Applied Data Science

DATA ANALYSIS ROADMAP

Shen, Yuchen

Zhejiang University

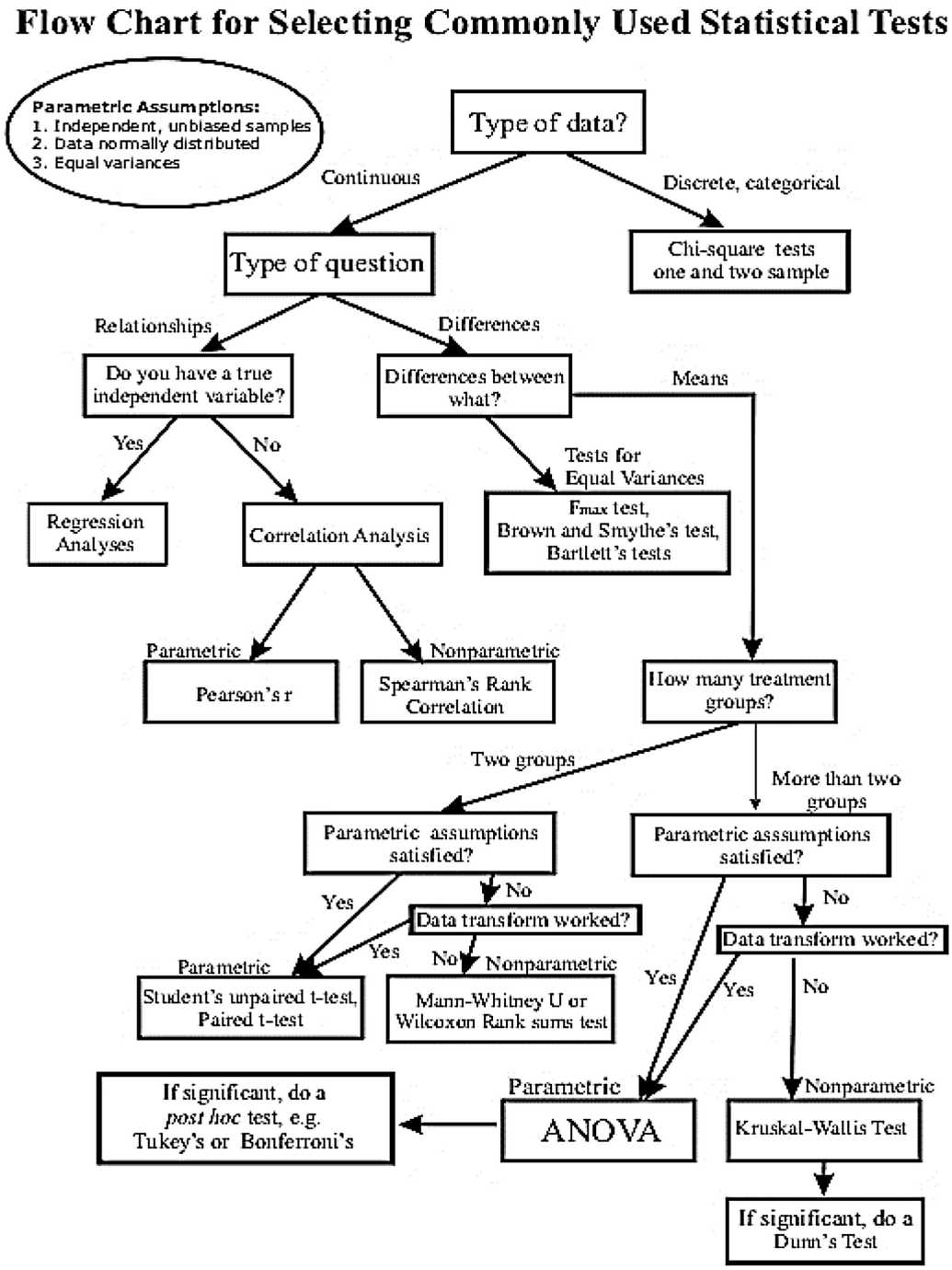
Data Analysis Roadmap

Yuchen SHEN

Table of contents

TABLE OF CONTENTS

Overview



General Reminders

1. It is of extreme importance to notice *if the data is paired*, e.g., one before drug injection one after drug injection. If so, only the *difference* between two time points are meant to used. So you should calculate it correspondingly.
2. One hack of re-structuring data during data cleaning is to let the independent variables be column names, while dependent variables be the filling values.
3. All the plots must be colored.
4. Just don't choose Kruskal-Wallis test when running ANOVA.
5. ANOVA plots = boxplot/violin plot (preferred, go visit www.r-graph-gallery.com)

Data import, exploration & cleaning

Import the Data

```
#Note: set ```{r eval=T, echo=F}
data <- read.table(file = "", header = , sep = "\t") # .tsv + .txt
data <- read_csv(file = "", col_names = , sep = "") # .csv
```

If built-in datasets are to be used, such as `data("cars")`:

```
data("cars")
Cars <- cars # next, use `Cars`
```

Explore the data (must)

Explore the data to see if needs data cleaning.

```
str(data)
head(data) # must be written in the report!!!
table(data$col) # check the distribution of a column
```

Self-exploration (set `echo = F`)

```
dfSummary(data,graph.col=TRUE,style="grid") %>% stview()
```

Data cleaning

- Explanation
- Cleaning for each block
- Final re-examination

1 Remove unnecessary columns

Remove the column if that index is not necessary.

```
data <- data[,-x]
```

2 Check NAs (must)

First, to find out if any missing value exists.

```
anyNA(data)
```

Next, to clarify where exactly are they.

```
which(is.na(data), arr.ind = TRUE) # To see the index of NAs

data[apply(data, 1, function(row) { # To see the rows with NAs
  any(is.na(row))
}), ]
```

Why NAs are associated with [experimental design] is unknown and worth discussion. But since we did not obtain sufficient evidence from the question to retain these abnormal values, we can just drop them.

```
data.noNA = data[complete.cases(data), ] # One way
df_after <- df_before %>% drop_na() # Another way
```

3 Check duplications (must)

First, to see if any duplicated entries exist and where are they.

```r
data[duplicated(data)  duplicated(data, fromLast = TRUE), ]
which(duplicated(data)  duplicated(data, fromLast = TRUE))
```

Since duplicated entries are evident mistakes, they must be removed from the table.

```r
data <- data[!duplicated(data),]
```

4 Check Typos (must)

```r
table(data$col)
```

That appears to be a typo, so next just zoom in and confirm.

```r
data %>%
  filter(col == "A weird value")
```

A clear typo confirmed. Need to correct it.

```r
data.noTypo <- data
data.noTypo[data.noTypo$X== row number, "A weird value"] <- "Corrected value"
table(data.noTypo$col)
```

5 Change long/wide data format

Long format

ID	Treatment	Measurement	Glucose	Comment
1	Vehicle	Glucose_before	11.51	
2	Vehicle	Glucose_before	10.95	Died
3	Vehicle	Glucose_before	9.54	

→

Wide format

ID	Treatment	Comment	Glucose_before	Glucose_after
1	Vehicle		11.51	12.24
2	Vehicle	Died	10.95	NA
3	Vehicle		9.54	7.98

```r
# Option 1
df_wide <- df_long %>%
  spread(key = Measurement, value = Glucose)

# Option 2
df_wide <- df_long %>%
  pivot_wider(
    names_from = Measurement, # Column name
    values_from = Glucose # Values of column
    )
```

Wide format

ID	Treatment	Comment	Glucose_before	Glucose_after
1	Vehicle		11.51	12.24
2	Vehicle	Died	10.95	NA
3	Vehicle		9.54	7.98

→

Long format

ID	Treatment	Measurement	Glucose	Comment
1	Vehicle	Glucose_before	11.51	
2	Vehicle	Glucose_before	10.95	Died
3	Vehicle	Glucose_before	9.54	

```r
# Option 1
df_long <- df_wide %>%
  gather(key = "Measurement", value = "Glucose",
         Glucose_before, Glucose_after)

# Option 2
df_long <- df_wide %>%
```

```
  pivot_longer(
    cols = starts_with("Glucose"),
    names_to = "Measurement",
    values_to = "Glucose"
  )
```

6 Rename

Set column name

```
colnames(df)[3] <- "new_column_name" # option 1

df <- df %>% # option 2
  rename(NewName1 = OldName1,
         NewName2 = OldName2)
```

Set row name

```
increasing_rownames <- as.character(1:nrow(df1))
rownames(df1) <- increasing_rownames
# or
rownames(df)[3] <- "new_row_name"
```

7 Convert data type

Especially check independent variable must be `factor`.

```
str(data)
```

The [data's column] is [old `data type`], should be convert to [new `data type`].

```
data$col1 <- as.factor(data$col1)

# Organize more than 1 columns
data <- data %>%
  mutate(col2 = factor(col1, levels = c(0.5, 1, 2), ordered = T),
         col1 = as.factor(col1)) %>%
  relocate(col1, col2)
```

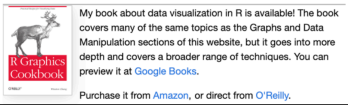
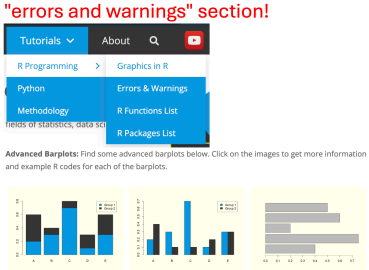
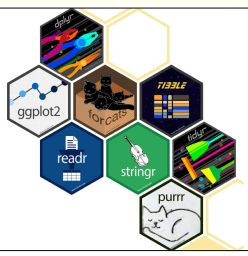
8 Check outliers

```
plot(density(data$col),
     main = "Check outliers for [data]'s [col]",
     lwd = 3)
lines(density(data$col),
      col = "coral2",
      lwd = 3)
```

Now data is clean

Visualization

Variable type	Suggested plots
Categorical X	See below
– With unpaired continuous y	Boxplot, Whisker plot (+SD/SE/CI/IQR), Strip chart
– With paired continuous y	Boxplot (differences), Strip chart with linked points, Histogram of differences
– With categorical y	Bar chart, Table (with percentages), Pie chart
Continuous X	2D Scatter plot, Trend line

Stat + Plot	https://www.statology.org/tutorials/	Different types of statistical operations and plottings: explanations with their implications in R.
R Operations	https://www.statology.org/r-guides/	R operation walk through
Stat	https://statisticsbyjim.com/	
Plot	https://r-graph-gallery.com/	
Plot	https://www.r-charts.com	
R Markdown	https://bookdown.org/yihui/bookdown/	
R Operations	https://www.cookbook-r.com/	
R Operations + ***Stat***	https://rc2e.com/	VERY VERY VERY DETAILED VERY DETAILED FOR MARKDOWN, STATS, OPERATIONS... **R Cookbook, 2nd Edition** *James (JD) Long* *Paul Teetor* *2019-09-26* 
Plot	https://r-graphics.org/	Very detailed information about each type of graph 
Math	https://www.desmos.com	
R Operation + ***R Markdown***	https://intro2r.com/	Can directly copy; very detailed markdown elaboration. **An Introduction to R** *Alex Douglas, Deon Roos, Francesca Mancini, Ana Couto & David Lusseau* *April 9, 2024*
Plot ***R Package*** ***R Function***	https://www.statisticsglobe.com	Very detailed, well designed. Note the "errors and warnings" section! 
R Packages	https://www.tidyverse.org/	

T-test

Run a t-test

Formulate hypotheses

- **Reminder:**
  - Hypothesis depends on tail and if they are paired!
  - Hypothesis depends on the question!
  - Same batch of cell cultures are paired two-samples. Not for mice.
- **One-tailed**
  - H_0: A is not greater (or smaller) than B
  - H_A: A is greater (or smaller) than B
- **Two-tailed**
  - H_0: The difference between A and B is 0
  - H_A: The difference between A and B **is not** 0
- **Paired two-samples (= one sample)**
  - H_0: The difference between both conditions is 0
  - H_A: The difference between both conditions **is not** 0

State assumptions

The below are for **choosing**:

1. The test units are chosen randomly;
2. The dependent variable is continuous; (3. The mean and standard error are independent.)
3. For **unpaired t-test**, samples must be independent;
4. For **paired t-test**, each pair of measurements must be independent;
5. For **unpaired t-test**, normality must be satisfied.
6. For **paired t-test**, the differences between different conditions must be normally distributed.
7. (For **two-samples** t-test - Since two samples are involved, the homogeneity of variances should be checked.) - `Don't write it.`

Transformation (optional)

If one sample is transformed, then the reference should also be transformed.

Log transformation

```r
# + 1 if data includes 0
transformed_data <- log(data)
```

Square transformation

```r
# + 1 if data includes 0
transformed_data <- sqrt(data)
```

Check assumptions

- **Randomness**:
  - `sampled_data <- sample(data, size = 10, replace = F)`
  - or just explained by words in the question.
- **Continuous dependent variable**:
  - `is.numeric(data$col)`
- **Independency**:
  - Text-explained
- **Normality**:
  - Reminders
    - H_0: the data is normally distributed; H_A: the data is not normally distributed
    - If 2 sample, check for EACH, unless they are PAIRED!
    - Sometimes, need to declare a LACK OF OBSERVATIONS: ...thus, the histogram (and any diagnostic test as well) is difficult to judge. But it is difficult to reject H_0 of normality either. Thus, I assume that the assumption of normality of [ ] is satisfied.
  - Methods
    - **Q-Q plot** `qqnorm(data$col)` with Q-Q line `qqline(data$col, col = "red", lwd = 3)`
      - The data points are evenly distributed along the qqline. So normality is satisfied. We choose t-test instead of non-parametric tests.
    - **Histogram**: `hist(data$col, breaks = 10, main = '', xlab = "", ylab = "")`
      - The data distribute normally. So normality is satisfied. We choose t-test instead of non-parametric tests.
    - **Shapiro-Wilk test**: `test_res <- shapiro.test(data$col)`

        - p > 0.05: the null hypothesis of normality is not rejected (ideal).
- **Variance Homogeneity (if 2-sample)**
  - F-test: `var.test(data$col1, data$col2)` or `numeric ~ factor, data`
    - p > 0.05: student's t-test
    - p <= 0.05: welch's t-test
      - p-value = [ ], So we use Student's t-test instead of Welch's t-test.
- **Mean and standard error independency**
  - Each sample is independent and the sampling process is random, so this assumption is automatically satisfied.

Select the t-test

Questions (ask one by one)	Answer
1. Is it between 1 sample & value or 2 samples?	1 sample & a value → 1-sample test 2 samples → 2-samples test
2. *If are 2-samples*, are the variances equal?	Yes → Student's t-test No → Welch's t-test
3. Is the comparison direction stated?	Yes → 1-tailed No → 2-tailed
4. *If are 2-samples*, across different time points/conditions, is the object the same?	Yes → paired No → unpaired
5. Failed to satisfy normality?	Yes → Non-parametric test No → Student's t-test
6. For non-parametric test:	Paired/One-sample → Wilcoxon signed-rank test Unpaired → Mann-Whitney U test

- We got [two] groups.
- They are [not] paired.
- The direction of comparison is [not] stated.

So we use [ ]-tail [ ]-samples [ ]paired [Student's] t-test/non-parametric test.

Run a t-test

- **Student's t-test 1**

```r
sd(x)
sd(y)
```

```r
t.test(x,
       y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0,
       paired = F,
       var.equal = F
       )
```

- **Student's t-test 2**

```r
sd(x)
sd(y)
```

```r
t.test(DN ~ Condition, apoptosis, paired = T)
```

Compare whether the mean of DN is significantly different under two different Conditions.

- **Non-parametric test**: same as t-test other than the function names.
  - Wilcoxon signed-rank test
    - `wilcox.test(paired = TRUE)`
  - Mann-Whitney U test
    - `wilcox.test(paired = FALSE)`

Report the findings

- **Any unpaired test**
  - There was a [increase/decrease] of alcohol consumed in the week after the end of semester (M = 8.7, SD = 3.1) compared to the week before the end of semester (M = 3.2, SD = 1.5), $t(52) = 4.8$, $p < 0.001$.
  - But that was [not] significant. So we can/cannot reject the null hypothesis.

  - Note: `(t() = dof)`
- **Paired test**
  - The average difference between the treatment condition is 7.82% ($P_{(t)} = 0.1$). Thus, we cannot reject H_0.

ANOVA

Overview

- **Purpose**: compare means of more than two groups
  - more than 2 populations
    - life expectancy of China, Canada and Japan
  - more than 1 predictor
    - effect of diet (1_{st} predictor) and exercise (2_{nd} predictor) on weight
  - exactly 2 populations
    - one-way ANOVA = independent two-sample t-test
    - $F = t^2$
- **F-statistic** = $\frac{VarianceBetweenGroups}{VarianceWithinGroups}$
  - the higher, the more likely to reject H_0
- **Types (focused)**
  - 1-way ANOVA: 1 factor
    - e.g. effect of 3 doses of drug A on heart rate
  - 2-way ANOVA: 2 factors
    - e.g. effect of age and sex on salary
    - Interactions (3 p-values):
      - Does the first factor influence the outcome?
      - Does the second factor influence the outcome?
      - Is there an interaction between both factors?

1-way ANOVA by simulation

See practical notes for details.

1-way ANOVA by statistical tests

Visualization

```r
ggplot(data, aes(x = GroupLabelColumn, y = ValueColumn)) +
  geom_boxplot() +
  geom_jitter(width = 0.1, alpha = 0.5) +
  labs(title = "", x = "", y = "") +
  theme_minimal()
```

Justify statistical choice

- We want to compare the means of more than 2 groups
- There is 1 factor to compare

So we should try to use a 1-way ANOVA (parametric, use mean) if the data fit the requirements or we can run Kruskal-Wallis test (rank, use median).

Construct ANOVA model

```r
model <- aov(Value ~ GroupingFactor, data = data)
```

Test assumptions

We still need to test assumptions first!

- **Independent random sampling**: from the given description of the experiment, we believe this assumption is met.
- **Normality of residuals**:
  - Q-Q plot: `plot(model, 2)`. Dots should be aligned along diagonal
  - Histogram: `hist(resid(model), main = 'residuals')`. Normality should be directly seen
  - Shapiro-Wilk test: `shapiro.test(resid(model))`. If p-value > 0.05, the null hypothesis of normality is not rejected (ideal).
- **Equality of variance**:
  - Residuals vs fitted plot: `plot(model, 1)`. Dots should be randomly scattered, similar heights of 'columns' should be seen.

Formulate hypotheses

- H_0: there is no difference in means between [`grouping factor name`] groups
- H_A: at least one [`grouping factor name`] group mean is different from the others

ANOVA

Perform the actual ANOVA

```r
summary(ANOVA)
```

Post-hoc test (*required even if not asked*)

```r
TukeyHSD(model)
```

Visualize the post-hoc results

```r
TukeyHSD(model) %>%
  as.data.frame() %>%
  ggplot(aes(x = diff, y = reorder(comparison, diff))) +
  geom_point() +
  geom_errorbarh(aes(xmin = lwr, xmax = upr), height = 0.2) +
  labs(title = "Tukey's HSD Post-hoc Test",
       x = "Difference in Means",
       y = "Comparison") +
  theme_minimal()
```

Present and interpret the results

Subjects treated with [intervention name or dose] showed a [increase/decrease] in [response variable] compared to before treatment, with a mean change of [effect size in units]. However, this difference [did/did not] reach statistical significance (red{p = [p-value]}). There was [no effect / minimal effect / mixed effect] observed at the [lower/higher/alternate] dose compared with the control group.
Despite the lack of statistical significance, the effect size measures indicate a [small/moderate/large] effect:

- Eta squared (η^2) = [value], suggesting that [X]% of the variability is explained by the treatment *(small: 0.01, medium: 0.06, large: 0.14)*;
- Cohen's f = [value], which corresponds to a [small/medium/large] effect according to conventional thresholds *(small: 0.1, medium: 0.25, large: 0.4)*.

2-way ANOVA by statistical tests

Question: explore the effects of attendance and previous grades on course performance

Visualization

```r
ggplot(data, aes(x = Attendance, y = Performance, color = PreviousGrades)) +
  geom_boxplot() +
  geom_jitter(width = 0.1, alpha = 0.5) +
  labs(title = "Course Performance by Attendance and Previous Grades",
       x = "Attendance",
       y = "Performance") +
  theme_minimal()
```

Justify statistical choice

- There is [ ] groups to compare
- We are looking at the effects of more than 1 predictors

So we should try to use a 2-way ANOVA if the data fit the requirements. I see no reason to run ANOVA without interactions, so I would like to have a model like Course performace = $Attandence + PreviousGrades + interaction$.

Construct ANOVA model

```r
model <- aov(weight_gain ~ genotype * diet, data = mouse) # 2 factors, with interaction
model <- aov(weight_gain ~ genotype + diet, data = mouse) # 2 factors, without interaction
```

Test assumptions

We still need to test assumptions first!

- **Independent random sampling**: from the given description of the experiment, we believe this assumption is met.
- **Normality of residuals**:
  - Q-Q plot: `plot(model, 2)`. Dots should be aligned along diagonal

ANOVA

  - Histogram: `hist(resid(model), main = 'residuals')`. Normality should be directly seen
  - Shapiro-Wilk test: `shapiro.test(resid(model))`. If p-value > 0.05, the null hypothesis of normality is not rejected (ideal).
- **Equality of variance**:
  - Residuals vs fitted plot: `plot(model, 1)`. Dots should be randomly scattered, similar heights of 'columns' should be seen.
- **Equality of group size**:
  - The group size can be noticed in the data diagnosis step.

So we use *parametric* ANOVA.

Formulate hypotheses

- Hypotheses set 1 for *main effect*:
  - H_0: the mean of [y] of [x1] and [x2] is the same
  - H_A: the mean of [y] of [x1] and [x2] is not the same
- Hypotheses set 2 for *interactions*:
  - H_0: there is no interaction between [x1] and [x2]
  - H_A: there is an interaction between [x1] and [x2]

Perform the actual ANOVA

```
summary(model)
```

Post-hoc test (*required even if not asked*)

```
TukeyHSD(model)
```

Visualize the post-hoc results

```
TukeyHSD(model) %>%
  as.data.frame() %>%
  ggplot(aes(x = diff, y = reorder(comparison, diff))) +
  geom_point() +
  geom_errorbarh(aes(xmin = lwr, xmax = upr), height = 0.2) +
  labs(title = "Tukey's HSD Post-hoc Test",
       x = "Difference in Means",
       y = "Comparison") +
  theme_minimal()
```

Present and interpret the results

- Overall statement: almost all terms show significant differences. This indicates that the [independent factors] have distinct effects on [response variable];
- Main effect A: treatment 1 is generally better than treatment 2 (~ `diff`, $p < 0.05$);
- Main effect B: treatment I is generally better than treatment II (~ `diff`, $p < 0.05$);
- Interaction: the interaction between Factor A and Factor B is also statistically detectable ($p =$ `interaction p-value`); Or: the interaction term was not significant ($p =$ `interaction p-value`), indicating that the effect of Factor A does not depend on the level of Factor B.
- Conclusion: altogether, these results suggest that [interpretation: e.g., the new formulation can substitute the natural source only under certain conditions].

Power Analysis

Overview

- **Definition**: statistical power is the likelihood that a study will detect an effect when THERE IS an effect there to be detected
- **Scenario**: Lack of statistical significance does not prove that there is no difference, instead, it may be a consequence of low power
- **Power requirements**
- **Effect factors**
  - Effect size $\Delta\mu = \mu_1 - \mu_0$: Power $\propto \frac{|\mu_1 - \mu_0|}{\sigma}$
  - Standard deviation σ: higher $\sigma \Rightarrow$ wider distributions $\Rightarrow$ more overlap $\Rightarrow$ lower power
  - Sample size n: SE $= \frac{\sigma}{\sqrt{n}} \Rightarrow$ larger $n \Rightarrow$ smaller SE $\Rightarrow$ higher power
  - Desired significance level α: higher $\alpha \Rightarrow$ lower $\beta \Rightarrow$ higher power

Balance α and β level

- **General rules**
  - A Type I error is considered worse so α is rarely > 0.05
  - If we can tolerate a 5% type I error, we can tolerate a 20% type II error
  - Some say $Type I error = Type II error$
- **Clinical settings**
  - Large clinical trials use 0.9 or 0.95 (90-95% power, $\beta = 0.1 - 0.05$)
  - Clinical trials phase III: $min = 80$
- **Animal studies**
  - Animal studies usually use 0.8 (80% power, $\beta = 0.2$)
- **Omics studies**
  - Omics studies: aim for high power, because you want to minimize Type II error

T-test: calculate power (simulation)

- **Given information**
  - Sample size (paired): $n = 5$
  - Mean difference: $\Delta = 7.82$
  - Standard deviation: $\sigma = 8.20$
  - Significance level: $\alpha = 0.05$

```r
ps <- replicate(1e5, t.test(rnorm(5, 7.82, 8.2), mu = 0)$p.value)
power <- length(which(ps < 0.05)) / 1e5
power
```

E.g., if the drug is indeed effective as showed on the animal model, then what is the probability that they do not see a significant effect of the drug (p-value cutoff = 0.05)?

```r
p.10 <- replicate(1e5, t.test(rnorm(10, 117, sd = 30), mu = 130, alternative = "less")$p.value)
length(which(p.10 <= 0.05)) / length(p.10) * 100
```

T-test: calculate power (built-in function)

- **Given information**: same as above

```r
power.t.test(n = 5, delta = 7.80, sd = 8.2, sig.level = 0.05, type = "paired")
```

T-test: calculate the minimum sample size for an aimed power (simulation)

- **Goal**: what n gives us 80% power?
- **Interpretation**: $power = 0.8$, so $\beta = 0.2$
- **Given information**: same as above

```r
power.t.test(power = 0.8, delta = 7.82, sd = 8.2, sig.level = 0.05, type = "two.sample")
# Output: n = 18.27 (per group)
# next, clarify which one is best: 18 or 19
power.t.test(delta = 7.82, sd = 8.2, sig.level = 0.05, n = 18, type = "two.sample")
power.t.test(delta = 7.82, sd = 8.2, sig.level = 0.05, n = 19, type = "two.sample")
```

T-test: parameter settings (both)

- **Consider**:
  - Study design (e.g., paired or unpaired …)
  - Effective or not in reality
  - See the effect or not

Simulation

A college is providing a free bottle of milk per day to the students. **Based on the history of a nearby country, giving a bottle of milk to the college students could improve the average height from 175cm to 178cm.** Now, you go to the college and measure the heights of 10 male students, and perform a t-test to see if the students are **indeed higher than** the average of the country. Simulate the data you get from the students, and record the p-value in the t.test, which is the number?

```r
p <- replicate(1e4, t.test(rnorm(10, 178, sd = 10), mu = 175,
                           alternative = "greater")$p.value) # Note `alternative = "greater"`
length(which(p <= 0.05)) / length(p) * 100 # Note `p <= 0.05`
```

If the **drug is indeed effective as showed on the animal model**, then what is the probability that they **do not see a significant effect** of the drug (p-value cutoff = 0.05)? Please perform a simulation as you did before to give an answer.

```r
p <- replicate(1e5, t.test(rnorm(10, 117, sd = 30), mu = 130,
                           alternative = "less")$p.value) # Note `alternative = "less"`
length(which(p <= 0.05)) / length(p) * 100 # Note `p <= 0.05`
```

Built-in functions

If the company truly believes the effect of the drug and want to be sure that they will **not largely miss the effect** in the trial (type II error rate < 0.2), then how many volunteers do they need to recruit?

```r
power.t.test(power = 0.8, delta = 13, sd = 30, sig.level = 0.05,
             type = "two.sample", alternative = "one.sided") # Note: use `"one.sided"`
```

If the company changes their strategy, asking **all the volunteers to take the pills** and measuring their weights **before and afterward**, how many volunteers do they need?

```r
power.t.test(power = 0.8, delta = 13, sd = 30, sig.level = 0.05,
             type = "paired", alternative = "one.sided") # Note: use `type = "paired"`
```

ANOVA: calculate power (simulation)

```r
sample_sig_interactions <- sapply(seq(0, 100, 5), function(sample_size) {
  mean(replicate(100, {
    sample_mouse <- mouse[sample(nrow(mouse), sample_size), ]
    sample_model <- aov(weight_gain ~ genotype * diet, data = sample_mouse)
    sample_interaction_pvalue <- summary(sample_model)[[1]][3, 5]
    sample_interaction_pvalue <= 0.05 # see the effect
  })) * 100
})

# Get sample size closest to 80% power
sample_sizes <- seq(0, 100, 5)
min_sample_size <- sample_sizes[which.min(abs(sample_sig_interactions - 80))]
print(min_sample_size)
```

ANOVA: calculate the minimum sample size for an aimed power (simulation)

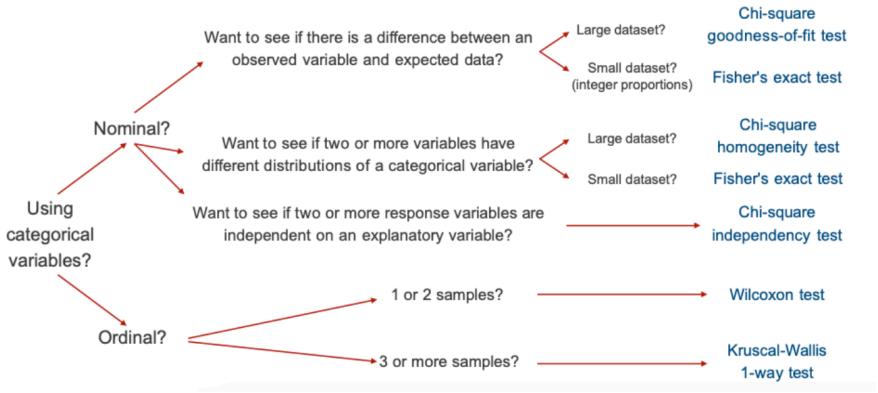
```r
power_at_60 <- mean(replicate(100, {
  sample_mouse <- mouse[sample(nrow(mouse), 60), ]
  sample_model <- aov(weight_gain ~ genotype * diet, data = sample_mouse)
  sample_interaction_pvalue <- summary(sample_model)[[1]][3, 5]
  sample_interaction_pvalue <= 0.05
```

```
})) * 100
print(power_at_60)
```

For an integrated method: see week 2 practical notes.

Chi-Squared Test

Overview



Test Type	#	Purpose	Scenario
Goodness-of-Fit	1	Does the observed distribution match an expected one?	- Is there *a difference between* the season preferences? - Is their estimation *correct*?
Test of Independence	2	Are two categorical variables dependent (aka affecting each other)?	- Does gene X *affect* mice lifespan?
Test of Homogeneity	2	Are distributions of a categorical variable the same across groups?	- Is there a difference between the *distribution* of allergic reactions *in different seasons?*
3-way	3	3 categorical variables	- Are there relationships between gene X, sex and lifespan?
Fisher's Exact Test	x	For small sample sizes's test for independence	

Goodness-of-Fit Test

From dataframe to table

```
table(genotype$sex, genotype$genotype) # sex: row name
```

Visualize

```
t <- as.data.frame(table(genotype$sex, genotype$genotype))
pie(
  t$Freq[t$Var1 == "male"],
  labels = t$Var2[t$Var1 == "male"],
  main   =  "")
)
```

What would you expect?

Make a markdown table (table 1) and describe with a few sentences.

Choose and justify the appropriate statistical test

- It is an expected vs observed question.

So we use Chi-squared goodness-of-fit test.

Test Assumptions

- The variables must be categorical. *Fits, due to our examination.*
- Observations must be independent. *Fits, can assume from the task.*
- Cells in the contingency table are mutually exclusive. *Fits, due to our examination.*
- The expected value of cells should be 5 or greater in at least 80% of cells. *Fits, see table 1.*

Thus, we can run Chi-squared goodness-of-fit test safely.

Form hypotheses

- H_0: The data follow the expected distribution (table 1).
- H_a: The data does not follow the expected distribution (table 1).

Perform Chi-squared test

Note: values of the expected dataframe should be provided in *frequency*!

```r
Seasons <- c(40, 30, 18, 28)
chisq.test(Seasons, p = rep(0.25, 4), correct = F)
```

Present and discuss the results

Clearly, [independent variable] affects the [dependent variable]. Observed objects are [percent]% ([`p >/<`]) of the expected number. Moreover, ...(if any)

Calculate p-value (not required)

```r
pchisq(X2, dOf, lower.tail = FALSE)
```

Test for homogeneity

Form hypotheses

- H_0: the distribution of allergic reactions (a variable) is the same for people who prefer different seasons (another variable).
- H_a: the distribution of allergic reactions (a variable) is not the same for people who prefer different seasons (another variable).

Perform the test: Chi-squared or Fisher's exact test

```r
# Two_categories: a dataframe with the column being seasons and the row being allergic reactions
chisq.test(Two_categories)
fisher.test(Two_categories)
```

Test for independence

Form hypotheses

- H_0: mice survival (a variable) is independent of gene X (another variable).
- H_a: mice survival (a variable) is dependent of gene X (another variable).

Perform the test: Chi-squared or Fisher's exact test

```r
# Two_categories: a dataframe with the column being gene X and the row being mice survival
chisq.test(Two_categories)
fisher.test(Two_categories)
```

3-way Chi-squared test

dof

- dof = (#1st variable)*(#2nd variable)*(#3rd variable) - 1 - (#1st variable - 1) - (#2nd variable - 1) - (#3rd variable - 1)

Form hypotheses

- H_0: there is no inter-dependency between gene X, sex and lifespan
- H_a: there is an inter-dependency between gene X, sex and lifespan

Bootstrapping

Overview

- **When to use**:
  - Don't know population distribution
  - The *complex distribution (e.g., non-normal distribution)* limit the use of classical statistical methods (t-test, ANOVA…)
  - The population size is extremely small
  - Data lacks independence
  - Options of categorical data are not mutually exclusive
- **What for?**
- For hypothesis testing
  - for a particular pair
  - simulate, with my own efforts, how distribution would look like if H_0 was true
  - then examine if the observed distribution is extreme (compare p and α)
- For calculating confidence interval
  - not for a particular pair, multiple categories exist
  - after repeating, rank acquired values
  - get 2.5% as lower bound and 97.5% as upper bound for 95% Confidence Interval

Justify statistical test choice

- [Characteristics of this study] + [So A test and B tests are not appropriate].
- Bootstrapping is the only possible alternative to take account of these complications.

Steps

- Make a bootstrapped sample
  - Note: sampling from original data *with replacement*
- Calculate something (mean, median, sd…)
- Record the calculation
- Repeat step 1 ~ 3 for a bunch of times
  - Note: the number of bootstrapping replication will NOT change p-value systematically, but WILL narrow the p-value variation, that is, the more likely you will get a stable and accurate p-value

Examine original distribution

I first assess whether the data is normally distributed which would allow me to perform parametric tests such as a t-test.

```
shapiro.test(data$value)
hist(data$value)
```

Since [ ], the Shapiro-Wilks test shows that the data in normally distributed, however, the histogram disagrees.
Probably the lack of significance in the Shapiro-Wilks test is due to a lack of statistical power rather than the data being truly normally distributed. To be conservative, I choose Bootstrapping.

Bootstrap for hypothesis testing

The question: is group 1 and group 2 significantly different?

Formulate hypotheses

- H_0: there is no difference between the median of group 1 and group 2
- H_A: there is a difference between the median of group 1 and group 2

Visualization with box plot

```
boxplot(
  # datavalue~datacategory,
  # ylim = c(0, 10),
  las = 2,
  # ylab, xlab, main
)
```

The actual condition:

```
group1 <- median(subset(data, data$col ==  "Group1")) # might change for mean or sd with `sd()`
group2 <- median(subset(data, data$col ==  "Group2"))
median_diff <- group1 - group2
median_diff
```

BOOTSTRAPPING

Simulate the world when H_0 is true

```
number_group1 <- nrow(subset(data, data$col == "Group1"))
number_group2 <- nrow(subset(data, data$col == "Group2"))

# for reproducibility
set.seed(123)

# for recording
bootstrap_median_record <- vector()

# repeats
for (a in 1:100) {
  # no filtering, but with same number
  bootstrap_group1 <- median(sample(data$col, number_group1, replace = TRUE))
  bootstrap_group2 <- median(sample(data$col, number_group2, replace = TRUE))
  bootstrap_median <- bootstrap_group1 - bootstrap_group2
  bootstrap_median_record <- c(bootstrap_median_record, bootstrap_median)
}

# visualization
hist(bootstrap_median_record)
abline(v = median_diff, col = 'red') # for comparison
```

To get the p-value for reaching a definitive conclusion

```
length(subset(bootstrap_median_record, bootstrap_median_record >= median_diff))/100
```

- I then obtain a p-value for this test by determining how many of my bootstrap values have a more extreme value than my observed value. In this case, this length is [ ] which corresponds to p=n.
- n > 0.05, which prevents me from rejecting the H_0 at the standard p-value = 0.05 threshold.
- I can therefore conclude that there is no difference in [ ]

Bootstrap for confidence interval

The question: what is the possible range for a statistic for several groups?

Formulate hypotheses

- H_0: there is no difference in values between categories
- H_A: there is a difference in values between categories

Visualization with dot plot

```
plot(
  data$value,
  # ylab, xlab
  xaxt = "n"
)

axis(
  side = 1,
  at = seq(1, nrow(data), 1),
  labels = data$category,
  las = 2
)
```

Repeat the bootstrapping for the entire dataset (also a duo-comparison below).

```
total_values <- sum(data$values)
this_value_record <- vector()
lower_bound_record <- vector()
upper_bound_record <- vector()

for (row in 1:nrow(data)) {
```

```
  # define
  this_category <- data[row, 1]
  this_value <- data[row, 2]
  not_this_value <- total_values - this_value

  # create a new dataframe with:
  # category repeat for the value times &
  # categorize into only two classes
  re-classified <- c(rep(this_category, this_value),
                     rep("not_this_category", not_this_value))

  # initialize boostrapping record
  bootstrap_record <- vector()

  for (replication_number in 1:100) {

    # the actual bootstrapping
    bootstrap_sample <- sample(re-classified, length(ore-classified), replace = T)

    # write record
    bootstrap_record <- c(bootstrap_record,
                          length(
                            subset(bootstrap_sample, bootstrap_sample == this_category)
                                 )
                          )
  }

  lower_bound <- quantile(bootstrap_record, 0.025)
  upper_bound <- quantile(bootstrap_record, 0.975)

  this_value_record <- c(this_value_record, this_value)
  lower_cis <- c(lower_bound_record, lower_bound)
  upper_cis <- c(upper_bound_record, upper_bound)
}
```

Visualize the results in a useful way

```
ymax <- ceiling(max(upper_bound_record) * 100) / 100

plot(
  this_value_record,
  xaxt = "n",
  ylim = c(0, ymax),
  # xlab, ylab
  pch = "."
)

axis(
  side = 1,
  at = seq(1, nrow(data), 1),
  labels = data$category,
  las = 2
)

for (a in 1:length(lower_bound_record)) {
  lines(x = c(a, a), y = c(lower_bound_record[a], upper_bound_record[a]))
  lines(x = c(a- 0.1, a + 0.1),
  y = c(lower_bound_record[a], lower_bound_record[a]))
  lines(x = c(a- 0.1, a + 0.1),
  y = c(upper_bound_record[a], upper_bound_record[a]))
}

points(
```

```
    x = seq(1, nrow(data), 1),
    y = this_value_record,
    pch = 20
)
```

- As the confidence intervals defined here represent a [95%] certainty that the true proportion of new cases is within this interval
- I can therefore conclude that pairs of categories with non-overlapping confidence intervals are significantly different at the standard p-value threshold of [0.05].

```
# Initialize the values & vectors
first_satisfied <- 864
first_unsatisfied <- 714
second_satisfied <- 980
second_unsatisfied <- 473
first_bootstraps <- vector()
second_bootstraps <- vector()

# Create tag vectors
first_results <- c(rep(1, first_satisfied), rep(0, first_unsatisfied))
second_results <- c(rep(1, second_satisfied), rep(0, second_unsatisfied))

# Bootstrap for 100 times
for (a in 1:100) {
  first_sample <- mean(sample(first_results, length(first_results),
                              replace = TRUE))
  second_sample <- mean(sample(second_results, length(second_results),
                               replace = TRUE))
  first_bootstraps <- c(first_bootstraps, first_sample)
  second_bootstraps <- c(second_bootstraps, second_sample)
}

# Generate CI
first_upper <- quantile(first_bootstraps, probs = c(0.975))
second_lower <- quantile(second_bootstraps, probs = c(0.025))

# Gives conclusion
first_upper < second_lower
```

Visualization

```
boxplot(first_bootstraps, second_bootstraps, notch = TRUE,
        names = c("early", "late"),
        ylab = "Prop. of satisfied button presses")
```

Correlation and Regression

Regression (assumptions are met)

Formulating a linear model and exploring it

```
model <- lm(y ~ x, dataframe)
```

Check assumptions

- Independence of observations
- Linear relationship between variables
- Homoscedasticity of residuals
- Normal distribution of residuals

1. Independence of observations
If not dependent measurements such as time series: "Independence of observations can be assumed from the task itself."
2. Linear relationship between variables
The relationship between the variables seems to be quite linear according to the scatter plot.

```
plot(x, y, main = " ", xlab = "Age, yrs", ylab = "Weight, kg")
```

3. Homoscedasticity of residuals
Choose one of the following:

- Visual examination: `plot(model, 1)`. Discussion: "The model residuals are not strictly homoscedastic, but the reason to reject homoscedasticity is not sufficient either."
- Statistical test: Breusch-Pagan test with `lmtest::bptest(model)`. If p-value > 0.05, the null hypothesis of homoscedasticity is not rejected (ideal).

4. Normal distribution of residuals
Choose one of the following:

- Visual examination: `plot(model, 2)` or `hist(residuals(model))`. Discussion: The model residuals have some values that do not strictly correspond to the expected values, but the reason to reject normality is not sufficient either.
- Statistical test: Shapiro-Wilk test with `shapiro.test(residuals(model))`. If p-value > 0.05, the null hypothesis of normality is not rejected (ideal).

"Overall, the model is usable. However, some values seem to be out of place. For instance, there are really few measurements from the cars moving at a lower speed. It skews the model."

Check for outliers

```
plot(cooks.distance(model), type = "h",
     main = "Cook's Distance", ylab = "Distance")
```

Since there's no clear reason for why the data at [data point] is abnormal, it is better to be safe than sorry.

Discuss the model

Use `str(model)` or `model` or the following (most often)

```
summary(model)
```

Interpretation:

- F-statistic: if our model is significant.
  - If the p-value is less than 0.05, we can reject the null hypothesis that the model is not significant.
  - So we can conclude that x is not a significant predictor of y.
- R-squared: how much of the variance in y is explained by x.
  - If R-squared is 0.5, it means that 50% of the variance in y is explained by x.

They are both indicators of the model's goodness of fit.
The model [is/is not] statistically significant (p-value $< 0.05 \rightarrow$ significant). The variables correlate at [$r \approx 0.8$] ($R^2 = 0.65$). The model explains about [65%] of the total variance ($R^2 = 0.65$). Thus, the model is quite good. But there [is/is not] still quite a lot of unexplained variance. It would be better to account for more factors.

Visualize the model (optional)

```
ggplot(df, aes(x = Age, y = Weight)) +
  geom_point(aes(shape = "Primary points"), color = "black") +
```

```
  geom_smooth(aes(color = "Model"), method = "lm", se = FALSE) +
  scale_color_manual(values = c("Model" = "red")) +
  scale_shape_manual(values = c("Primary points" = 1)) +
  labs(x = "Speed, km/h", y = "Distance to stop, m") +
  theme_minimal(base_size = 14) +
  theme(
    legend.title = element_blank(),
    legend.position = "right" # bottom/top
  )
```

Use it to predict

```
predict.lm(
  object = model, # The regression model must be here
  newdata = data.frame(Age = c(10, 12, 14, 16)) # Predict multiple at once
) # Input values in a data frame
```

Perform and discuss correlation

```
cor.test(x, y, method = "pearson")
```

Interpretation:

- $0 \leq |r| < 0.3$ — Weak correlation
- $0.3 \leq |r| < 0.5$ — Moderate correlation
- $0.5 \leq |r| < 0.7$ — Strong correlation
- $0.7 \leq |r| \leq 1$ — Very strong correlation

Regression (assumptions NOT met)

Options:

- Transform the data: log, $\log_{10}$, square root
- Remove outliers: identify with `plot(model, 4)` explain and (replace)
- Non-parametric test: Spearman's correlation, non-linear models

Perform and discuss correlation

```
cor.test(x, y, method = "spearman")
```

Optimize a multi-variable model

Formulate start and end model

```
model_1 <- lm(weight ~ age, data = df)
model_full <- lm(weight ~ age + height + sex + diet, data = df)
```

Perform step-wise regression

```
step(
  object = model_1,
  direction = "forward",
  scope = formula(model_full)
)
```

Time Series

Core

- Definition: a random variable is measured during a **prolonged period** of time in **regular intervals**
- Examples: the birth rate in a country over some time; deaths from a certain reason over some time; concentration of a certain compound (in the air/solution) over a certain time period; heart electrical activity; hormone level in the blood
- Components
  - trend m_t
    - linear regression: $x_t = \beta_0 + \beta_1 * t$ - get β_1
    - moving average: $m_t = \frac{1}{k} \sum x_t$ - get m_t
  - cyclicity/seasonality s_t
    - additive model: $s_t = x_t - m_t$ - get s_t
  - error z_t
    - calculated from equation

Construct time series object

```r
library(stats)
library(forecast)

ts_data <- ts(orig_data,
  start = c(year, month),
  frequency = n
)
```

- n = 12: monthly data
- n = 4: quarterly data
- n = 1: yearly data

Visualize time series

```r
plot(ts_data,
  main = "Time Series Plot",
  ylab = "Dependent variable",
  xlab = "Time, yrs",
  col = "blue",
  lwd = 2
)
```

Data transformation (optional)

Methods:

- $\log/\log_{10}$
- standard practice for exponential growth data: `after <- log10(before)`
- Differencing (must integrated after modeling)
- good for random walk data (random walk to white noise)
- may not good for non-random walk data

Check assumptions

- Absence of missing values
- Normality of residuals
- Stationarity of the time series
- Independence of residuals
- Observations made in regular intervals

Navigate the time series component

Is there a trend?

- Fit
- Plot
- Quantify uncertainty

1. Fit the model

```r
ts_data_lm <- lm(ts_data ~ time(ts_data))
summary(ts_data_lm)
```

2. Plot the trend (if using linear regression)

```r
plot(
  time(ts_data),
  ts_data,
  main = "Trend examination",
  ylab = "Dependent variable",
  xlab = "Time, yrs"
)

abline(ts_data_lm, col = "red", lwd = 2)
```

Interpretation: If p-value <0.05: Time is a statistically significant regressor.
3. Confidence interval for the trend to quantify uncertainty

```r
# Extract coefficient and sd
coef_estimate <- coef(summary(ts_data_lm))["time", "Estimate"]
se_estimate <- coef(summary(ts_data_lm))["time", "Std. Error"]

# Calculate 95% CI
coef_estimate - qt(0.975, df = ts_data_lm$df.residual) * se_estimate # lower
coef_estimate + qt(0.975, df = ts_data_lm$df.residual) * se_estimate # upper
```

Is there a seasonality?

- ACF
- Decompose
- Plot

1. Let's use Autocorrelation Function (ACF) plot to check for repeating patterns.

```r
acf(ts_data, main = "ACF Plot of ts_data")
```

2. Decompose the series (only works for regular frequency series)

```r
ts_data_decomp <- decompose(ts_data)
```

3. Plot the decomposed series

```r
plot(ts_data_decomp, main = "Decomposed Time Series")
```

Predictions

- Create time pin
- Predict
- Convert (optional)
- Plot

1. Create new time points

```r
future_time <- time(ts_data)[length(ts_data)] + seq(1, 10)
```

2. Predict on log-scale in this case

```r
future_preds_log <- predict(ts_data_lm,
  newdata = data.frame(ts_data_log = future_time)
)
```

3. Convert back to original scale (optional)

```
future_preds <- 10^(future_preds_log)
```

4. Plot predictions

```
plot(uspop,
  main = "Forecast",
  ylab = "Dependent variable",
  xlab = "Time, yrs"
)

lines(future_time, future_preds, col = "red", lwd = 2)
points(future_time, future_preds, col = "red", pch = 19)
```

Improve model and prediction

```
# Residuals from trend model
residuals_trend <- resid(ts_data_lm)

# Check autocorrelation
acf(residuals_trend)

# Fit ARIMA model to residuals
residuals_arima <- auto.arima(residuals_trend)
summary(residuals_arima)

# Forecast future trend
future_trend <- predict(ts_data_lm, newdata = data.frame(time.ts_data = future_time))

# Forecast residuals
future_residuals <- forecast(residuals_arima, h = 10)$mean

# Combine trend + residuals
future_log_combined <- future_trend + future_residuals
future_combined <- 10^(future_log_combined)

# Plot combined forecast
plot(ts_data,
  main = "Improved Forecast",
  ylab = "Dependent variable",
  xlab = "Time, yrs"
)
lines(future_time, future_combined, col = "green", lwd = 2)
points(future_time, future_combined, col = "green", pch = 19)
```

Ordinary differential equations modeling

Concept overview

- Purpose: to describe **how a quantity changes over time**.
- General form: $\frac{dy}{dt} = f(y, t)$, which means: the rate of change of variable y over time t can be expressed with a function.

Tumor growth

- N: tumor size
- $r_{Logistic}$: intrinsic growth rate
- $r_{Gompertz}$: growth rate constant
- $K_{Logistic}$: carrying capacity
- $K_{Gompertz}$: asymptotic limit (interpreted as maximum tumor size)
- α: treatment effect

*r and K interpreted differently, but same use & value in equations.

Model	Equation	Characteristics
Exponential Model	$\frac{dN}{dt} = rN$	- Rapid, unchecked growth - Not realistic in the long term
Logistic Model	$\frac{dN}{dt} = rN\left(1 - \frac{N}{K}\right)$	- Growth slows as it nears carrying capacity K - More realistic
Logistic with treatment *only when treatment is mentioned*	$\frac{dN}{dt} = rN\left(1 - \frac{N}{K}\right) - \alpha N$	- Adds treatment effect α - Simulates therapy scenarios
Gompertz Model	$\frac{dN}{dt} = rN\ln\left(\frac{K}{N}\right)$	- Slower growth over time - Asymmetric S-curve - **Default tumor fit**

```r
# Construct Logistic model
logistic_model <- function(t, state, parameters) {
  with(as.list(c(state, parameters)), {
    dN <- r * N * (1 - N / K)
    return(list(c(dN)))
  })
}

# Construct Gompertz model
gompertz_model <- function(t, state, parameters) {
  with(as.list(c(state, parameters)), {
    dN <- r * N * log(K / N)
    return(list(c(dN)))
  })
}

# Common parameters
parameters <- c(r = 0.05, K = 1000)
state <- c(N = 10)
times <- seq(0, 200, by = 1)

# Solve both models
out_logistic <- ode(y = state, times = times, func = logistic_model,
                    parms = parameters) %>% as.data.frame()
out_gompertz <- ode(y = state, times = times, func = gompertz_model,
                    parms = parameters) %>% as.data.frame()

# Differentiate them
out_logistic$model <- "Logistic"
out_gompertz$model <- "Gompertz"

# Combine
out_combined <- rbind(out_logistic, out_gompertz)

# Plot
```

```r
ggplot(out_combined, aes(x = time, y = N, color = model)) +
  geom_line(linewidth = 1.2) +
  labs(title = "Comparison of Tumour Growth Models", x = "Time", y = "Tumour Size") +
  theme_minimal()
```

Interpretation:

- The Gompertz model reaches its inflection point earlier, and it predicts a slower, more gradual approach to the carrying capacity than the Logistic model.
- The Gompertz model is better becauseit accounts for the decelerating growth seen in tumors as they experience environmental constraints like lack of nutrients, hypoxia, or immune response. Logistic growth is symmetric and may overestimate growth in later stages.

Bacteria growth

- B: #bacteria
- r: replication rate
- K: carrying capacity

Model	Equation	Characteristics
Logistic model	$\frac{dB}{dt} = rB\left(1 - \frac{B}{K}\right)$	- Used in lab environments - Growth limited by nutrients, space, etc.

```r
# Construct growth model for later use
# t&state: variable
# parameters: parameter
bacteria_model <- function(t, state, parameters) {
  with(as.list(c(state, parameters)), { # necessary
    dB <- r * B * (1 - B / K)
    return(list(c(dB))) # necessary
  })
}

# Set variables and parameters' values
parameters <- c(r = 0.4, K = 500)
state <- c(B = 5)
times <- seq(0, 50, by = 0.5)

# Call ODE to embed the model
out <- ode(y = state, times = times, func = bacteria_model, parms = parameters)
out <- as.data.frame(out)

# Plot
ggplot(out, aes(x = time, y = B)) +
  geom_line(color = "cyan", linewidth = 1) +
  labs(title = "Bacterial Growth", x = "Time", y = "Bacterial Population") +
  theme_minimal()
```

Full SIR model - non Markovian, continuous

- β: transmission rate
- γ: recovery rate
- $R_0 = \frac{\beta}{\gamma}$: the basic reproduction number ($R_0 > 1$: outbreak spreads, $R_0 < 1$: outbreak dies out)

Component	Equation	Interpretation
#%Susceptible (S)	$\frac{dS}{dt} = -\beta SI$	- Individuals who CAN catch the disease (but still healthy)
#%Infected (I)	$\frac{dI}{dt} = \beta SI - \gamma I$	- Individuals who HAVE the disease and thus have the ability to transmit it - Calculated as new infections minus recoveries
#%Recovered (R)	$\frac{dR}{dt} = \gamma I$	- Individuals who have recovered (OR DIED) and are NO LONGER infectious

```r
# Construct SIR model for later use
sir_model <- function(t, state, parameters) {
  with(as.list(c(state, parameters)), {
    dS <- -beta * S * I
    dI <- beta * S * I - gamma * I
    dR <- gamma * I
    return(list(c(dS, dI, dR)))
  })
}

# Set variables and parameters' values
parameters <- c(beta = 0.3, gamma = 0.1)
state <- c(S = 0.99, I = 0.01, R = 0.0)
times <- seq(0, 160, by = 1)

# Call ODE to embed the model
out <- ode(y = state, times = times, func = sir_model, parms = parameters)
out <- as.data.frame(out)
out_long <- pivot_longer(out, cols = -time, names_to = "Compartment", values_to = "Proportion")

# Plot the results
ggplot(out_long, aes(x = time, y = Proportion, color = Compartment)) +
  geom_line(linewidth = 1) +
  labs(title = "SIR Model Simulation", x = "Time", y = "Proportion of Population") +
  theme_minimal()
```

Interpretations:

- The number of infected individuals peak when the rate of new infections equals the rate of recoveries, that is, when $\frac{dI}{dt} = 0$.
- Increasing γ flattens the curve, reduce peak, make the end faster.

Matrix-based modeling

Concept overview

- **Matrix-based modeling**: use matrix/-cies to represent relationships between variables
  - **Transition matrix (P)** : express the relationship between variables, e.g., $P = \begin{pmatrix} 0.8 & 0.2 \\ 0.1 & 0.9 \end{pmatrix}$ (columns represent transitions *from* states A and B, rows represent transitions *to* states A and B)
  - **Initial state (S_0)**: initial value of all variables in the system, e.g., $S_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
  - **After/on the point of n transition (S_n)**: value of all variables in the system after n transitions, e.g., $S_1 = \begin{pmatrix} 0.75 \\ 0.25 \end{pmatrix}$
  - **Transition matrix equation**: equation expressed by matrix-vector multiplication as an alternative for "tree diagram thinking". $S_1 = P \cdot S_0 \rightarrow S_2 = P \cdot S_1 = P^2 \cdot S_0 \rightarrow$ *Generalization*: $S_n = P^n \cdot S_0 \quad$ for $n \geq 0$
- **Markov chains**: process moves between states with probabilities and the next sole depends on current
- **Difference equations**
  - Note: *difference equations* and *differential equations* are different ways to model a (potentially the same) question, with different ways to approach *time*

	Difference Equation	**Differential Equation**
Purpose	Model the difference between consecutive steps	Quantify instantaneous change rate
Expression	$x_{t+1} = Ax_t \quad$ or $\quad x_{t+1} - x_t = f(x_t)$	$\frac{dx}{dt} = f(x(t))$
Time	Discrete time steps (t = 0, 1, 2, 3, ...)	Continuous time flow
Application	Population growth, disease spread, blood flow...	See above

- **Division of difference equation(DE)**

	Description	**Equation & example**
Simple DE	Single variable	• $P_{t+1} = A \cdot P_t \quad$ where $\quad A = (1 + \gamma)$ • P_t is the population size • γ is the growth rate
System of DE	Multiple interacting variables forms a system	• $\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = A \cdot \begin{bmatrix} x_t \\ y_t \end{bmatrix}$ • A is a matrix. It captures the interactions • Two compartment DE system: – $x_{t+1} = 0.8x_t + 0.2y_t$ – $y_{t+1} = 0.1x_t + 0.9y_t$ – initial condition: $x_0 = 100, \quad y_0 = 50$

SIR early-stage model

Early epidemic phase is often modeled with linearized SIR model, since $I(t)$ is small; $S(t) \approx S_0$ is constant; and $R(t) \approx 0$. Thus, $\frac{dI}{dt} = (\beta S_0 - \gamma)I$ since $\frac{dI}{dt} = \beta S(t)I - \gamma I$ and $(\beta S_0 - \gamma)$ is a constant, same role as r.

Single population SIR:

- $\frac{d}{dt}[I] = (\beta S_0 - \gamma)[I]$ as it is

```r
# Parameters
beta <- 0.9
gamma <- 0.2
S0 <- 0.90
r <- beta * S0 - gamma

# Initial infected proportion
I0 <- 0.10

# Time settings
time_steps <- 10 # total time course
infected <- numeric(time_steps)
infected[1] <- I0 # initialize a vector

# linearised infection growth
for (t in 2:time_steps) {
  infected[t] <- (1 + r) * infected[t-1] # use the formula
```

```
}

# Susceptible stays constant, no recovery yet
# So same throughout the time course
susceptible <- rep(S0, time_steps)
recovered <- rep(0, time_steps)

# Plot
plot(1:time_steps, infected, type = "b", col = "red",
     xlab = "Time step", ylab = "Proportion",
     ylim = c(0, 1),
     main = "linearised SIR: Infected over Time")
lines(1:time_steps, susceptible, type = "l", col = "blue")
lines(1:time_steps, recovered, type = "l", col = "green")
legend("topright", legend = c("Infected (I)", "Susceptible (S)", "Recovered (R)"),
       col = c("red", "blue", "green"), lty = 1, bty = "n")
```

Multiple population SIR:

- Variable I is not singular; parameter β and γ is not singular
  - $\vec{I} = \begin{bmatrix} I_A \\ I_B \end{bmatrix}$
  - Transmission rate β table:
  - Recovery rate γ: γ_A, γ_B

From / To	Infecting A	Infecting B
A	β_{AA}	β_{AB}
B	β_{BA}	β_{BB}

- Initial state is also not singular, so the constant $\beta S_0 - \gamma$ is not singular either, thus it should be expressed as a matrix: $A = \begin{bmatrix} \beta_{AA}S_A - \gamma_A & \beta_{BA}S_B \\ \beta_{AB}S_A & \beta_{BB}S_B - \gamma_B \end{bmatrix}$
- Complete equation: $\frac{d}{dt}\begin{bmatrix} I_A \\ I_B \end{bmatrix} = A \cdot \begin{bmatrix} I_A \\ I_B \end{bmatrix}$
- $A\vec{v} = \lambda\vec{v}$
  - Eigenvalue λ: the sign and size of the *dominant (largest in magnitude) eigenvalue* informs to what extent the epidemic is growing or fading
  - If $|\lambda_{\max}| < 1$: infection dies out
  - If $|\lambda_{\max}| > 1$: infection spreads
  - If $|\lambda_{\max}| = 1$: infection is in equilibrium
  - Eigenvector $\vec{v}$: shows the mode of disease spread across groups
    - If A < B: B is the main infected group, dominant the spread direction
    - If A > B: A is the main infected group, dominant the spread direction

Summary:

	Single-variable model	Multi-group model
Expression	$\frac{dI}{dt} = rI$	$\frac{d\vec{I}}{dt} = A\vec{I}$
Solution	$I(t) = I_0 e^{rt}$	$\vec{I}(t) = \sum c_i e^{\lambda_i t}\vec{v}_i$
Growth?	$r > 0$	$\lambda_{\max} > 0$
Spreading direction	No directionality	Depends on eigenvector $\vec{v}$

Full SIR - Markovian, discrete

- Expressed in Markovian form: $x_{t+1} = P^T x_t \mid x_t = (P^T)^t x_0$ or $x_{t+1} = Px_t \mid x_t = P^t x_0$
  - When $P * x = x$ or $P^T * x = x$
    - Most people recover, no new infections
    - The system is in equilibrium, disease is ended

Interpretation of the matrix:

$$\begin{bmatrix} 0.80 & 0.20 & 0.00 \\ 0.00 & 0.70 & 0.30 \\ 0.00 & 0.00 & 1.00 \end{bmatrix}$$

- Each row represent probabilities of transitioning between states, where row 1 represents transition starts with the susceptible (1_{st} column), to the infected (2_{nd} column) and directly jump to the recovered (3_{rd} column).

- The matrix is not meaningful to read as column.
- The sum of each row is 1 because we have to ensure conservation of probability.

Construct matrix

```r
P <- matrix(c(
  0.80, 0.20, 0.00,
  0.00, 0.70, 0.30,
  0.00, 0.00, 1.00
), nrow = 3, byrow = TRUE)

P <- t(P) # very important!!!
```

State distribution after 1 time step

```r
# Initial state vector: conditions at time step 0
state_vec <- c(0.99, 0.01, 0.00)  # The sum is 100% of the population
state_after_one <- P %*% state_vec
print(state_after_one)
```

The proportion of infected individuals [increases slightly] after one time step.

Steady-state distribution

```r
# Initial state vector
state_vec <- c(0.99, 0.01, 0.00)
steps <- 150
states_vec_over_time <- matrix(0, nrow = steps, ncol = 3)
states_vec_over_time[1, ] <- state_vec

for (t in 2:steps) {
  states_vec_over_time[t, ] <- P %*% states_vec_over_time[t-1, ]
}

# Plot
matplot(1:steps, states_vec_over_time, type = "l", lty = 1, col = c("blue", "red", "green"),
        xlab = "Time Step", ylab = "Proportion",
        main = "Markov Chain Infection Modeling")
legend("right", legend = c("Susceptible", "Infected", "Recovered"),
       col = c("blue", "red", "green"), lty = 1, bty = "n")

# Calculate state at time step 20 using X0 %*% P^20
state_at_10 <- state_vec %*% (P %^% 10)
```

Steady-state largest eigenvalues and eigenvectors

$$\vec{(X)} = P^t \cdot \vec{(X)}$$

```r
res <- eigen(P)
res$values
res$vectors[, 1] # first eigenvector
```

So the steady state distribution is given by the first eigenvector, which corresponds to the largest eigenvalue (1 in this case). Actually in this scenario, as long as the R in SIR equals 1, the steady state distribution will always be $(0, 0, 1)$, regardless of the transition probabilities in the matrix.

Interpretation:

- Changing the transition probabilities in the matrix affects
  - How quickly individuals move from Susceptible → Infected → Recovered
  - The steady-state distribution
- But the proportion of infected individuals eventually stabilizes, indicating the steady-state condition.

Vessel blood flow model

- **Braching point**: several flow in & out (driven by *pressure difference*) equation. Poiseuille's law (analogy to Ohm's law)
  - $Q = \frac{\Delta p}{R}$
    - Q: flow
    - Δp: pressure difference $= p_{in} - p_{out}$
    - R: resistance
  - e.g.,
    - $p_1 - p_2 = R_1 Q_1$
    - $p_2 - p_3 = R_2 Q_2$
    - $p_2 - p_4 = R_3 Q_3$
- **Vessel**: one mass conservation equation
  - $Q_{Ins} = Q_{Outs}$
  - e.g., $Q_1 = Q_2 + Q_3$
- **Taken together**:
  - $p_2 + R_1 Q_1 = p_1$
  - $p_2 - R_2 Q_2 = p_3$
  - $p_2 - R_3 Q_3 = p_4$
  - $Q_1 - Q_2 - Q_3 = 0$
- **Matrix form**: $A\vec{x} = \vec{b}$
  - A: matrix of coefficients; $\vec{x}$: vector of unknowns; $\vec{b}$: vector of constants
    - $A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ -R_1 & 1 & -1 & 0 \\ 0 & -R_2 & 1 & 0 \\ 0 & -R_3 & 0 & 1 \end{pmatrix}$
    - $\vec{x} = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix}$
    - $\vec{b} = \begin{pmatrix} Q_1 \\ Q_2 \\ Q_3 \\ 0 \end{pmatrix}$
  - **Solve**: $\vec{x} = A^{-1}\vec{b}$
- **Flow**: mass conservation equation
  - $Q = \frac{dV}{dt}$
    - Q: flow
    - V: volume
    - t: time

Code implementation:

$\vec{x} = A^{-1}\vec{b}$ now is $A \cdot \begin{bmatrix} p_2 \\ Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} = \begin{bmatrix} p_1 \\ p_3 \\ p_4 \\ 0 \end{bmatrix}$

```r
#=== Step 1: Define pressure and resistance (known parameters) ===#
# Known pressures (in mmHg)
p1 <- 10  # Node 1 (inlet)
p3 <- 4   # Node 3 (outlet)
p4 <- 4.5   # Node 4 (outlet)
# p2 unknown!!!

# Resistances (arbitrary units)
R1 <- 2    # 1 to 2
R2 <- 4    # 2 to 3
R3 <- 3    # 2 to 4

#=== Step 2: Set up equation system ===#
# Coefficient matrix A
A <- matrix(
  c(
    1, R1,  0,  0,    # (p1 - p2) = R1 * Q1
    1,  0, -R2, 0,    # (p2 - p3) = R2 * Q2
    1,  0,  0, -R3,   # (p2 - p4) = R3 * Q3
    0,  1, -1, -1     #  Q1 = Q2 + Q3
```

```
    ), nrow=4, byrow=TRUE)

# Known parameters grouped into vector b
b <- matrix(c(
  p1,
  p3,
  p4,
  0
), nrow=4, byrow=TRUE)

# Solve the linear system for [p2, Q1, Q2, Q3]
solution <- solve(A, b)
solution

# Extract values
p2 <- solution[1]
Q1 <- solution[2]
Q2 <- solution[3]
Q3 <- solution[4]

list(
  p2 = p2,
  Q1 = Q1,
  Q2 = Q2,
  Q3 = Q3
)

#=== Step 3: Visualize the flows ===#
barplot(c(Q1, Q2, Q3),
        names.arg = c("Q1 (1->2)", "Q2 (2->3)", "Q3 (2->4)"),
        main = "Flow Rates in 4-Node Bifurcation",
        ylab = "Flow Rate",
        col = "blue")
```

What about adding one more branch?

```
# New knowns
p5 <- 5
R4 <- 5

# New Coefficient Matrix A_ext (5 equations, 5 unknowns: p2, Q1, Q2, Q3, Q4)
A_ext <- matrix(c(
    1, R1,  0,  0,  0,     # (p1 - p2) = R1 * Q1
    1,  0, -R2, 0,  0,     # (p2 - p3) = R2 * Q2
    1,  0,  0, -R3, 0,     # (p2 - p4) = R3 * Q3
    1,  0,  0,  0, -R4,    # (p2 - p5) = R4 * Q4
    0,  1, -1, -1, -1      # Q1 = Q2 + Q3 + Q4
), nrow=5, byrow=TRUE)

# Right-hand side vector b_ext
b_ext <- matrix(c(
  p1,
  p3,
  p4,
  p5,
  0
), nrow=5, byrow=TRUE)

# Solve for [p2, Q1, Q2, Q3, Q4]
solution_ext <- solve(A_ext, b_ext)
solution_ext

p2_ext <- solution_ext[1]
Q1_ext <- solution_ext[2]
```

```r
Q2_ext <- solution_ext[3]
Q3_ext <- solution_ext[4]
Q4_ext <- solution_ext[5]

list(
  p2 = p2_ext,
  Q1 = Q1_ext,
  Q2 = Q2_ext,
  Q3 = Q3_ext,
  Q4 = Q4_ext
)

barplot(c(Q1_ext, Q2_ext, Q3_ext, Q4_ext),
        names.arg = c("Q1 (1->2)", "Q2 (2->3)", "Q3 (2->4)", "Q4 (2->5)"),
        main = "Flow Rates in 5-Node Extended Network",
        ylab = "Flow Rate",
        col = "green")
```

Conditional probability

Conditional probability is for two event-analysis.

Basics

Probability and Odds

- Probability: the chance that A happens
- Odds: the ratio of $\frac{\text{The chance that A happens}}{\text{The chance that A not happens}}$

Code implementation

Principles:

Function type	Meaning	Purpose
`r...`	Random	Generate a series of random numbers
`d...`	Density (value)	Probability or density at a *specific* value
`p...`	Cumulative	Cumulative probability up to a value
`q...`	Quantile	Value required to a defined cumulative probability

Binomial distribution and Normal distribution:

Function	Explanation
`rbinom(n, size, prob)`	Generate `n` random values from a binomial distribution with `size` trials and success `prob`.
`dbinom(x, size, prob)`	Calculate the probability of exactly `x` successes in `size` trials with success `prob`.
`pbinom(q, size, prob)`	Compute the cumulative probability of `q` successes in `size` trials with success probability `prob`.
`qbinom(p, size, prob)`	Find the number of successes `x` such that the cumulative probability is at least `p` (i.e., `P(X x) p`).
`rnorm(n, mean, sd)`	Generate `n` random numbers from a normal distribution with given `mean` and standard deviation `sd`.
`dnorm(x, mean, sd)`	Compute the height of the normal distribution (density) at value `x` with given `mean` and `sd`.
`pnorm(q, mean, sd)`	Calculate the cumulative probability that a normally distributed value is `q`, with given `mean` and `sd`.
`qnorm(p, mean, sd)`	Find the value `x` such that the cumulative probability `P(X x)` is equal to `p`, given the `mean` and `sd`.

Logic notations

Symbol	Description	Truth Condition
$\neg A$	"not A"	True if A is false
$A \wedge B$ or $A\&B$	"A and B"	True if both A and B are true
$A \vee B$	"A or B"	True if at least one of A or B is true
$A \to B$	"If A then B"	True unless A is true and B is false
$A \leftrightarrow B$	"If and only if A then B"	True if A and B have the same truth value

Probability notations

- Conditional probability: $P(A \mid B)$
- Joint probability: $P(A \cap B)$ or $P(A\&B)$

Probability calculation

	A−	**A+**	**P(B)**
B−	A−B−	A+B−	Total B−
B+	A−B+	A+B+	Total B+
P(A)	Total A−	Total A+	**100%**

- $P(A \cap B) = P(B \cap A)$
  - $P(A \cap B) = P(A \mid B) \cdot P(B)$
  - $P(B \cap A) = P(B \mid A) \cdot P(A)$
- $P(A \mid B) \cdot P(B) = P(B \mid A) \cdot P(A)$
- $P(B) = \sum P(B_i \mid A_i) \cdot P(A_i)$
- Joint probability
  - Calculated using *conditional probability*: $P(A \cap B) = P(A) \cdot P(B \mid A)$
  - Calculated using *individual probability*: $P(A \cap B) = P(A) \cdot P(B)$

Note

1. If A and B are independent, then $P(B \mid A) = P(B \mid \neg A) = P(B)$
2. $P(A \mid B) \neq P(B \mid A)$

Tools

- **Venn plot = contingency table:** for depicting event probability landscape
- **Probability trees:** for branching problems

Conditional probability

- $P(A \cap B \mid B) = \frac{P(A \cap B)}{P(B)}$
- $P(A \mid B) = \frac{P(A \cap B)}{P(B)}$
- $P(\text{What we want to know}) = \frac{P(\text{What we also know})}{P(\text{What we know})}$

Interpretation: the probability of an event to happen, scaled by the knowledge we already have.

The A and B can be calculated using: 1. Raw counts 2. Original probability

Bayesianism

Bayes' Theorem

Expression

Since $A \cap B = B \cap A$, so if conditional probability is used twice, it can be deducted that

- $P(A \cap B \mid B) = \frac{P(A \cap B|A) * P(A)}{P(B)}$
- $P(A \cap B \mid A) = \frac{P(A \cap B|B) * P(B)}{P(A)}$

Interpretation

Bayes' Theorem is useful because

1. It tell us that the conditional probability given that we know *one thing* about an event can be derived from knowing *the other thing* about the event. In the end there is only one event. And sometimes $A \cap B \mid A$ is easier to calculate than $A \cap B \mid B$ and *vice versa*.
2. Always, the other component of the formula does not involve conditional probability calculation or very easily calculated or is given information.
3. It can be applied to the temporal dimension of an event, that is it unveils the relationship between prior belief and posterior belief.
4. The conditional probability given that we know one thing about an event can be derived from knowing the other thing about the event.

Bayesianism vs. Frequentism

	Frequentism	Beyesianism
Probability is	Long-run calculation	A degree of belief (how you see something based on the fact in hand)
Expression	$P(X) = \frac{X}{X+X^c}$	$P(X) \propto$ Likelihood $\cdot$ Prior knowledge
Want to add new data	Start a whole new long-run	By Bayesian updating

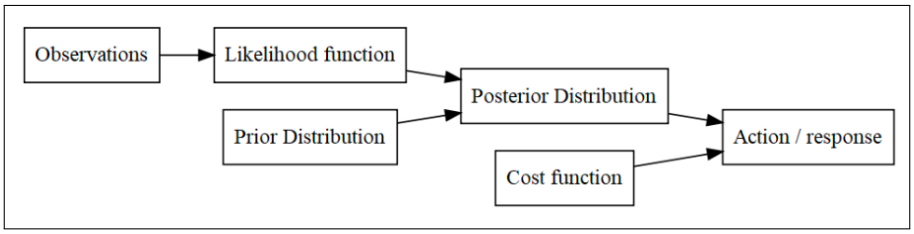
Bayesian *logic* components

Bayesianism inference is not a probability type that parallel with, for example, Bernoulli. It is a *way of thinking, an approach of seeing the world.*

- **Prior** $P(H)$: the *hypothesis*
- **Likelihood** $P(data \mid H)$: the *truth/fact*
- **Posterior** $P(H \mid data)$: the *compromise between the hypothesis and the truth/fact (weighted multiplied average of prior and likelihood)*, the more we do the experiment, the more we know about the truth, the more impact likelihood has on the posterior
- **Marginal likelihood** $P(data)$: the *total probability of the data.* This is a θ-independent constant so it does not influence the posterior shape (peak position & width).

Some use θ for prior, as a representation of prior's parameter.

If prior is not set for a new question, it is acceptable to formulate your own prior even if it is not based on the common sense; however, you should note that how you formulate prior will definitely affect the posterior accuracy.



Bayesian factor

- Purpose: to quantify and compare how strongly the data support two (and only two) competing hypotheses. So if there are too many hypotheses to compare, it is suggested to preclude some hypotheses that are not likely to be true.
- Definition: $\frac{\text{Posterior Odds}}{\text{Prior Odds}}$
- Note: not necessarily have to use. Not a standard procedure. Only used if it comes to comparing the effectiveness of models.
- Expression:

$$\text{Ratio of priors} = \text{Prior odds} = \frac{P(H_1)}{P(H_2)}, \text{where } P(H_1) + P(H_2) + ... + P(H_i) = 1$$

$$\text{Ratio of posteriors} = \text{Posteriors odds} = \frac{P(H_1 \mid data)}{P(H_2 \mid data)}, \text{where } P(H_1 \mid data) + ... + P(H_i \mid data) = 1$$

$$\frac{\text{Likelihood under } H_1}{\text{Likelihood under } H_2} = BF(H_1 : H_2) = \frac{P(data \mid H_2)}{P(data \mid H_1)} = \frac{\frac{P(H_1|data)}{P(H_2|data)}}{\frac{P(H_1)}{P(H_2)}}$$

- Interpretation: BF = 10 suggest 10 times more likely to observe the data under H_1 than under H_2

A dice game

Your friend got 7 sixes in 20 rolls of a dice. Is this dice fair? What number of six do you think is most likely to be?

For this question, Bayesian is the way of thinking while Binomial/Bernoulli is the way of calculating.

Hypotheses:

- H_0: the die is fair (precluded)
- H_1: the die has 1 six
- H_2: the die has 2 six
- ...
- H_6: the die has 6 six (precluded)

```r
# 1. Prior can be set as a uniform distribution or whatever you want
P_hypotheses <- rep(0.2, 5)

# 2. Likelihood for each hypothesis: binomial formula
Probabilities_sixes <- c(1 / 6, 2 / 6, 3 / 6, 4 / 6, 5 / 6)
P_givenData_Expectation <-
  dbinom(x = 7, size = 20, prob = Probabilities_sixes)
dice <- cbind(Probabilities_sixes, P_givenData_Expectation)

# 3. Marginal probability
P_givenData <- sum(P_hypotheses * dice[, 2])

# 4. Posterior (for hypothesis 1)
(P_givenData_Expectation[1] * P_hypotheses[1]) / P_givenData
```

R as a calculator

Basic arithmetic operations

```r
# Addition, subtraction, multiplication, division
6 * 7       # 42
20 / 5      # 4

# Exponentiation and modulo
2 ^ 3       # 8
10 %% 3     # 1    (remainder)
10 %/% 3    # 3    (integer division)

# Negatives and absolute value
-5 * 2      # -10
abs(-7.3)   #  7.3
```

Mathematical functions

```r
# Square root, exponentials, logarithms
sqrt(16)          # 4
exp(1)            # e^1 = 2.718282
log(10)           # natural log ln(10) = 2.302585
log10(1000)       # log base 10 of 1000 = 3
log(x, base = 2)  # log2(x)

# Rounding functions
round(3.14159, 2) # 3.14
floor(3.9)        # 3
ceiling(3.1)      # 4
trunc(3.9)        # 3  (drops fractional part)
```

Vectorized operations and vector functions

```r
# Define two vectors
x <- c(1, 2, 3, 4, 5)
y <- c(10, 20, 30, 40, 50)

# Common vector functions
sum(x)        # 15
prod(x)       # 120
mean(x)       # 3
median(x)     # 3
var(x)        # 2.5
sd(x)         # sqrt(2.5) = 1.5811
range(x)      # 1, 5
min(x)        # 1
max(x)        # 5
```

Matrix operations

```r
# Create a 3×3 matrix (filled by column)
m <- matrix(1:9, nrow = 3, ncol = 3)
#      [,1] [,2] [,3]
# [1,]    1    4    7
# [2,]    2    5    8
# [3,]    3    6    9

# Indexing
m[1, 2]     # 4
```

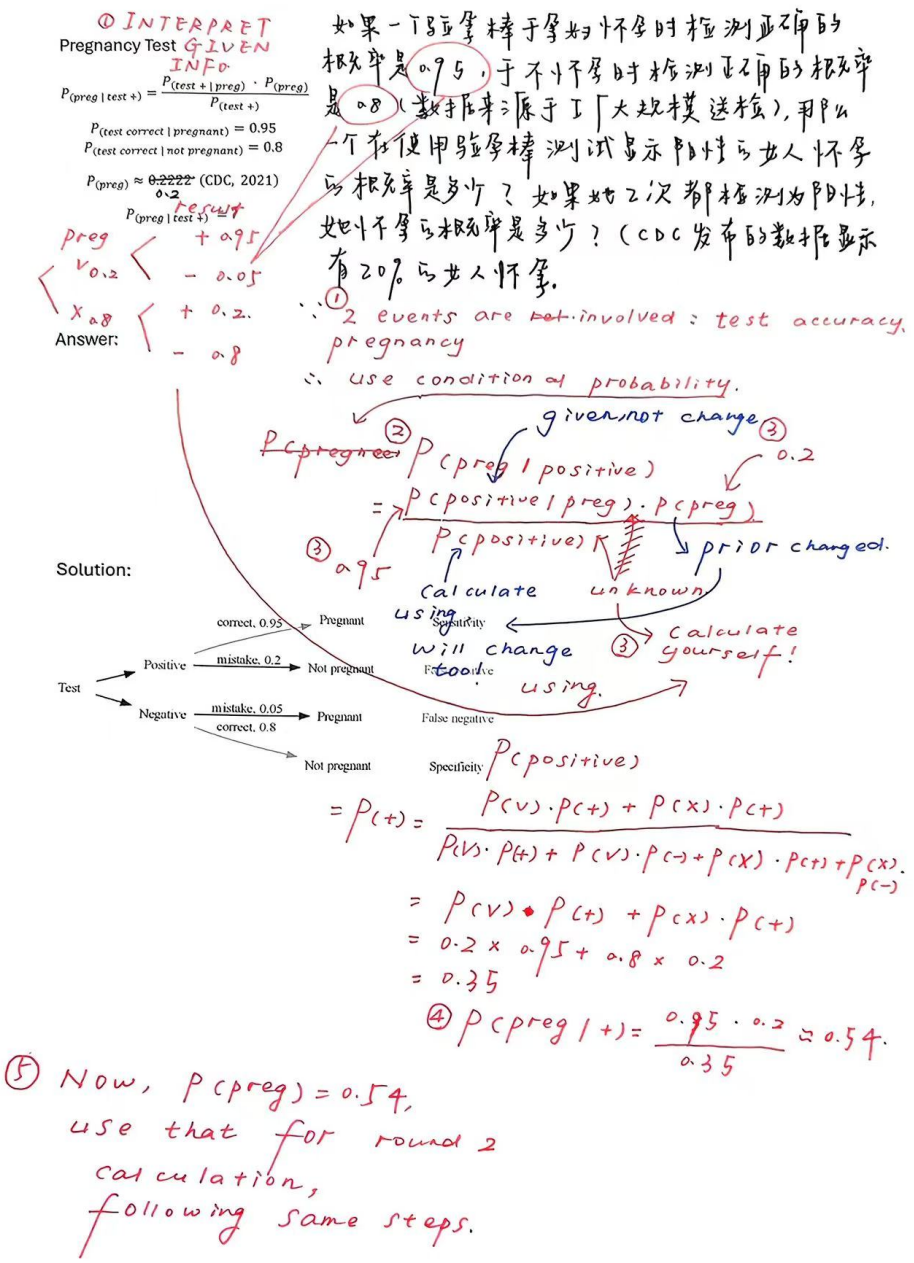
```r
m[, 3]       # 7, 8, 9 (entire 3rd column)
m[2, ]       # 2, 5, 8 (entire 2nd row)

# Common matrix functions
t(m)         # transpose
diag(m)      # diagonal elements: 1, 5, 9
det(m)       # determinant
solve(m)     # inverse (if invertible)
eigen(m)     # eigenvalues and eigenvectors

# Matrix multiplication vs. element-wise
A <- matrix(c(1, 2, 3, 4), nrow = 2)
B <- matrix(c(5, 6, 7, 8), nrow = 2)
A %*% B      # matrix multiplication
A * B        # element-wise multiplication

# Row/column summaries
rowSums(m)  # 12, 15, 18
colSums(m)  # 6, 15, 24
rowMeans(m) # 4, 5, 6
colMeans(m) # 2, 5, 8
```

Solving equations

```r
# Solve a single equation, e.g., x^2 - 3x + 2 = 0
f <- function(x) x^2 - 3*x + 2
uniroot(f, lower = 0, upper = 3)  # finds a root at x = 1 or x = 2
```

Probability practice problems

1 IVF test



Q: What can you say about this test kit? Is it good/specific/sensitive? Support your claims with odds and odds ratios.

- **Specific** = true negative = when test shows negative, the women is actually not pregnant
- **Sensitive** = true positive = when test shows positive, the women is actually pregnant
- **Good**: a overall assessment of specificity and sensitivity

Actually specificity and sensitivity cannot be assessed in this way, but in this case, we can just take it.

- **Is positive test result reliable**: $\text{Odds}_{(\text{pregnant vs non-pregnant AND test}^+)} = \frac{0.211}{0.1556} \approx 1.356$
- **Is negative test result reliable**: $\text{Odds}_{(\text{non-pregnant vs pregnant AND test}^-)} = \frac{0.6222}{0.0111} \approx 56.054$
- **What test result is more reliable**: Odds Ratio $= \frac{1.356}{56.054} \approx 0.024$

In conclusion:

- The chances of being correct are too low for a positive test result, that is there is too much false positive.
- The chances of being correct are too high for a negative test result (more being specific than being sensitive).
- So the overall quality is bad.

2 Varicocele surgery

Varicocele causes infertility in men. The conception rate of patients within 1 year after surgery is around 24.7%. A new procedure emerged. 18 out of 53 patients reported conception within 1 year after that new surgery.

How likely is it that this novel procedure is better than the traditional surgery?

What would happen if the sample size is increased to 80 patients with the same proportion of the positive outcome?

```r
# --------------------------------
# STEP 1: Frequentism probability
# --------------------------------

# Probability of  18 conceptions out of 53 if true rate is 0.247
pbinom(q = 17, size = 53, prob = 0.247, lower.tail = FALSE)
# [1] 0.08326483

# Larger sample case (80 patients with same proportion)
bigger_sample_success <- round((18 / 53) * 80)
pbinom(
  q = bigger_sample_success - 1,
  size = 80,
  prob = 0.247,
  lower.tail = FALSE
)
# [1] 0.04360466



# --------------------------------
# STEP 2: Bayesian Inference Setup
# --------------------------------

# Hypothesis space
model <- seq(0, 0.5, by = 0.025)  # probabilities of success to test

# Prior belief: using Beta(7, 21) as described in the slide (prior mean ~0.25)
prior <- dbeta(model, shape1 = 7, shape2 = 21)

# Likelihood of observing 18 successes out of 53
likelihood <- dbinom(x = 18, size = 53, prob = model)

# Posterior (unnormalized)
posterior_unnorm <- likelihood * prior

# Normalized posterior
posterior <- posterior_unnorm / sum(posterior_unnorm)

# --------------------------------
# STEP 3: Plotting the results
# --------------------------------

# Plot prior
plot(model, prior, type = "h", col = "gray30", lwd = 2, ylim = c(0, max(c(prior, likelihood, posterior))),
     main = "Bayesian Inference: Varicocele Case",
     ylab = "Density / Likelihood",
     xlab = "Probability of Success")
# Add likelihood
lines(model, likelihood, type = "h", col = "red", lwd = 2)
# Add posterior
lines(model, posterior, type = "h", col = "blue", lwd = 2)
legend("topright",
       legend = c("Prior", "Likelihood", "Posterior"),
       col = c("gray30", "red", "blue"),
       lwd = 2)

# --------------------------------
```

```r
# STEP 4: Comparing Hypotheses with Bayes Factor
# ------------------------------

# Hypotheses from the slide: H0 = 0.25, H1 = 0.35
H0 <- 0.25
H1 <- 0.35

P_data_H0 <- dbinom(18, size = 53, prob = H0)
P_data_H1 <- dbinom(18, size = 53, prob = H1)

# Bayes Factor BF[H1:H0] (Evidence in favor of H1 over H0)
BF_H1_H0 <- P_data_H1 / P_data_H0
BF_H1_H0  # ~0.35

# Reciprocal if needed (BF[H0:H1])
BF_H0_H1 <- 1 / BF_H1_H0
BF_H0_H1  # ~2.86
```

Critical thinking & future directions

A general rule for the question of "What would you suggest doing next?" is to think like a biologist. Think about the biological principles, not only statistics.

- Use **stratified sampling** to ensure that the sample includes **diverse** groups to account for potential **confounding variables** (e.g., age, gender, genetic background).
- Include different population subsets: for human-based studies, ensure that you have subgroups that represent the diversity of the population, such as by age, gender, or health condition, to improve the generalizability of the findings.
- Refine the control group to ensure it is as **similar** as possible to the experimental group, except for the tested key variable.
- Consider adding **more control groups** (e.g., positive control, negative control) to help validate the experiment.
- Use **randomization**. Randomly assign subjects to the experimental or control group to reduce selection bias.
- Use **double blinding**. Reduce bias in data collection and analysis, where the experimenters are unaware of the group assignments.
- **Longitudinal study design**: the study is cross-sectional, consider a longitudinal design to track changes over time in the same subjects. This can provide insights into the dynamics of the effect and its persistence.
- **Dose-response relationships**: Include multiple doses/concentrations of a treatment to clarify how the magnitude of the effect varies with doses/concentrations. This can provide insights into the pharmaco-dynamics or effects of a compound.
- The experiment is unable to address the long-term effects. So **follow-up experiments** needs to be designed to address these gaps.
- The experiment is unable to offer specific **mechanistic insights**, such as what is the underlying molecular signalling pathway responsible for it?
  - Consider using techniques such as Western Blotting, immunohistochemistry to explore underpinning pathways.
  - Other Options:
    - Pathway-specific inhibitors
    - RNA interference (RNAi)
    - CRISPR/Cas9 gene knockouts: manipulate genes or proteins.
    - Genomics, transcriptomics, or proteomics: identify changes at a molecular level (scRNA-seq find DEGs).
- Consider using a **different model**. Testing in other organisms or systems (e.g., cell culture vs. animal models) to broaden the findings.
- **Use other tests**
- The ethical issues when it comes to **choosing power**:
  - Too small a sample size with an underpowered study:
    - Waste resources
    - Can't reject H_0
    - Misleading conclusions if results are nonsignificant
    - Unethical if the conclusions lead to inferior treatment clinically
  - Too big a sample size with an overpowered study:
    - Waste resources; especially for needless sacrifice of animals
    - Pick up essentially trivial results which are meaningless
    - Cost of collecting data > potential benefits
  - Stopping rules: see the slides
- The use of **bootstrapping**
  - The data we have is incomplete
  - Fundamentally not an exact method
  - *("But it is all we have!")*
- Power analysis of ANOVA
  - `library(pwr)`
  - `pwr.anova.test(k=GroupNumber, n=SampleNumberPerGroup, f = Cohen'sf, sig.level=alpha)` a power of 0.8 is sufficient.
  - `pwr.anova.test(k=GroupNumber, n=SampleNumberPerGroup, f = Cohen'sf, sig.level=alpha, power=0.8)` to see how many samples per group are needed to achieve a power of 0.8.
- If there are animal died/participants dropped out, it is reasonable to run toxicological tests to find out the most affected organs and check what causes this toxicity.

Concept Table

Distribution terms

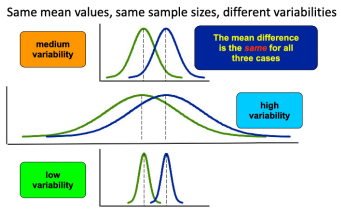
Concept	Definition/Purpose	Equation
Uniform distribution	All outcomes within a specified range are equally likely, e.g., rolling a dice	x
Bernoulli distribution	A discrete distribution representing *a single trial* with two possible outcomes (success or failure), with a probability p of success	x
Binomial distribution	A discrete distribution representing *the number of successes* in n independent trials, each with a probability p of success (that is, a series of Bernoulli distribution)	x
Normal distribution	A continuous, symmetric distribution characterized by its bell-shaped curve, described by its mean μ and standard deviation σ	$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ • Mean: μ • Variance: σ^2 • Characteristics: the 68-95-99.7 rule
Sampling distribution	Take samples of size n from a population which follows a particular distribution pattern and record the mean for multiple times. The distribution of the sample mean is sampling distribution • the more sampling time increases the more sampling distribution resembles the sample • the more sample size increases the more towards-the-center the sampling distribution will get	x

Sampling terms

Concept	Definition/Purpose
Single random sampling	Every member of the population has an equal chance of being selected
Stratified sampling	The population is divided into strata based on certain characteristics, and random samples are taken from each strata according to strata size
Two-stage cluster sampling	• The population is divided into clusters groups • Some clusters are *randomly* selected • Individuals within those selected clusters are randomly sampled • A cost-effective method for large populations spread over wide geographical areas
Extreme sampling	Select the most extreme or deviant cases (very high or very low) in a population • Not representative • Provide an upper/lower bound
Sampling error	• The difference between sample estimate and actual population • Due to randomness, cannot be eliminated • All samples are wrong, some are more useful than others so we can: – Collect more samples – Collect better samples
Sampling bias	• Due to non-random selection of items, whether intentionally or not • Due to sampling method, can be eliminated
Central limit theorem	Even if a population is not normally distributed, the sampling distribution for large enough sample sizes n will tend to be normal. **Summary of n: the more n increases...** • the more towards-the-center will the sampling distribution get • the more normal will the sampling distribution be • the less SEM will be

General statistical terms

CONCEPT TABLE

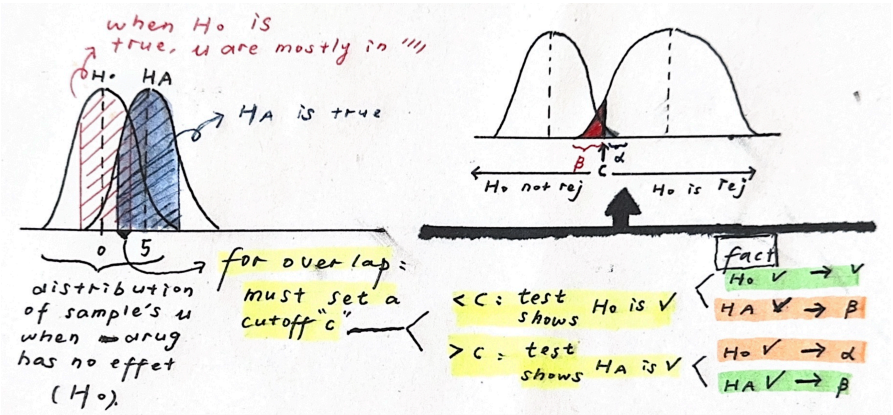
Concept	Definition/Purpose	Equation
Standard score (Z-score)	Z-score provides a standard measure of how many standard deviations a data point is from the mean (distance from the mean), this concept is the basis of statistic such as F-statistic. • $Z = 0$ indicates that the data point is exactly at the mean • Positive Z indicates values above the mean • Reported as "the data point is 2 standard deviations above the mean"	$Z = \frac{X-\mu}{\sigma}$ • The Z-score • X: The raw score (data point) • μ: The mean of the population • σ: The standard deviation of the population
Standard error of the mean (SEM)	A measure of how well the sample mean estimates the population mean • the larger SEM is, the better the sample is, the more appropriate the sampling method is	$SEM = \frac{\sigma}{\sqrt{n}}$ • σ represents the standard deviation of the population • n is the sample size
Compare 2 Samples non-statistically	Depends on 1. Effect Size 2. Sample Size This is the base of the rationale of t-statistic	$t = \frac{\bar{x}-\mu}{\frac{s}{\sqrt{n}}}$
Effect size (Cohen's d)	A quantitative measure of the magnitude of the difference between groups. It helps to uncover the practical meaning of findings, beyond just statistical significance, but magnitude of the effect	**Cohen's d**: Used to measure the standardized mean difference between two groups:$d = \frac{\bar{X}_1-\bar{X}_2}{s}$ • $\bar{X}_1$ and $\bar{X}_2$ are the means of the two groups. • s is the pooled standard deviation. – Small effect: $d = 0.2$ – Medium effect: $d = 0.5$ – Large effect: $d = 0.8$
Confidence Interval	- Confidence level: 95% - Explanation: I am 95% sure that the real value lies in the range ()	95% CI = estimate $\pm\, t_{0.975,df} \cdot$ SEofEstimate
Statistical inference	Infer properties of the population from sampled data using statistical tests Infer parameter (non-hat) from the manipulation of estimation (hat)	*estimation* *(means sampling and get statistics here)* + *hypothesis testing* 
Variability	• Without variability , there is no need for statistics • The lower the variability, the higher probability that we will reject the null hypothesis	

Hypothesis testing terms

	$\mathbf{H_0}$ **is true**	$\mathbf{H_a}$ **is true**
Test shows significance	**V** ***(not wanted)*** $= \alpha(SignificanceLevel)$ Type 1 error False Positive	**S** ***(wanted)*** $= Sensitivity = Power = 1 - \beta$ Correct decision True Positive
Test shows non-significant	**U** ***(wanted)*** $= Specificity$ Correct decision True Negative	**T** ***(not wanted)*** $= \beta$ Type 2 error False Negative

Terms	Name	Formula
FDR	False Discovery Rate	FDR $= \frac{V}{V+S}$
TPR (Sensitivity)	True Positive Rate	TPR $= \frac{S}{S+T}$
FPR (Type 1 error, α**)**	False Positive Rate	α = Type 1 error = FPR $= \frac{V}{V+U}$
Type 2 error, β	False Negative Rate	Type 2 error $= \frac{T}{S+T}$

CONCEPT TABLE



As long as the distribution of H_0 and H_A overlapped, the cutoff is a trade-off between α and β. You have to sacrifice one of those to compensate the other.
For how to balance these two risks, see ABOVE.

Correlation terms

Concept	Definition/Purpose	Equation
Variance	- To measure how much each value in the data set deviates from the mean - Does **NOT** equal to *variability*	$s_x^2 = \frac{\sum(X-\bar{X})^2}{n-1}$
Covariance	- To find out whether two variables move in the same direction (positive covariance) or in opposite directions (negative covariance)	$\text{Cov}(x,y) = \frac{\sum(X-\bar{X})(Y-\bar{Y})}{n-1}$
Correlation coefficient	- Correlation is the standardized covariance - Range: [-1, 1]	$r = \frac{\text{Cov}(x,y)}{s_x s_y}$
Linear regression	- Y: dependent variable/response variable, - X: independent variable/regressor/predictor variable, - β_0: Y-intercept, - β_1: curve slope, - ϵ: error term (unexplained variance)	$Y = \beta_0 + \beta_1 X + \epsilon$
Residuals	Random error that failed to be explained by the model, unwanted	$Residuals = ActualValue - PredictedValue$