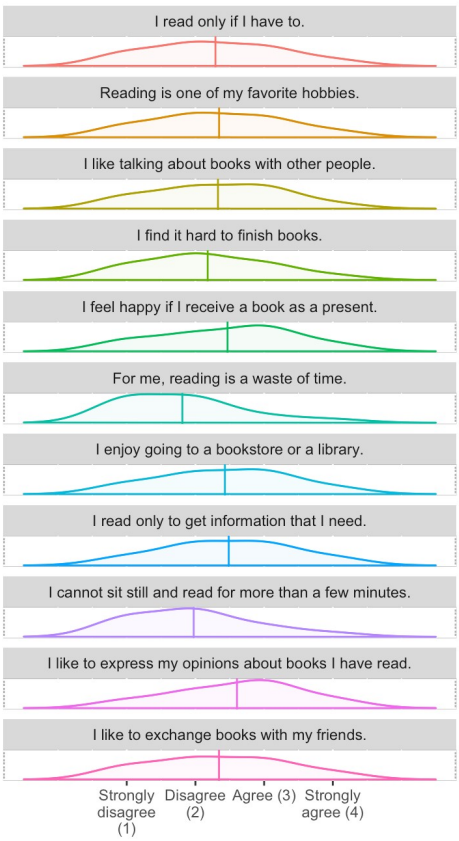
`likert` Plots

```
plot(l24, type='heat')
```



`likert` Plots

```
plot(l24, type='density')
```



Dual Scales

Some problems[1]:

- The designer has to make choices about scales and this can have a big impact on the viewer
- "Cross-over points" where one series cross another are results of the design choices, not intrinsic to the data, and viewers (particularly unsophisticated viewers)
- They make it easier to lazily associate correlation with causation, not taking into account autocorrelation and other time-series issues
- Because of the issues above, in malicious hands they make it possible to deliberately mislead

```
library(DATA606)
shiny_demo('DualScales', package='DATA606')
```

My advise:

- Avoid using them. You can usually do better with other plot types.
- When necessary (or compelled) to use them, rescale (using z-scores)

[1] http://blog.revolutionanalytics.com/2016/08/dual-axis-time-series.html [2] http://ellisp.github.io/blog/2016/08/18/dualaxes