CHAPTER 1 & 2: Statistical Inferences: the way of drawing conclusions about the parameter of interest from the whole population that we do not have based on some observed data sampled from it.

- **Population**: The whole group of individuals or objects that have common similarities that we are interested in studying.
- **Population parameter**: The quantitative value we want to measure and infer from the whole population. Example: Proportion of red beads in the whole population. **Hypothesis testing**: claims about an unknown parameter.
- **Observation**: The quantity or quality of a single member of a population.
- **Sample**: A group of individuals or objects that are selected from a larger population as a representation.
- **Sampling**: The process of obtaining a set of values (sample) from the whole population.
- **Random sampling:** Selecting a sample from a population where each individual/object has an equal probability of being included. It is important to avoid any bias and allow generalization while improving accuracy when making inferences of the larger population it was drawn.
- **Point estimate (sample statistics):** The summary (single value) of the statistic we calculated from a sample that estimates the population parameter that we don't have. Example: Sample proportion, sample mean, etc.
- **Sample Distribution**: The variation of values obtained in a single sample. Has a similar shape to the popula
- **Sampling Distribution**: All the possible values of point estimates (statistics) calculated from different random samples from the same population. It is centered on the true population parameter. The process of repeatedly taking a random sample and calculating a point estimate to see the distribution to make a good guess of the unknown population parameter we are interested in.
  - In the sampling distribution of a sample mean, the mean of the sampling dist is always μ (population mean).
  - There is only 1 possible sampling dist of a point estimator, given a population and sample size.
  - If we know the sampling dist, we don't need to estimate the population parameter.
- **Standard Error**: is the standard deviation of the sampling distribution. From this, we can tell how wide the sampling distribution of point estimates will roughly be.

Code library(infer) for virtual sampling

We compute the standard deviation of point estimates (aka **standard error**!)

$$\sigma_{\hat{p}} = \sqrt{\frac{p(1-p)}{n}}$$

- $\sigma_{\hat{p}}$: standard deviation of point estimates
- p: population proportion
- n: sample size

- **Representative:** Representative if the sample's characteristics cover all possible variations/good representation of the population's characteristics (TAs, Profs in course staffs). Selecting a sample in a way that accurately reflects the characteristics of the population as a whole.
- **Generalizable:** The outcome of the sample can be used to say the same thing to the whole population
- **Unbiased:** Every part of the population has an equal chance of getting sampled (no certain parts are higher than the others).
- As the sample **size increases**,
  - Sampling distribution becomes narrow and more likely to shape like a normal distribution (bell-shaped), with less variation and standard error. - There are more sample point estimates closer to the true population proportion. - Point estimates will decrease in variance and will be more concentrated towards the true population parameter, hence will result in higher precision.
  - The standard error of the estimator decreases.
- As sample **repetition increases**, The distributions become more "filled in" (or, similarly, there appear to be fewer missing values) as the number of sample repetitions increases. - The distributions become smoother as the number of sample repetitions increases.
- **Replicates** because our point estimate might be unreliable since our estimates of random samples can vary due to **sampling variability**. Hence, we cannot rely just on our point estimate based on a single sample alone, instead, we need to repeat and get more random samples to verify our results. **Under-est**: Taking bootstrap sizes of a larger size than the original samples results in a narrower bootstrap distribution **Over-est**: Taking fewer samples, does not do a good job of estimating the sampling distribution for the original sample size.

True Population Dist	Sample Dist	Sampling Dist

`data <- filter(!is.na(variable), variable = "") |> select(variable) |> mutate(column = column * something)`
 	
`options(repr.plot.width = 4, repr.plot.height = 3)`
`pop_dist <- data |> ggplot(aes(x = variable)) + geom_histogram(bins = 50, binwidth = 10) + labs(x = "", y = "Count") + ggtitle("Something Distribution") + theme(axis.text.x = element_text(angle = 45, hjust = 1))`
 	
`pop_mean <- data |> summarise(pop_mean = mean(variable))`
	`set.seed(1)`
	`sample <- data |> rep_sample_n(40)`
	
	`sample_mean <- sample |> summarise(sample_mean = mean(variable))`
	
	`sample_dist <- ggplot(sample, aes(x = variable)) + geom_histogram(bins = 50) + xlab("Land Value (CAD)") + ggtitle("Sample distribution") + theme(axis.text.x = element_text(angle = 45, hjust = 1))`
		`samples <- data |> rep_sample_n(size = 40, reps = 1500)`
		
		`sample_estimates <- samples |> group_by(replicate) |> summarise(sample_mean = mean(variable))`
		`or`
		`var() or sd()`
		`median()`
		`prop = sum(variable == "")/n()`
		
		`sampling_distribution <- ggplot(sample_estimates, aes(x = sample_mean)) + geom_histogram(bins = 30) + xlab("Mean Price ($)") + ggtitle("Sampling distribution of the sample means") + theme(text = element_text(size = 20))`

CHAPTER 3 & 4: Bootstrapping & Confidence Intervals: a method used to estimate the sampling distribution by sampling with replacement from the original sample allowing duplicates and then calculating the statistic of interest for each sample, repeatedly. In real life, we will only have one or a few samples from the entire population, we are unable to construct sampling distribution without having the population as a whole. Centered @ stat

- **Replacement**: So that we won't end up with the same sample values again and just be the same as our original sample.
- Contrast the bootstrap and sampling distributions: Bootstrap distribution was created from samples drawn from a single sample while sampling distribution was created from samples drawn from the population. In real life, we will only have one sample and cannot create a sampling distribution, so the distribution of the bootstrap sample estimates can illustrate how we might expect our point estimate to behave if we took another sample. Unlike sampling distribution which has a center at a population parameter, we can see that the bootstrap distribution is centered toward the original sample's mean since we are repeatedly taking samples from the original sample.
- A **confidence interval** is a range of values that are calculated from a sample, and it is where we might expect the true population parameter to lie. Useful for making inferences about population parameters because they provide a range of likely values rather than a single point estimate.

```
bootstrap_sample <- sample %>%
                    rep_sample_n(size = 35, replace = TRUE)

bootstrap_sample_mean <- bootstrap_sample |>
                         summarise(mean = mean(full_years))|>
                         select(mean)|>
                         as.numeric()

bootstrap_sample_dist <- bootstrap_sample %>%
                         ggplot(aes(x = full_years)) +
                         geom_histogram(binwidth = 1, colour = "white") +
                         ggtitle("Bootstrap Sample Distribution") +
                         xlab("# Full Years At UBC") +
                         scale_x_continuous(breaks = seq(0, 10, 1), limits = c(-0.5, 10.5))
```

```
sample_100 <- multi_family_strata %>%
              rep_sample_n(size = 100, replace = TRUE) %>%
              ungroup() %>%
              select(current_land_value)

options(repr.plot.height = 5, repr.plot.width = 4)
bootstrap_dist_100 <- sample_100 %>%
                      rep_sample_n(size = 100, reps = 2000, replace = TRUE) %>%
                      group_by(replicate) %>%
                      summarise(mean_land_value = mean(current_land_value)) %>%
                      ggplot(aes(x = mean_land_value)) +
                      geom_histogram(binwidth = 15000) +
                      xlab("Mean Land Value (CAD)") +
                      ggtitle("n = 100")
```

```
ci <- bootstrap_dist %>%
      summarize(ci_lower = quantile(mean_diameter, 0.05),
                ci_upper = quantile(mean_diameter, 0.95))

intervals_captured <- intervals %>%
                      mutate(captured = (ci_lower <= pop_mean & pop_mean <= ci_upper))
```

Theory-based: Sample size large, independent samples so CLT can work to assume its close to normal. **Bootstrap**: No idea what pop dist and dist of statistics, sample size small

CHAPTER 6: Hypothesis Testing: Trying to find plausible values for a statistic when we have a fixed value for the parameter. For CI, we find parameter. Before taking sample: Elements, prop, std error, boundaries CI, are RANDOM, parameter p CONSTANT. After taking, all CONSTANT except elem bootstrap
The **null hypothesis** is the hypothesis when no change is happening (default), while Ha is the claim we want to investigate with evidence.
Test statistic: A formula of a summary statistic of an observation used for hypothesis testing. Make decisions for our hypothesis based on this.
Observed test statistic: The computed value of our test statistic that we observe in reality
Null Distribution: The distribution of point estimates when the null hypothesis is assumed to be true
Rejection Region: All possible test statistic values that we can reject the null hypothesis p-value < 10%, we have enough evidence to reject $H0$
p-value: How likely it is to get test results at least as extreme as the actual observed result when the null hypothesis is assumed to be true.
Significance level: Denoted by alpha which represents the type I error, it is the cutoff threshold for the p-value where we will reject the hypothesis.
Not reject = lower significance level (a) than reject
Type errors: I: Rejecting H0 when it is actually true (false positive). **II:** Failing to reject H0 when it is actually false (false negative)
Statistically significant: When we reject the null hypothesis (if the p-value is less than alpha/ in the rejection region)
Bootstrapping: Doesn't improve the quality of our estimate, only allows us to study the sampling distribution of our statistic, which would be unknown.
Infer package: NULL DIST

```
specify(formula = body_mass ~ col_to_compare, success = "promoted") or (response = varX, success = "M")
hypothesize(null = "independence" or "point" when one value, mu = 44)
generate(reps = 1000, type = "permute" for null dist or "bootstrap") since we are resampling without replacement
calculate(stat = "diff in props", order = c("male", "female"))
get_p_value(obs_stat = obs_test_stat, direction = "left") or
get_ci(level = 0.90, type = "percentile")

visualize(null_distribution, bins = 10) +
  shade_p_value(obs_stat = obs_diff_prop, direction = "right") for pm-pf > 0 if ≠ "both"
```

$$CI = \bar{x} \pm z\frac{s}{\sqrt{n}}$$

$$CI = \hat{p} \pm Z^*\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

CHAPTER 7: Sample size increase = Narrow sampling dist, possible to *guarantee* (with probability 1!) that the sample mean is as close as we want from the population mean, regardless of the population distribution. **Population distribution affects** how much the **sample mean varies.**
Normal/Gaussian Dist: Symmetric around the mean
The parameter μ controls the location of the curve, while the parameter σ controls its spread. As σ **increases**, the Normal curve becomes wider. Approximately **68%** of the observations are between $[\mu-\sigma;\mu+\sigma]$. **95.5%** of the observations are between $[\mu-2\sigma;\mu+2\sigma]$ **99.7%** of the observations are between $[\mu-3\sigma;\mu+3\sigma]$. μ = 0 and σ = 1, it's called **the *standard normal distribution* or the *z-curve*.**
Central Limit Theorem: Summing up a very large number of random components makes the distribution of this sum is approximately Normal. Large sample size = sampling distribution of the sample will get close/similar to Normal dist, *regardless of the population distribution*.
CLT Limitations: 1. Sample size not large enough.
2. Sample not taken independently (sample size is too large compared to the population's size)
3. Estimator not a sum of random components

We compute the standard deviation of point estimates (aka **standard error**!)

$$\sigma_{\hat{p}} = \sqrt{\frac{p(1-p)}{n}}$$

```
mean_max_flow_ci <-
    mean_max_bootstrap_dist %>%
    summarise(lower_ci = quantile(stat, 0.05), upper_ci = quantile(stat, 0.95))
```

Confidence Intervals from CLT
SHADE CI: visualize(bootstrap_dist) + shade_confidence_interval(endpoints = mean_flow_ci) +xlab('Mean')

```
estimates <- penguins %>%
             filter(species == 'Chinstrap' & !is.na(body_mass_g)) %>%
             summarise(sample_average = mean(body_mass_g), sample_std_error = sd(body_mass_g)/sqrt(n()))

mean_body_mass_chinstrap_ci <- tibble(
    lower_ci = qnorm(0.005, estimates$sample_average, estimates$sample_std_error),
    upper_ci = qnorm(0.995, estimates$sample_average, estimates$sample_std_error)
)
```

Proportion

```
phat <- mean(body_mass_g_adelie > 4000)
answer3.5_mean <- phat
answer3.5_std_error <- sqrt(phat*(1-phat)/length(body_mass_g_adelie))
```

```
salmon_clt_ci <- tibble(lower_ci = salmon_x_bar + qnorm(0.036) * salmon_std_error,
                        upper_ci = salmon_x_bar - qnorm(0.036) * salmon_std_error)
```

Diff means (94% CI)

```
parking_clt_ci <- tibble(lower_ci = (downtown_mean - kits_mean - qnorm(0.97) * sqrt(kits_var + downtown_var)),
                         upper_ci = (downtown_mean - kits_mean + qnorm(0.97) * sqrt(kits_var + downtown_var)))
```

```
p_summary <- p_summary %>%  mutate(p_diff = p_yes - p_no, p_diff_std_error = sqrt(p_yes*(1-p_yes)/n_yes + p_no*(1-p_no)/n_no))
                            mutate(lower_ci = qnorm(0.025, p_diff, p_diff_std_error), upper_ci = qnorm(0.975, p_diff, p_diff_std_error))
```

CHAPTER 8: T Dist: Symmetric, unimodal, bell-shaped, Theory-based relies on pre-defined statistical models and assumptions about the population
Is always centered around 0, has only 1 parameter: degrees of freedom (spread), has heavier tails for uncertainty(for low values of degree freedom), converges to the Normal distribution for large degrees of freedom (50 or more is identical)
Z-score to standardize: (Shows how many standard units away that value is from the mean)

$$z = \frac{x-\mu}{\sigma}$$

CASE1: Population is normally distributed: Assume H0 true, n-1 degrees of freedom
CASE2: Not normal: According to CLT, If n is large enough will become normal

To test H_0, we use the following test statistic:

$$= \frac{(\bar{x}_a - \bar{x}_\tau) - (\mu_a - \mu_\tau)}{\mathrm{SE}_{\bar{x}_a - \bar{x}_\tau}} = \frac{(\bar{x}_a - \bar{x}_\tau) - (\mu_a - \mu_\tau)}{\sqrt{\frac{s_a^2}{n_a} + \frac{s_\tau^2}{n_\tau}}}$$

To test H_0, we use the following test statistic:

$$T = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$$

To test H_0, we use the following test statistic:

$$Z = \frac{\hat{p} - p_0}{\sqrt{p_0(1-p_0)/n}}$$

where $\bar{x}$ is the sample mean, s is the sample standard deviation, n is the sample size, and μ_0 is the value of μ under H_0.

where $\hat{p}$ is the sample proportion, n is the sample size, and p_0 is the value of p under H_0. Since, in this case, the population distribution is clearly not Normal (the random variable is a 0-1 variable), we need to rely on the CLT.

t.test(body_temp$body_temp, mu = 98.2)

answer3.2.3_test_statistic <- (answer3.2.2_phat - 0.5) / sqrt(p0 * (1-p0)/nrow(data))
answer3.2.4_pvalue <- pnorm(answer3.2.3_test_statistic, lower.tail = FALSE)
pt(test_statistic, vdegrees_of_freedom or nrow(..)-1, lower.tail=FALSE)

```
answer3.2.6 <-
    tidy(
        prop.test(
        x = # the number of successes,
        n = # the number of trials,
        p = # p0 (i.e., the value of p under H0),
        alternative = # alternative hypothesis: "less", "greater", "two.sided"
        conf.level = # the desired confidence level,
        correct = FALSE))
```

1. Mean (μ): $\mu = n \cdot p$
   where n is the number of trials and p is the probability of success.
2. Variance (σ^2): $\sigma^2 = n \cdot p \cdot (1-p)$

$$Z = \frac{\hat{p}_1 - \hat{p}_2}{\sqrt{\hat{p}(1-\hat{p})\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

where $\hat{p}_1$ and $\hat{p}_2$ are the sample proportions in samples 1 and 2, respectively, n is the sample size, and $\hat{p}$ is the pooled proportion, given by:

$$\hat{p} = \frac{n_1\hat{p}_1 + n_2\hat{p}_2}{n_1 + n_2}$$

Testing more hypothesis: More prone to Type I error, so increase p-values so tests less likely to pass

CHAPTER 10: Errors (n= samples, 0.05 = 5% sig level,)
Power of tests:

```
pfizer_errors <- tibble(type_I_error = 0.05,
                        type_II_error = 1-pnorm(qnorm(0.05, 275, 35/sqrt(n)), 260, 35/sqrt(n)),
                        power_of_test = pnorm(qnorm(0.05, 275, 35/sqrt(n)), 260, 35/sqrt(n)))
```

By only reporting the p-value, we are missing information on: observed effect size, error of statistic
Powerful if: High chance detecting when H0 is false. (1 – Type II error)
Large overlap between null and sampling dist = higher chance actual sampling dist at non-rejection region (to the right)
Affect TYPE II error: Effect size (True parameter – hypothesized value), sample size (INCREASE = HIGHER POWER, sampling dist narrower, smaller overlap) *if H0 = true, p-val compared against random sample 0-1*

CHAPTER 11: > 2 Group Comparisons

Bonferroni: Aggressive, won't detect subtle effect size, limits seeing at least 1 false positive (Type I) to the specified sig level/alpha
K hypothesis → 1. Adjust sig level to alpha/K or 2. Multiply p-value by K
pval_bonf <- p.adjust(gwas$p_value, method = "bonferroni" or "BH")
BH Adjustment: limits the false discovery rate= the prop discoveries (rejections null) false