Data Management for Research and Institutional Decision Making

Duplicates in R

- Duplicates are repeated information in rows of data sets.
- Managing duplicates involves identifying and dropping them.
- Functions to identify duplicates

duplicated()

- Functions to create unique data sets

distinct()

unique()

Combining data sets

- Combining data sets involves the following:

1. Adding rows from different files into one file by appending data sets.

2. Adding columns by adding columns from one file into one file.

Appending data sets

- Combines data sets by adding additional rows to the data set
- Data sets should have the same number of columns
- Data sets should have the same column names

YEAR: 2017 Table 1

Year	Month	Product A	Product B
2017	January	45112	45564
2017	February	50000	4556214
2017	March	75100	45871

YEAR: 2018 Table 2

Year	Month	Product A	Product B
2018	January	45564	4500
2018	February	4556214	15245
2018	March	45871	45872

YEAR: 2019 Table 3

Year	Month	Product A	Product B
2019	January	4500	45112
2019	February	15245	50000
2019	March	45872	75100

RESULTING TABLE

Year	Month	Product A	Product B
2017	January	45112	45564
2017	February	50000	4556214
2017	March	75100	45871
2018	January	45564	4500
2018	February	4556214	15245
2018	March	45871	45872
2019	January	4500	45112
2019	February	15245	50000
2019	March	45872	75100

Append

❖Demonstration

- bind_rows() function to combine agriculture data from town A and town B into one data set.

```
agric_AB <- bind_rows(agriculture_townA,agriculture_townB)
```

Merging

- Merging adds columns from one data set to another
- All rows in both data frames are included in the result

Merging

❖Demonstration

- inner_join() function to merge agriculture data and agriculture2 data into one data set.

```
agric_gender <- inner_join(agriculture, agriculture2, by = "Farm_ID")
```

Data Exploration

- Exploration involves taking a dive into data sets to understand meanings and possible relationships.

- Summarizing data:
  - Involves creating frequency distribution tables for categorical variables
  - Obtaining descriptive statistics for numerical variables

- Relationships help us to get insights into our data sets
  - Involves using appropriate statistical tests
  - Involves correct interpretation of statistical tests

Relationships between categorical variables

- This involves checking whether two categorical variables are related to each other.
- Using the chisquare test
- Significant p values (pvalue<=0.05) imply significant relationships.

Function to perform chisquare is chisq.test()

Steps :

1. Summarise data in a contingency table
2. Perform the chisquare test

Chi square test of independence

- Using the agriculture data set. Check whether irrigation type is associated with location

  Null: No association

  Alternative: There is association

```
table(agriculture$location, agriculture$Irrigation_Type)

      Drip Flood Manual Rain-fed Sprinkler
rural   41    34     19       10        11
urban   19    19     12       14        28
```

Cross tabulation

```
> chisq.test(table(agriculture$location, agriculture$Irrigation_Type))

        Pearson's Chi-squared test

data:  table(agriculture$location, agriculture$Irrigation_Type)
X-squared = 19.657, df = 4, p-value = 0.0005837
```

Chi square test

P-value

Significant(p-value<0.005)

Categorical and Numerical variables

- Exploring relationships between the two involves understanding in variation of the quantitative variable across the levels of the categorical variable.

- Example:

  - average yield across location
  - average fertilizer used in different crops

Categorical variables vs Numerical variables

Steps:

Check the distribution of the quantitative variable.

Obtain a frequency table for the categorical variable.

If the categorical variable has 2 categories, perform the t test.

If the categorical variable has more than 2 categories, perform the ANOVA test.

The t test of equality of means

- Using the agriculture data set , check whether average yield varies by location using t.test()

  Null: No difference between the means

  Alternative: The two means are not equal

```
> t.test(yield_tonnes ~ location, data = agriculture)

        Welch Two Sample t-test

data:  yield_tonnes by location
t = -0.53279, df = 189.81, p-value = 0.5948
alternative hypothesis: true difference in means between group rural and group urban is not equal to 0
95 percent confidence interval:
 -0.4722748  0.2714052
sample estimates:
mean in group rural mean in group urban
           7.823913            7.924348
```

P-value

Mean yield in rural

Mean yield in urban

Wilcoxon rank-sum test

Wilcoxon rank sum test is the non-parametric test of equality of two medians.

- Does not require normally distributed populations

Steps:

Check the distribution of the numeric variable

If it is not normally distributed, test for equality of medians using Wilcoxon rank-sum test

Wilcoxon ranksum test

- Check whether Farm area vary by location

  Ho: No difference in median Farm area

  Ha: The median Farm area differs by location

```
> shapiro.test(agriculture$Farm_Area_acres)

        Shapiro-Wilk normality test

data:  agriculture$Farm_Area_acres
W = 0.9358, p-value = 6.575e-08
```

P-value
Farm area not
normally distributed

```
> wilcox.test(Farm_Area_acres ~ location, data = agriculture)

        Wilcoxon rank sum test with continuity correction

data:  Farm_Area_acres by location
W = 5360, p-value = 0.871
alternative hypothesis: true location shift is not equal to 0
```

P-value > 0.05

END