WORKSHOP 1: INTRODUCTION TO R

R Basics:

MINJEONG PARK, GRADUATE ACADEMIC ASSISTANT
MOHAMMAD ZUBAIR, GRADUATE ACADEMIC ASSISTANT

UBC Library Research Commons
Background

• Have you used R before?

• Have you worked with other data analysis software?
Outline

• Overview of quantitative research
• Software for statistical models
• R environment
• Data preparation & management
• Descriptive statistics
• Statistical models
Overview of quantitative research

- **Research purpose/question** (e.g., association, group difference, prediction, change)
- **Research design** (e.g. Cross-sectional, Longitudinal)
- **Data collection**
- **Statistical analysis**
  - Manipulating data (e.g., recode data, select cases)
  - Applying statistical models (e.g. correlation, t-test, ANOVA, regression)

→ **Software programs help you to conduct statistical analysis**
Software for statistical analysis

- SPSS
- R
- Stata
- SAS
- Matlab
- Minitab
- Mplus
- HLM7
- ...

Software selection can be dependent on…
- *Convention in disciplines*
- *Availability*
- *Characteristics of software*
# Software for statistical analysis

<table>
<thead>
<tr>
<th></th>
<th>SPSS</th>
<th>Stata</th>
<th>R &amp; R studio</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Price</strong></td>
<td>Gradpack (Base version) 34.49 USD* (6 month) Free trial available</td>
<td>Stata IC 125 USD* (annual) 30 days money back option</td>
<td>Non-commercial purpose (Free)</td>
</tr>
<tr>
<td></td>
<td>Gradpack (standard version) 49.00 USD* (6 month) Free trial available</td>
<td>Stata SE 235 USD* (annual) 30 days money back option</td>
<td></td>
</tr>
<tr>
<td><strong>Availability in UBC</strong></td>
<td>Koerner Rm 217A, 218A Woodward Rm B25</td>
<td>Koerner Rm 217A, 218A Woodward Rm B25</td>
<td>Koerner Rm 217A, 218A Woodward Rm B25</td>
</tr>
<tr>
<td><strong>Interface</strong></td>
<td>clicking button + syntax</td>
<td>clicking button + syntax</td>
<td>syntax/command</td>
</tr>
<tr>
<td><strong>Data management</strong></td>
<td>Basic data management options</td>
<td>Basic data management options</td>
<td>Basic data management options + more data management options</td>
</tr>
<tr>
<td><strong>Statistical models</strong></td>
<td>Well-established statistical models</td>
<td>Well-established statistical models</td>
<td>Well-established statistical models + recent statistical models are updated</td>
</tr>
<tr>
<td><strong>Data visualization</strong></td>
<td>Basic plots, graphs, charts</td>
<td>Basic plots, graphs, charts</td>
<td>Basic plots, graphs, charts + more visualization options</td>
</tr>
<tr>
<td><strong>Discussion forum</strong></td>
<td>Stack Overflow Cross Validated Nabble</td>
<td>Statalist</td>
<td>Stack Overflow Cross Validated Nabble</td>
</tr>
</tbody>
</table>
• **R is a package-based program**
• **R package** is a library of prewritten code designed for particular task or a collection of tasks
R environment
R environment

• **Standard R interface in Windows & Mac**

• **R studio is an integrated development environment (IDE)**
  Code editor, Debugging & Visualization tools
  → R studio makes R easier to use
R environment

R studio is composed of 4 main panels:

- Editor
  Built-in text editor (e.g. R script)

- Console
  R codes are entered and the output is printed

- Environment & History
  List of datasets loaded
  History of commands used

- Files, Plots, Packages, Help
R environment

R studio is composed of 4 main panels:

- File tab: file explorer
- Plots tab: plots produced
- Package tab: list of available and installed packages
- Help tab: help commands and help search
Two core steps to start R

• Set up *working directory* where data are located
Two core steps to start R

• Installing and loading *packages*

1) Under Tools -> Packages tab

2) With code using `install.packages()` and `require()` functions

```r
install.packages( c("dplyr", "psych", "ggplot2") )
require(dplyr)
require(psych)
require(ggplot2)
```
Some useful tips

• **R is case sensitive**
  
  View(data) vs. view(data)

• **Shortcut for running R code in R script**

  
  - Ctrl
  - Alt

• **Shortcut for running previous R code**

  
  Arrow key to go back to previous code(s)
Data preparation and management in R
Example data

• Cross-sectional design:
  A set of variables measured from each person in one time point

• A set of variables:
  - Gender (Male = 0, Female = 1)
  - Age (range 10 – 80)
  - Marital status (Married, common law = 1, Widow, divorce, separate = 2, Single, never married = 3)
  - Employment (no job = 1, part time = 2, full time = 3)
  - Quality of life_total (range 0 – 20)
  - Distress_total (range 0 – 20)
  - Self-esteem items (range 0 – 3)

• Missing data are coded as 999
Data import

• Quick function to import data in various formats:

  In *Environment* tab,

  *Import Dataset* > choose “format” accordingly
Data import

- Quick function to import data in various formats:

Select Excel file

Specify data name

Import!
Data import

• Define missing data
  
  Define 999 as missing data

  mydata[mydata=="999"] <- NA

• Some useful functions to check data

  names() function: check variable names

  names(mydata)

  table() function: check variable values and frequency

  table(mydata$Gender)

  select variable of interest

  table() + is.na() functions: check # of missing data

  table(is.na(mydata$Age))  Is the value is missing? TRUE or FALSE
Data import

• Some useful functions to define data type

`is.factor()` function: check if the variable is defined as categorical

```r
is.factor(mydata$Gender) # [1] FALSE
```

`as.factor()` function: define as categorical variable

```r
mydata$Gender <- as.factor(mydata$Gender)
```

`is.numeric()` function: check if the variable is defined as continuous

```r
is.numeric(mydata$Age) # [1] TRUE
```

`as.numeric()` function: define as continuous variable

```r
mydata$Age <- as.numeric(mydata$Age)
```
Data management with “dplyr” package

• “dplyr” is a package providing common data management functions

• Popular functions:
  - mutate(): adds new variables
  - recode(): recodes values in the variable
  - filter(): selects cases based on their values
Recode to a new variables

- Recoding “Employment” variable with 3 levels into “Employment_new” variable with 2 levels

<table>
<thead>
<tr>
<th>Employment</th>
<th>Employment_new</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>job</td>
</tr>
<tr>
<td>1</td>
<td>no job</td>
</tr>
<tr>
<td>1</td>
<td>no job</td>
</tr>
<tr>
<td>2</td>
<td>job</td>
</tr>
<tr>
<td>1</td>
<td>no job</td>
</tr>
<tr>
<td>3</td>
<td>job</td>
</tr>
</tbody>
</table>

1 = no job
2 = part time
3 = full time

1 = no job
2 = job
Recode to a new variables

• Recoding “Employment” variable with 3 levels into “Employment_new” variable with 2 levels

• `mutate()` + `recode()` functions

```r
recode(mydata$Employment, "1" = "no job", "2" = "job", "3" = "job")
```

Value “1” in existing variable is recoded into “no job”

```r
mutate(mydata, Employment_new = recoding “Employment” var)
```

Recoding existing variable and assign to the new variable
Recode to a new variables

• Recoding “Employment” variable with 3 levels into “Employment_new” variable with 2 levels

• `mutate()` + `recode()`

```r
mydata <- mutate(mydata, Employment_new = recode(mydata$Employment, "1" = "no job", "2" = "job", "3" = "job"))
```

```r
> table(mydata$Employment_new)

  no job  job
     58  142
```
Compute a variables

- Compute “Esteem_total” score of 10 esteem questions
- mutate () + rowSums () functions
Compute a variables

- Compute “Esteem_total” score of 10 esteem questions
- `mutate()` + `rowSums()` functions

```
```

`dataset
Name of new var`

`mutate(mydata, Esteem_total = total score of “Esteen_Q1 ~ Q10”)`

`Computing total score`
Compute a variables

• Compute “Esteem_total” score of 10 esteem questions
• mutate () + rowSums () functions

\[
\text{mydata} \leftarrow \text{mutate(} \text{mydata, Esteem_total = rowSums(} \text{mydata[, c("Esteem_Q1", "Esteem_Q2", "Esteem_Q3", "Esteem_Q4", "Esteem_Q5", "Esteem_Q6", "Esteem_Q7", "Esteem_Q8", "Esteem_Q9", "Esteem_Q10")])}\right)
\]
Select cases based on conditions & subset

- Select all cases whose gender is male and age greater than 40
- filter () function
Select cases based on conditions & subset

• Select all cases whose gender is male and age greater than 40

• filter () function

```r
filter(mydata, Gender == "1" & Age > 40)
```

Select cases based on conditions & subset

• Useful filter functions

- equal to  ==
- greater  >  less  <
- equal and greater than  >=  equal and less than <=
Select cases based on conditions & subset

• Select all cases whose gender is male and age greater than 40
• `filter()` function

```r
subdata <- filter(mydata, Gender == "1" & Age > 40)
```

```
> table(subdata$Gender)

     1  63
```

```
> table(subdata$Age)

     42  43  44  45  46  47  48  53  56  57  59  63  65  67  68  72  73  74  75  78  79
      2   1   4   7   4   2   2   1   4   2   1   2   1   4   4   1   1   1   2   3   4
```
Descriptive statistics
Descriptive statistics

• Descriptive statistics provide a summary of your data

• Purpose of looking at descriptive statistics:
  
  (1) Check whether valid data are loaded properly  
  E.g., unexpected values (e.g., 999, -2) in “Age” variable (range 10-80)

  (2) Explore data  
  E.g., potential group differences, associations between variables

  (3) Sample description  
  E.g., % of gender, mean and standard deviation of quality of life score
Descriptive statistics

• Descriptive stat with basic R function:

**summary() function**

```
summary(mydata)
```

**dataset**

<table>
<thead>
<tr>
<th>ID</th>
<th>Gender</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>1.00</td>
<td>Min.</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>50.75</td>
<td>1st Qu.</td>
</tr>
<tr>
<td>Median</td>
<td>100.50</td>
<td>Median</td>
</tr>
<tr>
<td>Mean</td>
<td>100.50</td>
<td>Mean</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>150.25</td>
<td>3rd Qu.</td>
</tr>
<tr>
<td>Max.</td>
<td>200.00</td>
<td>Max.</td>
</tr>
<tr>
<td>NA's</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Descriptive statistics

- Descriptive stat with “psych” package:

describe() function:

```r
> describe(mydata)
```

```
vars n mean sd median trimmed mad min max range skew kurtosis se
ID 1 200 100.50 57.88 100.5 100.50 74.13 1 200 199 0.00 -1.22 4.09
Gender 2 200 0.55 0.50 1.0 0.56 0.00 0 1 1 -0.18 -1.98 0.04
Age 3 196 44.20 17.42 43.0 43.32 16.31 13 80 67 0.43 -0.81 1.24
Marital 4 200 2.04 0.82 2.0 2.06 1.48 1 3 2 -0.08 -1.52 0.06
```

describeBy() function: descriptive stat by group

```r
> describeBy(mydata, group = mydata$Gender)
```

```
Descriptive statistics by group
group: 0
```

```
vars n mean sd median trimmed mad min max range skew kurtosis se
ID 1 91 83.46 48.13 65 80.19 54.86 16 196 180 0.48 -0.83 5.05
Gender 2 91 0.00 0.00 0 0.00 0.00 0 0 0 NaN NaN 0.00
Age 3 89 43.78 17.33 38 42.95 14.83 19 80 61 0.45 -0.93 1.84

```

```
group: 1
```

```
vars n mean sd median trimmed mad min max range skew kurtosis se
ID 1 109 114.72 61.59 125 118.09 71.16 1 200 199 -0.45 -1.05 5.90
Gender 2 109 1.00 0.00 1 1.00 0.00 1 1 0 NaN NaN 0.00
```
Descriptive statistics with visualization (basic function)

- Create histogram for “Age”
- hist() function

\[
\text{hist}(\text{mydata$Age})
\]

\textit{dataset} \hspace{1cm} \textit{select variable of interest}

OR

\[
\text{hist}(\text{mydata$Age}, \text{col} = \text{"grey"}, \text{main} = \text{"Age distribution"}, \text{xlab} = \text{"Age"}, \text{ylab} = \text{"# of people"})
\]

\textit{name of x axis} \hspace{1cm} \textit{name of y axis}
Descriptive statistics with visualization (basic function)

- Create bar plot for “Gender”
- `barplot()` + `table()` functions

```r
barplot(table(mydata$Gender))
```

OR

```r
barplot(table(mydata$Gender), col="white", main="Gender bar plot",
       xlab="Gender", ylab="# of people", names.arg=c("Male, "Female"))
```

- Frequency of variable
- Color of plot
- Title of plot
- Name of x & y axes
- Name of each bar
Descriptive statistics with visualization (basic function)

- Create scatter plot for “Distress_total” and “QOL_total”
- plot () function

```r
plot(mydata$Distress_total, mydata$QOL_total)
```

OR

```r
plot(mydata$Distress_Total, mydata$QOL, col="mediumpurple", main="scatter plot", xlab="Distress", ylab="Quality of life")
```
Descriptive statistics with visualization ("ggplot2")

• **ggplot2** is a data visualization package providing various customization functions

• **Basic functions:**
  
  ggplot(): create a new plot
  
  aes(): specify how variables in the data are mapped to plot
  
  geom(): define graph type (e.g., histogram, bar plot)
    
    - geom_histogram() for histogram
    
    - geom_bar() for bar plot
    
    - geom_point() for scatter plot
Descriptive statistics with visualization ("ggplot2")

- Create histogram for "Age"
- `ggplot(dataset, aes()) + geom_histogram()`

```
ggplot(mydata, aes(x=Age)) +
  geom_histogram(breaks=seq(10, 80, by=10),
                 fill = "slategray2", col="white") +
  ggtitle("Age distribution") +
  xlab("Age in year") + ylab("# of people")
```

Variable mapped to X axis
Customizing x-axis ticks
Fill of plot  Border color  title of plot
name of x axis  name of y axis
Descriptive statistics with visualization ("ggplot2")

• Create histogram for “Age” by “Gender”
• `ggplot(dataset, aes()) + geom_histogram()+ facet_grid()`
Descriptive statistics with visualization ("ggplot2")

- `ggplot(dataset, aes()) + geom_histogram()`

```r
ggplot(mydata, aes(x=Age)) + geom_histogram(breaks=seq(10, 80, by=10), fill = "slategray2", col="white") + ggtitle("Age distribution") + xlab("Age in year") + ylab("# of people") + facet_grid(~Gender, labeller = labeller(Gender = c("0" = "Male", "1" = "Female")))
```

**Forms matrix of panels**

**change labels of gender**

`ggplot2` – customizing color:

Descriptive statistics with visualization (ggplot2)

- Create scatter plot for “Distress_total” and “QOL”
- `ggplot(aes()) + geom_point()` functions

Distress and QOL plot

```r
ggplot(mydata, aes(x=Distress_total, y=QOL_total)) + geom_point()
```

*Defining dataset and x and y-axis*  
*Type of graph*

- Create scatter plot for “Distress_total” and “QOL” by “Gender”

```r
ggplot(mydata, aes(x=Distress_total, y=QOL_total,
  colour=factor(Gender), shape=factor(Gender)))
```

*Defining color and shape of the dots*

+ `geom_point(size=4)`

*Defining size of dots*
Statistical models in R
Statistical models in R

- Statistical models we are covering today…

<table>
<thead>
<tr>
<th>Data type</th>
<th>Continuous</th>
<th>Categorical</th>
<th>Ordinal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inference</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Descriptive statistics</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inferential statistics</td>
<td>Independent sample T-test</td>
<td>Independent one-way ANOVA</td>
<td>Pearson correlation</td>
</tr>
</tbody>
</table>
Independent sample T-test

- **Independent T-test** compares *means* between *two groups*

- It is often used to see whether there is a *group difference* in *continuous data* between *two groups* (e.g., gender, treatment vs. control)

- Example

<table>
<thead>
<tr>
<th>Gender</th>
<th>Quality of life</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>9</td>
</tr>
<tr>
<td>Male</td>
<td>15</td>
</tr>
<tr>
<td>Male</td>
<td>18</td>
</tr>
<tr>
<td>Female</td>
<td>19</td>
</tr>
<tr>
<td>Female</td>
<td>16</td>
</tr>
<tr>
<td>Female</td>
<td>9</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

- **Model assumptions**
  1. Independence
  2. Normality
  3. Equal variance
Independent sample T-test

R code for independent T-test (equal variance assumption met):

```r
# Independent categorical (group) variable
# Dependent continuous variable

t.test(QOL_total ~ Gender, var.eq=TRUE)
```

Test statistics:
- t = -1.7377
- df = 198
- p-value = 0.08382

95% CI of group difference:
- Lower limit: -2.3027378
- Upper limit: 0.1454638

More information on `t.test` command:
`?t.test`
Independent sample one-way ANOVA

- Independent sample one-way ANOVA compares means between more than two groups

- It is often used to see whether there are group differences in continuous data between more than two groups

- Example

  - Data type

  - Research interest

- Model assumption:
  (1) Independence, (2) Normality, (3) Equal variance
Independent one-way ANOVA

R code for independent one-way ANOVA:

\[
\text{Code for ANOVA} \quad \text{Independent categorical variable}
\]
\[
\text{ANOVA} = \quad \text{aov} \quad (\text{QOL\_total} \sim \text{factor(Employment)})
\]

\text{Name of the model} \quad \text{(will be saved in R environment)} \quad \text{Dependent cont. variable}

Code of result:
\[
\text{summary(ANOVA)}
\]

Conclusion:
There is a significant difference in quality of life score among employment groups [F (2, 197) = 3.14, p = 0.045)]
Post-Hoc test

Tukey multiple pairwise-comparisons:

TukeyHSD (ANOVA)

```r
> TukeyHSD(ANOVA)
  Tukey multiple comparisons of means
  95% family-wise confidence level
  Fit: aov(formula = QOL_total ~ Employment)

$`Employment`

   diff     lwr     upr  p adj
part-time-no-job  1.8896552 0.09460024 3.684710 0.0364787
full-time-no-job  1.2717447 -0.56945185 3.112941 0.2349696
full-time-part-time -0.6179104 -2.34363982 1.107819 0.6751493
```

Plot(TukeyHSD(ANOVA))

**Conclusion:**
The level of quality of life for part-time group was *significantly higher* than no-job group (p = 0.03).
Pearson’s correlation

• Pearson’s correlation is used to examine **associations between variables** (represented by **continuous data**) by looking at the direction and strength of the associations

  *Example*

  ![Diagram showing associations between Distress, Quality of life, and Self-esteem](image)

<table>
<thead>
<tr>
<th>Distress</th>
<th>Quality of life</th>
<th>Self-esteem</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td>8</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>6</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>12</td>
<td>10</td>
<td>8</td>
</tr>
<tr>
<td>15</td>
<td>12</td>
<td>9</td>
</tr>
</tbody>
</table>

  *Checking outlier (scatter plot)*

  `plot(QOL_total, Distress_total)`
Pearson’s correlation

R code for Pearson correlation:

```r
cor(QOL_total, Distress_total, use = "pairwise.complete.obs", method= "pearson")
```

or

```r
cor.test(QOL_total,Distress_total, method="pearson")
```

Pearson's correlation

- **p-value**
- **95% CI of Pearson rho (r or ρ)**
- **Continuous variables**
- **Correlation method**

Continuous variables

- **Pearson rho (r or ρ)**
Ordinary least squares linear regression

- Ordinary least squares (OLS) or Linear regression is used to **explain/predict** the phenomenon of interest **(continuous data)**

**Research interest**

**Data type**

- Example

**Simple OLS regression**

- Dependent V (Quality of life)
- IV 1 (Distress level)

**Multiple OLS regression**

- Dependent V (Quality of life)
- IV 1 (Distress level)
- IV 2 (Self-esteem)
- IV 3 (Gender)

- **Model assumptions**
  1. Independence
  2. Normality
  3. Equal variance
  4. Linearity
Building multiple linear regression model

**Code for linear model**

```r
model1 = lm(QOL_total ~ Distress_total + Esteem_total + factor(Gender))
```

**Independent variables**

**Dependent cont. variable**

- Summary of the model

```
summary(model1)
```

- Conclusion:

Approximately, 55.8% of the variability in the quality of life can be explained by this multiple linear regression model.

The overall regression model significantly explained the quality of life.
Conclusion:
Distress and self-esteem significantly predicted the level of quality of life

Explaining coefficient for distress: On average, we would expect a decrease of -0.416 in quality of life score for an increase of one in distress score, when esteem score and gender remain constant
Exporting data from R

To a CSV File

```r
write.table(mydata, file="mydata.csv", sep="",")
```

To a tab delimited text file

```r
write.table(mydata, file="mydata.txt", sep="/t")
```

To an Excel Spreadsheet

```r
library(xlsx)
write.xlsx(mydata, file="mydata.xlsx")
```
RESEARCH COMMONS: AN INTERDISCIPLINARY RESEARCH-DRIVEN LEARNING ENVIRONMENT

- Literature review
- Systematic Review Search Methodology
- Citation Management
- Thesis Formatting
- Nvivo Software Support
- SPSS Software Support
- R Group
- Multi-Disciplinary Graduate Student Writing Group
R SERVICES BY RESEARCH COMMONS

- Workshops

- One-on-one Consultation

Request form to book R consultation:

THANK YOU!
QUESTION, COMMENT, IDEAS
FEEDBACK