ST2137 Cheatsheet
by Yiyang, AY22/23

4. Numerical Data Analysis

Variables can be **Quantitative** (**Discrete** or **Continuous**) or **Categorical** (**Ordinal** or **Nominal**).

Single Quantitative Variable

For a unimodal distribution, **Skewness** value represents the amount and direction of skew:

$$\frac{\sqrt{n(n-1)}}{n-2} \times \frac{m_3}{(m_2)^{3/2}}$$

where $m_2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$ and $m_3 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3$.
A distribution is **Skewed Right** / **Positively Skewed** if peak is towards the left and the right tail is longer (e.g. income). A **Symmetric** distribution has close to 0 skewness.

Kurtosis measures shape of a distribution, with higher (lower) values indicate a higher & shaper (lower & less distinct) peak:

$$\frac{n-1}{(n-2)(n-3)}\left[\frac{(n+1)m_4}{m_2^2} - 3(n-1)\right]$$

Graphical summaries for 1 quantitative: [1] Histogram & Density Plot, [2] Boxplot, [3] QQ Plots, plots of standardised sample quantiles against theoretical quantiles of a standard normal.

Association between Two Variables

Summaries for 2 quantitiative: [1] Correlation Val., [2] Scatterplot.
Summaries for quantitative & categorical: [1] Boxplots by Groups, [2] Histogram by Groups.

5. Robust Estimators

A statistical method is **Robust** wrt. a particular assumption if it performs adequately even when that assumption is modestly violated.

Robust Estimation of Location

Location Estimators: [1] Arithmetic mean, [2] Trimmed mean, [3] Winsorized mean, [4] M-Estimates.

The $100\alpha\%$ **Trimmed Mean** is calculated by: [1] Discard lowest $100\alpha\%$ and highest $100\ \alpha\%$. [2] Arithmetic mean of remaining.
Note: [1] 2α of extreme data discarded. [2] Usually $\alpha \in [0.1, 0.2]$.

The $100\alpha\%$ **Winsorized Mean** is calculated by: [1] Sort observations as $x_{(1)}, x_{(2)}, \ldots, x_{(n)}$. [2] Replace $[n\alpha]$ smallest observations with $x_{([n\alpha]+1)}$, and $[n\alpha]$ largest with $x_{(n-[n\alpha])}$. Here, $[a]$ denotes as the nearest integer of a. [3] Arithmetic mean of replaced.

M-Estimator with a non-constant error function ρ, T, is defined as

$$T = \arg\min_T \sum_{i=1}^{n} \rho(x_i - T)$$

Robust Estimation of Scale

Scale Estimators:

- **Inter-Quartile Range** $\text{IQR} = Q_3 - Q_1$
- **Median Abs Deviation** $\text{MAD} = \text{med}_i(|x_i - \text{med}_j(x_j)|)$
- **Gini's Mean Difference** $G = \sum_{i<j}|x_i - x_j|/C_2^n$

Note: For a normal distribution, $\text{IQR} = 1.35\sigma, \text{MAD} = 1.4826\sigma, \sqrt{\pi}G/2 = \sigma$.

6. Categorical Data Analysis

Summaries for 1 categorical: [1] Frequency Table (with category of highest frequency as **Modal Category**), [2] Bar plot.

Two Categorical

Association between 2 categorical: [1] Contingency Table, [2]

Contingency Table

Note: Row for explanatory variables x and column for response variables Y (success or fail). Measures of association: [1] Sample Diff. $= p_1 - p_2$, [2] Relative risk $= p_1/p_2$, [3] Odds Ratio.
For a success prob. π, **Odds of Success** is odds $= \pi/(1-\pi)$. For 2-way contingency table, the **Odds Ratio** (OR), θ, and **Sample OR**, $\hat{\theta}$, are defined

$$\theta = \frac{\pi_1/(1-\pi_1)}{\pi_2/(1-\pi_2)},\ \hat{\theta} = \frac{p_1/(1-p_1)}{p_2/(1-p_2)} = \frac{n_{11}\times n_{22}}{n_{12}\times n_{21}}$$

where $n_{11}, n_{12}, n_{21}, n_{22}$ are 4 cell counts.
The $100\%(1-\alpha)$ Confidence Interval for OR:

$$\exp\{\log\hat{\theta} \pm z_{\alpha/2} \times ASE(\log\hat{\theta})\}$$

where

$$ASE(\log\hat{\theta}) = \sqrt{1/n_{11} + 1/n_{12} + 1/n_{21} + 1/n_{22}}$$

Note: If x and Y independent, $\theta = 1$.

Prospective & Retrospective Studies

Prospective Studies sample subjects randomly from a population and randomly assign exposure variables or record exposure status. All 3 measures above are valid.
Retrospective Studies sample a group of cases and a group of controls (i.e. based on Y), and check each subject's exposure. As such, **cannot** obtain valid estimates of π_1, π_2, as we obtain $Pr(x|Y)$ but need to estiamte $Pr(Y|x)$.

Dependence Tests

Chi-squared Test

Assumption: All $e_{ij} \geq 5$. Note: **Fisher Exact Test** for small samples.
Null Hypo.: Two variables are independent.
Test Statistic

$$\chi^2 = \sum \frac{(|o_{ij} - e_{ij}| - 0.5)^2}{e_{ij}} \sim \chi_1^2$$

, where o_{ij}, e_{ij} are observed and expected count for each cell, and expected count is RowTotal × ColTotal/Total.

McNemar's Test

Settings: x and Y represent num. of students passing & failing a test before & after a lesson. **Dependent samples**.
Null Hypo.: Before and after are independent.
Test Statistic: let b and c denotes pass-then-fail & fail-then-pass:

$$\chi^2 = \frac{(b-c)^2}{b+c},\ \text{or (if small sample,)}\ \frac{(|b-c|-1)^2}{b+c} \sim \chi_1^2$$

Chi-Square Test for General Tables

Assumption: Large samples, or $\leq 25\%$ cells with expected < 5.
Settings: Contingency table with r rows & c cols now.
Same hypotheses and test statistic as previous case but follows χ^2 with d.f. $(c-1)\times(r-1)$ now.
Standardised / Adjusted Residual for each cell:

$$r_{ij} = \frac{o_{ij} - e_{ij}}{SE(o_{ij} - e_{ij})},\ SE = \sqrt{e_{ij}(1-p_{i+})(1-p_{+j})}$$

where p_{i+} and p_{+j} marginal prob. of row i and of col j. Note: $|r_{ij}| > 2$ indicates lack of fit of H_0 in the cell.

R Coding

Basic Syntax

```
# Vector
numeric(n); character(n)  # vector with n 0's / ""'s
rep(a, b)  # replicate item a by b times
seq(from=a,to=b,by=c); seq(from=a,to=b,length=d);
# Matrix
matrix(v, nrows=a, ncols=b, byrow=T)  # matrix from vec
rbind(...); cbind(...)  # Bind rows / cols to form matrix
# Dataframes
df <- data.frame(m)  # df from matrix
names(df) = c(...)  # set df colname
row.names(df) = ...  # set df rownames
df[a, b:c]  # get row a, col b toc
df$abc  # get col with name abc
df[order(val),], df[rev(order(val)),]  # asc, desc
merge(df1, df2, by="id")


# Conditioning
if (condition) {
  ... # then statements^^I
} else {
  ... # else statements
}
# While loop
while (condition) {
  ... # expression
```

```
}
# For loop
for (<variable> in <range>) {
  ... # expression
}

# IO
read.csv(..., header=T, width=c(...)), read.table(...)
# Note: Use width if each variable spans multiple lines
write.table(data, "C:/...")

# print
cat(...); sink()
```

Numerical Data Analysis

```
# descriptive stats, location
length(x); summary(x); mean(x); median(x); quantile(x)
# descriptive stats, variability
range(x); var(x); sd(x); IQR(x);  x[order(x)[1:5]] # largest 5
# skewness
skew <- function(x){
  n<-length(x); m3<-mean((x-mean(x))^3); m2<-mean((x-mean(x))^2);
  sk=m3/m2^(3/2)*sqrt(n*(n-1))/(n-2); return(sk)
}

# Histogram w. Density Plot
hist(mark, freq=FALSE, main = "Hist"),
  xlab = "mark", ylab="fre", axes = TRUE,
  col = "grey", nclass = 10)
x<-seq(0,30,length.out=98); y<-dnorm(x,mean(mark),sd(mark))
lines(x, y, col = "red")
# Boxplots
boxplot(mark, xlab = "mark")
# QQ plots
qqnorm(mark, pch = 20); qqline(mark, col = "red")

# For association between two
cor(v1, v2) # Correlation
plot(v1, v2, pch = 20) # scatter
boxplot(energy ~ type) # Boxplots by Group

# Others
par(mfrow=c(2,2)); ...; # Subplots
par(new=TRUE); ...; # add new plots to same graph
```

Robust Estimators

```
mean(x); mean(x, trim = 0.2) # arithmetic & 20% trimmed
winsor <- function(x, alpha = 0.2){
  n = length(x); xq = n * alpha; x = sort(x)
  m = x[(round(xq)+1)]; M = x[(n - round(xq))]
  x[which(x<m)] = m; x[which(x>M)] = M
  return(c(mean(x),var(x)))
}
# 20% Winsorized using library
library(MASS); hubers(x, k= 0.84)

median(abs(x-median(x))); mad(x); IQR(x)
```

Categorical Data Analysis

```
count = table(data$type); count # frequency table
barplot(count) # barplot

# Contingency table
ct <- matrix(c(...), ncol=2, byrow=2)
dimnames(ct) <- list(rowname=c(...), colname=c(...))
test <- prop.test(ct,correct=FALSE)
RR <- (test$estimate[1])/(test$estimate[2])
odds <- test$estimate/(1- test$estimate)
OR <- odds[1]/odds[2]

# Fisher Exact Test
fisher.test(ct, alternative = "two.sided")
# general Chi-squared Test
chisq.test(ct)
```

Python Coding

Basic Syntax

```
# matrix
mat = np.asmatrix([[...],[...],...])
np.vstack((...)); np.column_stack((...))
mat.T; mat.I
# Dataframe
data = {'X': [...], 'Y': [...]}
df = pd.DataFrame(data, columns =['X', 'Y'])
df1 = df.rename({'X': 'NewX', 'Y': 'NewY'}, axis=1)
```

Numerical Data Analysis

```
# Relevant libraries
import pandas as pd
import scipy.stats as scst
import matplotlib.pyplot as plt

# Descriptive stats
df['x'].median(); df['x'].var(); df['x'].std()
df['x'].quantile(0.25); df['x'].quantile(0.75)
# Histogram w. Density Plot
l = list(np.arange(0,30,0.5))
y = scst.norm.pdf(l,loc=mean(x),scale=st.stdev(x)) # qnorm in R
plt.plot(l, y); plt.hist(data['x1'], density=True)
plt.title('...'); plt.xlabel('...'); plt.ylabel('...')
# Boxplot
plt.boxplot(data['x1'])
# QQ Plot
scst.probplot(x, dist="norm", plot=pylab)
pylab.show()

# Scatterplot
plt.scatter(v1, v2)
# Scatterplot by Group (tut3Qn2)
groups = data.groupby("x11")
for name, grp in groups:
    plt.plot(grp["x"], grp["y"],
             marker="o", linestyle="", label=name)
# Boxplots by Group
fig, ax = plt.subplots(figsize=(7,5))
bats.boxplot(column=['energy'], by='type', ax=ax, color = 'b')

# Others
plt.legend()
plt.show()
```

Categorical Data Analysis

```
from statsmodels.stats.contingency_tables import mcnemar

# Table
tab = pd.crosstab(index=data["type"],columns="count")
# Barplots
plt.bar(type,counts)

# Contingency table, using df or Numpy 2D array
scst.chi2_contingency(ctable, correction = True)
# Fisher Exact Test
scst.fisher_exact(ctable, alternative='two-sided')
# McNemar Test
mcnemar(ctable, exact=False, correction=True)
# General Chi-squared
scst.chi2_contingency(obs, correction = True)
```

SAS Coding