Inferential Statistics - Hypothesis Testing.

Prepared By: Siman Giri, Instructor: Ronit and Shiv for Herald Center for AI.

Summer, 2025

1 Learning Objectives.

- Compute and summarize group statistics for hypothesis testing:
  - Learn to calculate sample sizes ($\mathbf{n_i}$), means $\mathbf{x_i}$, and standard deviations $\mathbf{s_i}$ for different groups, which serve as the foundation for conducting **t - tests, z - tests, ANOVA, and Chi - square Tests.**
  - Organize and interpret data for decision making: Understand how to structure group-level data in tables to compare means and variances across regions, enabling proper formulation and testing of null and alternative hypotheses using both single-sample and two-sample tests.

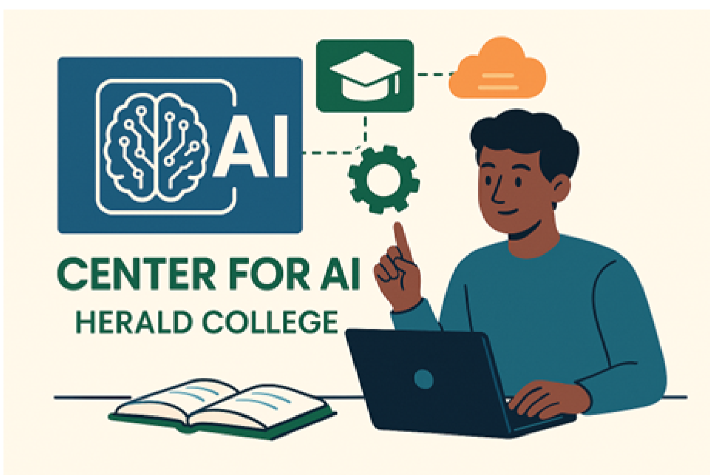


image generated via copilot.

2 Confidence Interval Approach to Hypothesis Testing:

2.1 How to ?

Confidence Interval Hypothesis Testing Procedure

1. **State Hypotheses:**
   - $\mathbf{H_0} : \mu = \mu_0$ **(NullHypothesis)**
   - $\mathbf{H_1} : \mu \neq \mu_0$ **(AlternativeHypothesis)**
2. **Choose Significance Level:**
   - $\alpha = \mathbf{0.05(95\%CI)}$ **or** $\alpha = \mathbf{0.01(99\%CI)}$
3. **Check Assumptions:**
   - **Sample size, normality, data conditions for CI method**
4. **Find Critical Value:**
   - **Using** $\mathbf{z}^*$ (normal) **or** $\mathbf{t}^*$ (t-distribution) **for given** α
5. **Compute Confidence Interval:**
   - $\mathbf{CI} = \bar{\mathbf{x}} \pm (\textbf{critical value} \times \mathbf{SE})$
6. **Decision Rule:**

   if $\mu_0 \notin CI$ **then**
   Reject H_0
   $\Rightarrow$ **Evidence for** H_1 $(\mu \neq \mu_0)$
   else
   Fail to reject H_0
   $\Rightarrow$ **No significant difference found**
7. **Interpretation:**
   - **State conclusion in context of the research question**

2.2 Exercises:

Problem 1 — Email Campaign: mean purchase amount

A marketing team tests a new email layout. A random sample of $\mathbf{n = 40}$ recipients shows an average purchase amount of $\bar{\mathbf{x}} = \$\mathbf{52}$ with sample standard deviation $\mathbf{s} = \$\mathbf{12}$. Test whether the true mean purchase amount differs from $\$\mathbf{50}$ at the $\alpha = \mathbf{0.05}$ level.

Problem 1 — Sample Solution (Email Campaign)

1. **Hypotheses Statement:**

$$\mathbf{H_0} : \ \mu = \mathbf{50} \qquad \text{vs.} \qquad \mathbf{H_a} : \ \mu \neq \mathbf{50}$$

2. **Choose test method, significance level and check assumptions:**
   - two tailed test, with significance level of $\alpha = \mathbf{0.05}$.
   - Unknown population standard deviation, thus use $\mathbf{t - distributions}$ with $\mathbf{df = n - 1 = 40 - 1 = 39}$.

3. **Find Critical Value:**

$$\text{SE} = \frac{\mathbf{s}}{\sqrt{\mathbf{n}}} = \frac{\mathbf{12}}{\sqrt{\mathbf{40}}} = \frac{\mathbf{12}}{\mathbf{6.3246}} \approx \mathbf{1.8974}.$$

   Two-sided $\mathbf{95}\%$ CI uses $\mathbf{t_{0.975,39}}$. From tables or python - scipy.stats:

$$\mathbf{t_{0.975,39}} \approx \mathbf{2.0227}.$$

4. **Compute Confidence Interval:**
   - **Margin of Error:**

$$\text{ME} = \mathbf{t_{0.975,39}} \times \text{SE} \approx \mathbf{2.0227} \times \mathbf{1.8974} \approx \mathbf{3.838}.$$

   - **Confidence interval:**

$$\bar{\mathbf{x}} \pm \text{ME} = \mathbf{52} \pm \mathbf{3.838} \Rightarrow (\mathbf{48.162}, \ \mathbf{55.838}).$$

5. **Decision by CI approach:**
   The null value $\mathbf{50}$ lies inside the $\mathbf{95}\%$ CI $(\mathbf{48.16}, \mathbf{55.84})$. Therefore **Fail to reject $\mathbf{H_0}$** at $\alpha = \mathbf{0.05}$.

6. **Business Interpretation:** Based on the sample, we do not have sufficient evidence **(at 5%)** to conclude the new email layout changes the average purchase amount from $\$\mathbf{50}$. The observed mean $\$\mathbf{52}$ is plausible under the claim of **$50**.

Problem 2 — Delivery time (one-sided)

A logistics manager tests whether a new routing algorithm reduces mean delivery time. From a sample of $\mathbf{n = 25}$ shipments the mean delivery time is $\bar{\mathbf{x}} = \mathbf{3.8}$ days with $\mathbf{s = 0.9}$ days. Test $\mathbf{H_0} : \mu = \mathbf{4}$ vs $\mathbf{H_a} : \mu < \mathbf{4}$ at $\alpha = \mathbf{0.05}$ using the CI approach.

Problem 3 — App feature adoption (proportion)

A product analyst samples $\mathbf{n} = \mathbf{500}$ users; $\mathbf{x} = \mathbf{290}$ have adopted a new feature ($\hat{\mathbf{p}} = \mathbf{0.58}$). Test whether the true adoption proportion differs from 0.60 at $\alpha = \mathbf{0.05}$ (two-sided) using the CI approach.

Problem 4 — Customer satisfaction benchmark (one-sided, high confidence)

A CX team measures customer satisfaction (1–5 scale). From $\mathbf{n} = \mathbf{100}$ responses the sample mean is $\bar{\mathbf{x}} = \mathbf{4.20}$ with $\mathbf{s} = \mathbf{0.80}$. Test whether the true mean exceeds the benchmark of 4.00 at the $\alpha = \mathbf{0.01}$ level using the CI approach.

Problem 5 — NFL linebacker weights (small sample)

Six randomly selected NFL linebackers have weights (lbs): $\mathbf{243}$, $\mathbf{238}$, $\mathbf{229}$, $\mathbf{253}$, $\mathbf{248}$, $\mathbf{225}$. Given $\bar{\mathbf{x}} = \mathbf{239.3}$ and $\mathbf{s} = \mathbf{10.9}$, test whether the mean weight differs from $\mathbf{230}$ lb at $\alpha = \mathbf{0.05}$ using the CI approach.

3 Critical and p - Value Approach:

3.1 Hypothesis Testing Step for t or z Test:

Step 1 - State Hypothesis:

- **Null Hypothesis:**

$$\mathbf{H_0} \text{ :No effect or no difference.} \rightarrow \boldsymbol{\mu = \mu_0}$$

- **Alternative Hypothesis:**

$$\mathbf{H_a} \text{ :There is an effect or Difference.}$$

  - **Two - tailed:** $\boldsymbol{\mu \neq \mu_0}$
  - **One - Tailed (right):** $\boldsymbol{\mu > \mu_0}$
  - **One - Tailed (left)** $\boldsymbol{\mu < \mu_0}$

Step 2 - Choose Significance Level:

- **Common choices:** $\boldsymbol{\alpha = 0.05}$ **Or;** $\boldsymbol{\alpha = 0.01}$

Step 4 - Compute Test Statistics:

- **Check and Validate assumptions regarding Normality, sample size requirements, independence.**
- **t - statistic formula:** When population variance is unknown and sample sizes are $\leq \mathbf{30}$.

t-statistic Formula

$$\mathbf{t} = \frac{\bar{\mathbf{x}} - \mu_\mathbf{0}}{\mathbf{s}/\sqrt{\mathbf{n}}}$$

Where:

  - $\bar{\mathbf{x}}$ = sample mean
  - $\boldsymbol{\mu_0}$ = hypothesized population mean
  - $\mathbf{s}$ = sample standard deviation
  - $\mathbf{n}$ = sample size
  - $\mathbf{df} = \mathbf{n} - \mathbf{1}$ (degrees of freedom)

- **z - statistic formula:** When population variance is known and sample sizes are $\geq \mathbf{30}$.

z-statistic Formula

$$\mathbf{z} = \frac{\bar{\mathbf{x}} - \mu_\mathbf{0}}{\sigma/\sqrt{\mathbf{n}}}$$

Where $\boldsymbol{\sigma}$ = population standard deviation (known)

Step 4 - Determine Critical Values:

Crtical Values using Python - scipy .stats.

```python
from scipy import stats
if test_type.lower() == "z":
    if tails == "two-tailed":
        crit_val = stats.norm.ppf(1 - alpha/2)
    elif tails == "right-tailed":
        crit_val = stats.norm.ppf(1 - alpha)
    elif tails == "left-tailed":
        crit_val = stats.norm.ppf(alpha)
else: # T-test
    if tails == "two-tailed":
        crit_val = stats.t.ppf(1 - alpha/2, df)
    elif tails == "right-tailed":
        crit_val = stats.t.ppf(1 - alpha, df)
    elif tails == "left-tailed":
        crit_val = stats.t.ppf(alpha, df)
```

Step 5 - Decision by Critical Value:

- **For Two Tailed Test:**

$$\text{Reject } \mathbf{H_0} \quad \text{if} \quad |\mathbf{z}| > \mathbf{z}_{\alpha/2} \quad \text{or} \quad |\mathbf{t}| > \mathbf{t}_{\alpha/2,\,\mathbf{df}}$$

- **For Right - Tailed Test:**

$$\text{Reject } \mathbf{H_0} \quad \text{if} \quad \mathbf{z} > \mathbf{z}_{\alpha} \quad \text{or} \quad \mathbf{t} > \mathbf{t}_{\alpha,\,\mathbf{df}}$$

- **For Left - Tailed Test:**

$$\text{Reject } \mathbf{H_0} \quad \text{if} \quad \mathbf{z} < -\mathbf{z}_{\alpha} \quad \text{or} \quad \mathbf{t} < -\mathbf{t}_{\alpha,\,\mathbf{df}}$$

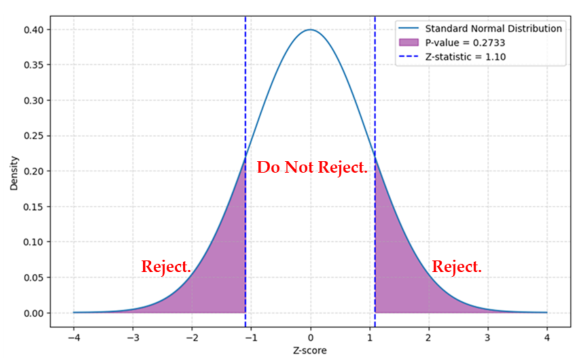


Figure 1: Example Representation of Reject or Not Reject Area on Z - test.

Step 6 - Verification by p - value:

Computing p - value using scipy stats.

```python
from scipy import stats
if test_type.lower() == "z":
    if tails == "two-tailed":
        p_value = 2 * (1 - stats.norm.cdf(abs(test_stat)))
    elif tails == "right-tailed":
        p_value = 1 - stats.norm.cdf(test_stat)
    elif tails == "left-tailed":
        p_value = stats.norm.cdf(test_stat)
else: # T-test
    if tails == "two-tailed":
        p_value = 2 * (1 - stats.t.cdf(abs(test_stat), df))
    elif tails == "right-tailed":
        p_value = 1 - stats.t.cdf(test_stat, df)
    elif tails == "left-tailed":
        p_value = stats.t.cdf(test_stat, df)
```

- If $\boldsymbol{p} \leq \boldsymbol{\alpha}$ or equivalently, the test statistic is in the rejection region.
  - Reject Null Hypothesis $\mathbf{H_0}$.
  - Conclude the result $\mathbf{H_a}$: is Statistically Significant.
- If $\boldsymbol{p} > \boldsymbol{\alpha}$ or equivalently, the test statistic is not in the rejection region.
  - Fail to Reject Null Hypothesis $\mathbf{H_0}$.
  - Conclude the result $\mathbf{H_a}$: is not Statistically Significant at the chosen significance level $\boldsymbol{\alpha}$.

3.2 Algorithm:

1. **State Hypotheses:**

   Two-tailed:
   $$H_0 : \mu = \mu_0, \quad H_1 : \mu \neq \mu_0$$
   Right-tailed:
   $$H_0 : \mu \leq \mu_0, \quad H_1 : \mu > \mu_0$$
   Left-tailed:
   $$H_0 : \mu \geq \mu_0, \quad H_1 : \mu < \mu_0$$

2. **Choose Significance Level:**
   $$\alpha = 0.05 \quad \text{or} \quad 0.01$$

3. **Check Assumptions:**

   Normality, sample size requirements, independence

4. **Calculate Test Statistic:**
   $$z = \frac{\bar{x} - \mu_0}{SE} \quad \text{or} \quad t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$$

5. **Find Critical Value:**
   $$z^* \text{ or } t^* \text{ for given } \alpha$$

6. **Decision Rule:**
   - **Two-tailed:** If $|z| > z^*$ or $|t| > t^*$ $\Rightarrow$ Reject H_0 $\Rightarrow$ Significant
   - **Right-tailed:** If $z > z^*$ or $t > t^*$ $\Rightarrow$ Reject H_0 $\Rightarrow$ Significant
   - **Left-tailed:** If $z < -z^*$ or $t < -t^*$ $\Rightarrow$ Reject H_0 $\Rightarrow$ Significant
   - **Else** $\Rightarrow$ **Fail to reject H_0** $\Rightarrow$ **Not significant**

7. **Verification by p-value:**
   - Two-tailed: $p = 2 \times [1 - T(|\text{statistic}|)]$
   - Right-tailed: $p = 1 - T(\text{statistic})$
   - Left-tailed: $p = T(\text{statistic})$

   If $p < \alpha$ $\Rightarrow$ Reject H_0 **else** $\Rightarrow$ **Fail to reject H_0**

8. **Interpretation:**

   Summarize in context of problem

3.3 Exercises:

Problem A - Retail Price Test:

A retailer advertises that the average basket value is $**75**. A quality analyst randomly samples $\mathbf{n = 12}$ recent transactions and finds a sample mean of $**79.8** with sample standard deviation $\mathbf{s} = \$\mathbf{9.6}$. Test at $\alpha = \mathbf{0.05}$ whether the true mean basket value differs from $**75**.

Problem B - Call Center Wait Time.

A call-center manager claims average wait time is at most **2.5 minutes**. A random sample of $\mathbf{n = 16}$ calls yields $\mathbf{mean - wait = 2.8min}$, $\mathbf{s = 0.9min}$. Test at $\boldsymbol{\alpha}\mathbf{= 0.10}$ whether the mean wait time exceeds **2.5minutes**.

Problem C - Email Click - Through.

A marketing team believes the click-through rate (CTR) of a new campaign is **8**%. In an **A/B trial**, the new campaign got $\mathbf{x = 78}$ clicks out of $\mathbf{n = 900}$ impressions. Test at $\boldsymbol{\alpha}\mathbf{= 0.05}$ whether the CTR differs from **8**% (two-sided).

Part D - Delivery Reliability.

A logistics KPI target is that on-time delivery $\mathbf{rate = 95}\%$. Over one quarter, a sample of $\mathbf{n = 1200}$ deliveries shows $\mathbf{1,100}$ on time. Use $\boldsymbol{\alpha}\mathbf{= 0.01}$ to test whether the true on-time rate differs from **95**% (two-sided).

Problem E - A/B Test Revenue Lift (One - sided, unequal variances, two - sample t/welch)

A product team runs an A/B test to check a new checkout flow.

- **Group A (control):** $\mathbf{n_1 = 45}$**, mean revenue per user** $\bar{\mathbf{x_1}} = \$\mathbf{24.50}, \mathbf{s_1} = \$\mathbf{7.1}$
- **Group B (treatment):** $\mathbf{n_2 = 50}$**, mean revenue per user** $\bar{\mathbf{x}}_\mathbf{2} = \$\mathbf{27.10}, \mathbf{s_2} = \$\mathbf{8.0}$

Test at $\boldsymbol{\alpha}\mathbf{= 0.05}$ whether the new flow increases average revenue $\mathbf{one - sided}, \boldsymbol{\mu_2 > \mu_1}$, using the critical-value (Welch) approach.

Problem E — Sample Solution (Retail Price Test)

1. **Provided Data:**
   - **For Group - Control (A):**

$$\mathbf{n_1 = 45; \bar{x}_1 = 24.50; s_1 = 7.2}$$

   - **For Group - Treatment (B):**

$$\mathbf{n_2 = 50; \bar{x}_2 = 27.10; s_2 = 8.0}$$

2. **Hypotheses Statement:**

$$\mathbf{H_0} : \boldsymbol{\mu_2} = \boldsymbol{\mu_1} \qquad \text{vs.} \qquad \mathbf{H_a} : \ \boldsymbol{\mu_2} > \boldsymbol{\mu_1}$$

3. **Choose test method, significance level and check assumptions:**
   - one right tailed test, with significance level of $\alpha = \mathbf{0.05}$.
   -

4. **Compute Test Statistic (welch - statistic):**

$$\text{SE} = \sqrt{\frac{\mathbf{s_1^2}}{\mathbf{n_1}} + \frac{\mathbf{s_2^2}}{\mathbf{n_2}}.} = \sqrt{\frac{\mathbf{7.2^2}}{\mathbf{45}} + \frac{\mathbf{8.0^2}}{\mathbf{50}}} \approx \mathbf{1.55949}$$

   Degrees of Freedom - Welch - Satterthwaite:

$$\mathbf{df} \approx \frac{(\mathbf{s_1^2/n_1 + s_2^2/n_2})^2}{\frac{(\mathbf{s_1^2/n_1})^2}{\mathbf{n_1-1}} + \frac{(\mathbf{s_2^2/n_2})^2}{\mathbf{n_2-1}}} \approx \mathbf{93.0}$$

5. **Compute Critical Value:**
   - Two-sided $\mathbf{95}\%$ CI uses $\mathbf{t_{0.975,11}}$.
   - From tables or python - scipy.stats: `t_critical = stats.t.ppf(1 - alpha, df)`
   - $\mathbf{t_{crit}} = \mathbf{t_{0.975,93}} = \mathbf{1.6614}$

6. **Decision by Critical value approach:**

$$|\mathbf{t}| > \mathbf{t_{crit}} \ \ \textbf{i.e. } \mathbf{1.6672 > 1.6614} \Rightarrow \texttt{Reject Null}(\mathbf{H_0})$$

7. **Verified by p - value:** Compute **p - value**:
   - **For two tailed test p - value is:**

$$\mathbf{p - value = P(T \geq |t_{observed}|)}$$

     T follows a t - distribution with df degrees of freedom.
   - **Using python and scipy.stats:** `1 - stats.t.cdf(t_statistic, df)`
   - **Decision:** since p - value $\leq \boldsymbol{\alpha}$, we Reject $\mathbf{Null(H_0)}$. $\rightarrow \mathbf{p - value = 0.05} \leq \alpha = \mathbf{0.05}$.

4 Test of Summary Statistics

4.1 One-Way ANOVA Testing Steps

1. **State Hypotheses:**

$$\mathbf{H_0 : \mu_1 = \mu_2 = \cdots = \mu_k} \quad \textbf{(All group means are equal)}$$

$$\mathbf{H_a} \textbf{ :At least one group mean differs}$$

2. **Choose Significance Level:** Select $\boldsymbol{\alpha}$ (e.g., $\boldsymbol{\alpha = 0.05}$).

3. **Calculate Group Means and Overall Mean:**
   Let $\mathbf{\bar{X}_i}$ be the mean of group $\mathbf{i}$, and $\mathbf{\bar{X}_{GM}}$ the grand mean.

   (a) **Group mean for group i:**

$$\mathbf{\bar{X}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} X_{ij}}$$

   where:

   - $\mathbf{X_{ij}}$ = $\mathbf{j}$-th observation in group $\mathbf{i}$
   - $\mathbf{n_i}$ = number of observations in group $\mathbf{i}$

   (b) **Grand mean (mean across all observations in all groups):**

$$\mathbf{\bar{X}_{GM} = \frac{1}{N} \sum_{i=1}^{k} \sum_{j=1}^{n_i} X_{ij}}$$

   where:

   - $\mathbf{k}$ = number of groups
   - $\mathbf{N = \sum_{i=1}^{k} n_i}$ = total number of observations across all groups

4. **Compute ANOVA Components:**

   - **Between-group variability:**

$$\mathbf{SS_B = \sum_{i=1}^{k} n_i(\bar{X}_i - \bar{X})^2}$$

   - **Within-group variability:**

$$\mathbf{SS_W = \sum_{i=1}^{k} \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_i)^2}$$

   - **Compute Mean Squares:**

$$\mathbf{MS_B = \frac{SS_B}{k-1}}, \quad \mathbf{MS_W = \frac{SS_W}{N-k}}$$

5. **Compute F-statistic:**

$$\mathbf{F} = \frac{\mathbf{MS_B}}{\mathbf{MS_W}}$$

6. **Decision Rule (Critical Value Approach):**
   - If $\mathbf{F} > \mathbf{F}^*_{(\mathbf{k-1,N-k})}$ at significance level $\boldsymbol{\alpha}$, reject $\mathbf{H_0}$.

7. **Verification by p-value:**
   - If $\boldsymbol{p} < \boldsymbol{\alpha}$, **reject** $\mathbf{H_0}$ **(statistically significant).**
   - Else, **fail to reject** $\mathbf{H_0}$ **(not significant).**

4.2 Chi-Square Goodness-of-Fit Test

Goal: Test whether observed frequencies match expected frequencies.

1. **State Hypothesis:**

$$\mathbf{H_0}: \textbf{Observed frequencies follow the expected distribution}$$

$$\mathbf{H_a}: \textbf{Observed frequencies do not follow the expected distribution.}$$

2. **Test Statistics:**

$$\chi^{\mathbf{2}} = \sum_{\mathbf{i=1}}^{\mathbf{k}} \frac{(\mathbf{O_i} - \mathbf{E_i})^{\mathbf{2}}}{\mathbf{E_i}}$$

   where:
   - $\mathbf{O_i}$ = **Observed frequency for category i**
   - $\mathbf{E_i}$ = **Expected frequency for category i**
   - $\mathbf{k}$ = **Number of categories.**

3. **Degrees of Freedom:**

$$\mathbf{df = k - 1}$$

   where:
   - $\mathbf{k}$ = **Number of Groups.**

4. **Decision Rule:**
   - **Reject Null Hypothesis** $\mathbf{H_0}$ if $\chi^{\mathbf{2}} > \chi^{\mathbf{2}}_{\text{critical},\mathbf{df},\alpha}$
   - Otherwise, **fail to reject** $\mathbf{H_0}$

5. **Verification by p-value:**
   - If $\boldsymbol{p} < \boldsymbol{\alpha}$, **reject** $\mathbf{H_0}$ **(statistically significant).**
   - Else, **fail to reject** $\mathbf{H_0}$ **(not significant).**

4.3 Chi-Square Test of Independence Steps

1. **State Hypotheses:**

$$\mathbf{H_0} : \textbf{No association between the two categorical variables}$$

$$\mathbf{H_a} : \textbf{There is an association between the two categorical variables}.$$

2. **Choose Significance Level:** Select $\boldsymbol{\alpha}$ (e.g., $\boldsymbol{\alpha = 0.05}$).

3. **Create a Contingency Table:** Organize observed frequencies O_{ij} in a table.

4. **Calculate Expected Frequencies:**

$$\mathbf{E_{ij}} = \frac{(\text{Row Total})_\mathbf{i} \times (\text{Column Total})_\mathbf{j}}{\text{Grand Total}}$$

5. **Compute Chi-Square Statistic:**

$$\chi^{\mathbf{2}} = \sum_{\mathbf{i}} \sum_{\mathbf{j}} \frac{(\mathbf{O_{ij}} - \mathbf{E_{ij}})^{\mathbf{2}}}{\mathbf{E_{ij}}}$$

6. **Decision Rule (Critical Value Approach):**
   - If $\boldsymbol{\chi^2 > \chi^2_{(\alpha,df)}}$ where $\mathbf{df} = (\mathbf{r} - \mathbf{1})(\mathbf{c} - \mathbf{1})$, **reject** $\mathbf{H_0}$.

7. **Verification by p-value:**
   - If $\boldsymbol{p < \alpha}$, **reject** $\mathbf{H_0}$ (statistically significant).
   - Else, **fail to reject** $\mathbf{H_0}$ (not significant).

4.4 Exercises:

Problem A — Which Creative Wins? Comparing Mean CTR Across Ads:

Fresh-mart tested four ad creatives in an email campaign. For each group, you collected CTR (%) from equal sub - samples.

- **Given (Summary Stats:**
  - **Group A:** $\mathbf{n_1 = 30, \bar{x}_1 = 5.2, s_1 = 0.9}$
  - **Group B:** $\mathbf{n_2 = 30, \bar{x}_2 = 5.7, s_2 = 1.0}$
  - **Group C:** $\mathbf{n_3 = 30, \bar{x}_3 = 5.1, s_3 = 0.8}$
  - **Group D:** $\mathbf{n_4 = 30, \bar{x}_4 = 5.9, s_4 = 1.1}$
- Test at $\boldsymbol{\alpha = 0.05}$ using **one - way ANOVA.**

Problem A - sample solution (marketing creatives):

1. **Provided Data:**

$$\begin{aligned}
&\text{Group A: } n_1 = 30,\ \bar{x}_1 = 5.2,\ s_1 = 0.9\\
&\text{Group B: } n_2 = 30,\ \bar{x}_2 = 5.7,\ s_2 = 1.0\\
&\text{Group C: } n_3 = 30,\ \bar{x}_3 = 5.1,\ s_3 = 0.8\\
&\text{Group D: } n_4 = 30,\ \bar{x}_4 = 5.9,\ s_4 = 1.1
\end{aligned}$$

2. **Hypothesis Statement:**

$$\mathbf{H_0} : \boldsymbol{\mu_A = \mu_B = \mu_C = \mu_D} \qquad \text{vs.} \qquad \mathbf{H_a} : \texttt{at least one mean differs}$$

3. **Compute Grand Mean:**

$$\mathbf{\bar{X}_G = \frac{\sum_{i=1}^{4} n_i \bar{X}_i}{\sum_{i=1}^{4} n_i} = \frac{30(5.2 + 5.7 + 5.1 + 5.9)}{120} = 5.475}$$

4. **Sum of squares between (SSB):**

$$\mathbf{SSB = \sum_{i=1}^{4} n_i(\bar{X}_i - \bar{X}_G)^2}$$

Compute for each term:

Table 1: Computation of Sum of Squares Between (SSB)

$\mathbf{Group(i)}$	$\mathbf{\bar{X}_i}$	$\mathbf{\bar{X}_i - \bar{X}_G}$	$\mathbf{(\bar{X}_i - \bar{X}_G)^2}$	$\mathbf{n_i(\bar{X}_i - \bar{X}_G)^2}$
1	5.2	-0.275	0.075625	2.26875
2	5.7	0.225	0.050625	1.51875
3	5.1	-0.375	0.140625	4.21875
4	5.9	0.425	0.180625	5.41875
			SSB	**13.425**

5. **Sum of squares within (SSW) from sample standard deviations:**

$$\mathbf{SSW = \sum_{i=1}^{4}(n_i - 1)s_i^2}$$

$$\begin{aligned}\mathbf{SSW} &= \mathbf{29(0.9^2) + 29(1.0^2) + 29(0.8^2) + 29(1.1^2)} \\ &= \mathbf{29(0.81 + 1.00 + 0.64 + 1.21) = 29(3.66) = 106.14}\end{aligned}$$

6. **Degrees of freedom:**

$$\mathbf{df_B = k - 1 = 4 - 1 = 3}, \qquad \mathbf{df_W = N - k = 120 - 4 = 116}.$$

7. **Mean Squares:**

$$\mathbf{MSB = \frac{SSB}{df_B} = \frac{13.425}{3} = 4.475}, \qquad \mathbf{MSW = \frac{SSW}{df_W} = \frac{106.14}{116} \approx 0.915}$$

8. **Compute F - statistic:**

$$\mathbf{F = \frac{MSB}{MSW} = \frac{4.475}{0.915} \approx 4.891}$$

9. **p-value (from F-distribution):** Using $\mathbf{df_1 = 3}$, $\mathbf{df_2 = 116}$:

$$\mathbf{p = P(F_{3,116} \geq 4.891) \approx 0.00308}$$

using python and scipy.stats - `p_value = stats.f.sf(F_value, df_between, df_within)`

10. **Decision at $\boldsymbol{\alpha = 0.05}$:**

$$\mathbf{Since \rightarrow p \approx 0.0031 < 0.05} \boxed{\texttt{Reject } \mathbf{H_0}}$$

11. **Business Interpretation:** There is strong evidence that at least one creative's mean CTR differs from the others. Next step: perform post-hoc pairwise comparisons (e.g., Tukey HSD) to identify which creatives differ and by how much.

Problem B - Regional Sales Performance

A retailer collected the following average order values ($) from three regions:

North (15 stores)	South (12 stores)	West (18 stores)
82.50	76.80	88.10
85.30	74.20	90.50
79.80	78.50	85.70
83.10	75.90	92.30
81.40	77.30	87.60

Tasks:

1. **Complete the summary statistics table:**

Region	number of sample n_i	Mean $\bar{x}_i$	Standard Deviation s_i
North	15 stores		
South	12 stores		
West	18 stores		

2. **Perform ANOVA at $\alpha = 0.01$**

Problem C — Channel Mix Shift - Has the Sales Channel Mix Changed?

- **Scenario:** Historically, order share by channel is:
  - **Web**: 60%
  - **App**: 30%
  - **Phone**: 10%
- During a new campaign week ($N = 500$), observed orders were:
  - **Web** = 275
  - **App** = 185
  - **Phone** = 40
- Test at $\alpha = 0.05$ using a **chi-square goodness-of-fit test**.

5 Appendix - Sample Python Implementations.

5.1 T Test and Z Test:

Hypothesis Testing with Python

```
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt

def hypothesis_test(sample_mean, mu0, s=None, n=None, sigma=None, alpha=0.05,
                   tails="two-tailed", test_type="t", plot=True):
    """
    General hypothesis testing with Z-test or T-test, including critical value, p-value, and
        visualization.
    """
    """
    Parameters:
        sample_mean : float
            Sample mean
        mu0 : float
            Hypothesized mean
        s : float, optional
            Sample standard deviation (needed for t-test)
        n : int
            Sample size
        sigma : float, optional
            Population standard deviation (needed for z-test)
        alpha : float
            Significance level
        tails : str
            'two-tailed', 'right-tailed', or 'left-tailed'
        test_type : str
            'z' for Z-test, 't' for T-test

    """

    # --- Step 1: Test Statistic ---
    if test_type.lower() == "z":
        if sigma is None:
            raise ValueError("Population standard deviation (sigma) required for Z-test")
        se = sigma / np.sqrt(n)
        test_stat = (sample_mean - mu0) / se
        df = None
    elif test_type.lower() == "t":
        if s is None or n is None:
            raise ValueError("Sample std deviation (s) and sample size (n) required for T-test")
        se = s / np.sqrt(n)
        test_stat = (sample_mean - mu0) / se
        df = n - 1
    else:
        raise ValueError("test_type must be 'z' or 't'")

    # --- Step 2: Critical Value ---
    if test_type.lower() == "z":
        if tails == "two-tailed":
            crit_val = stats.norm.ppf(1 - alpha/2)
```

```
            crit_val_lower = stats.norm.ppf(alpha/2)
            crit_val_upper = crit_val
        elif tails == "right-tailed":
            crit_val = stats.norm.ppf(1 - alpha)
            crit_val_lower = None
            crit_val_upper = crit_val
        elif tails == "left-tailed":
            crit_val = stats.norm.ppf(alpha)
            crit_val_lower = crit_val
            crit_val_upper = None
    else: # T-test
        if tails == "two-tailed":
            crit_val = stats.t.ppf(1 - alpha/2, df)
            crit_val_lower = stats.t.ppf(alpha/2, df)
            crit_val_upper = crit_val
        elif tails == "right-tailed":
            crit_val = stats.t.ppf(1 - alpha, df)
            crit_val_lower = None
            crit_val_upper = crit_val
        elif tails == "left-tailed":
            crit_val = stats.t.ppf(alpha, df)
            crit_val_lower = crit_val
            crit_val_upper = None

    # --- Step 3: Decision Rule (Critical Value Method) ---
    if tails == "two-tailed":
        reject_cv = abs(test_stat) > crit_val
        decision_cv = "Reject H" if reject_cv else "Do not reject H"
    elif tails == "right-tailed":
        reject_cv = test_stat > crit_val
        decision_cv = "Reject H" if reject_cv else "Do not reject H"
    elif tails == "left-tailed":
        reject_cv = test_stat < crit_val
        decision_cv = "Reject H" if reject_cv else "Do not reject H"

    # --- Step 4: P-value ---
    if test_type.lower() == "z":
        if tails == "two-tailed":
            p_value = 2 * (1 - stats.norm.cdf(abs(test_stat)))
        elif tails == "right-tailed":
            p_value = 1 - stats.norm.cdf(test_stat)
        elif tails == "left-tailed":
            p_value = stats.norm.cdf(test_stat)
    else: # T-test
        if tails == "two-tailed":
            p_value = 2 * (1 - stats.t.cdf(abs(test_stat), df))
        elif tails == "right-tailed":
            p_value = 1 - stats.t.cdf(test_stat, df)
        elif tails == "left-tailed":
            p_value = stats.t.cdf(test_stat, df)

    reject_pv = p_value < alpha
    decision_pv = "Reject H" if reject_pv else "Do not reject H"


    # --- Step 5: Plot ---
    if plot:
```

```
        x = np.linspace(-5, 5, 1000) if test_type.lower() == "t" else np.linspace(-4,4,1000)
        y = stats.t.pdf(x, df) if test_type.lower() == "t" else stats.norm.pdf(x)
        plt.figure(figsize=(8,4))
        plt.plot(x, y, label=f'{test_type.upper()} distribution')
        # Rejection regions
        if tails == "two-tailed":
            plt.fill_between(x, 0, y, where=(x <= crit_val_lower) | (x >= crit_val_upper), color='red',
                alpha=0.3, label="Rejection region")
        elif tails == "right-tailed":
             plt.fill_between(x, 0, y, where=(x >= crit_val_upper), color='red', alpha=0.3, label="
                 Rejection region")
        elif tails == "left-tailed":
            plt.fill_between(x, 0, y, where=(x <= crit_val_lower), color='red', alpha=0.3, label="
                Rejection region")

        plt.axvline(test_stat, color='blue', linestyle='--', linewidth=2, label=f"Test statistic ({
            test_stat:.2f})")
        if crit_val_lower is not None:
             plt.axvline(crit_val_lower, color='red', linestyle='-', linewidth=2, label=f"Critical value
                  ({crit_val_lower:.2f})")
        if crit_val_upper is not None:
             plt.axvline(crit_val_upper, color='red', linestyle='-', linewidth=2)


        plt.title(f"Hypothesis Test Visualization\nCritical Value Decision: {decision_cv}, P-value
            Decision: {decision_pv}")
        plt.xlabel("Test Statistic")
        plt.ylabel("Density")
        plt.legend()
        plt.grid(True)
        plt.show()

    return {
        "test_statistic": test_stat,
        "critical_value": crit_val,
        "reject_by_critical_value": reject_cv,
        "decision_critical_value": decision_cv,
        "p_value": p_value,
        "reject_by_p_value": reject_pv,
        "decision_p_value": decision_pv,
        "df": df
    }

# ---------------- Example Test ----------------
# One-tailed right T-test - Adjusted to get test statistic between 2 and 3
result = hypothesis_test(sample_mean=46.8, mu0=45, s=8, n=100, alpha=0.05,
                        tails="right-tailed", test_type="t")

print("\nHypothesis Test Results:")
print(f"Test Statistic: {result['test_statistic']:.4f}")
print(f"Critical Value: {result['critical_value']:.4f}")
print(f"P-value: {result['p_value']:.4f}")
print(f"Decision (Critical Value): {result['decision_critical_value']}")
print(f"Decision (P-value): {result['decision_p_value']}")
```

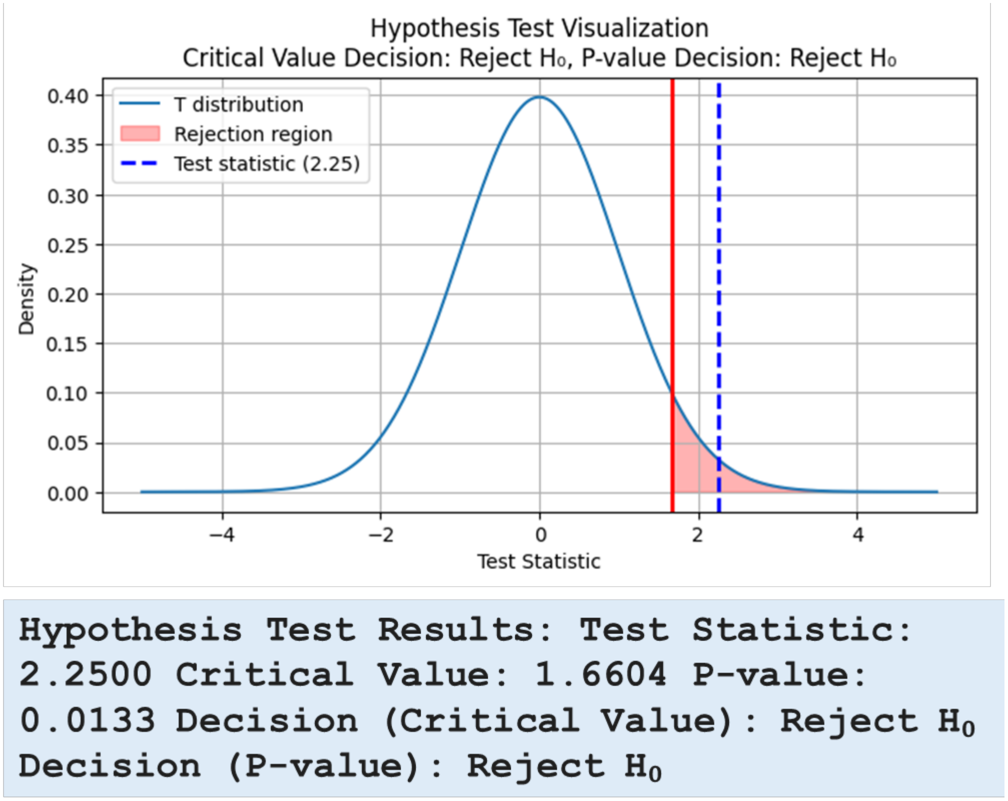


Figure 2: Sample output T - Test

5.2 One - Way ANOVA Test:

One-Way-ANOVA with Python

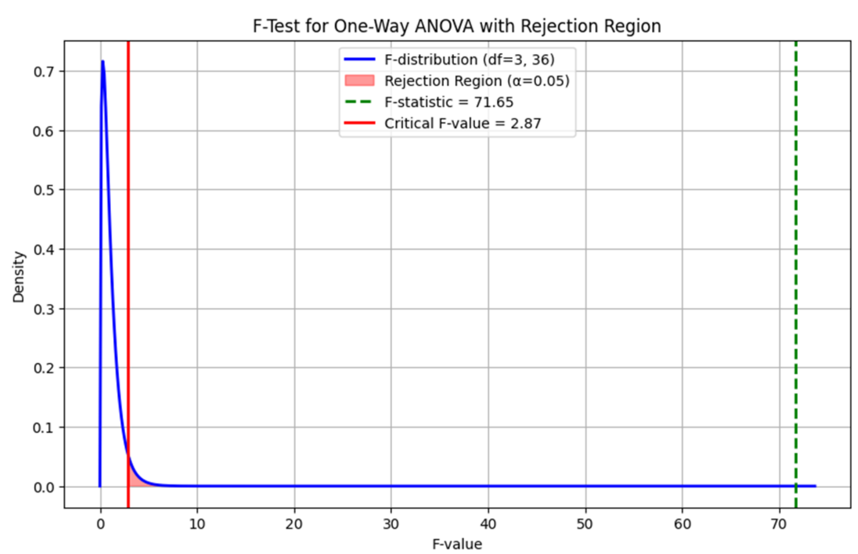
```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import f, stats

# Sample data: click-through rates (%) for 4 ad creatives
# Each group represents CTRs from 10 randomly sampled customers
group_A = np.array([5.2, 5.5, 5.1, 5.4, 5.3, 5.6, 5.2, 5.5, 5.4, 5.3])
group_B = np.array([5.7, 5.8, 5.6, 5.9, 5.5, 5.6, 5.7, 5.8, 5.7, 5.6])
group_C = np.array([5.1, 5.0, 5.2, 5.1, 5.3, 5.0, 5.2, 5.1, 5.3, 5.0])
group_D = np.array([5.9, 6.0, 5.8, 5.7, 5.9, 5.8, 5.9, 6.0, 5.8, 5.9])

# Perform one-way ANOVA
F_statistic, p_value = stats.f_oneway(group_A, group_B, group_C, group_D)
print(f"F-statistic: {F_statistic:.3f}")
print(f"P-value: {p_value:.4f}")
# Decision at alpha = 0.05
alpha = 0.05
if p_value < alpha:
    print("Reject H0: There is a significant difference among the group means.")
else:
    print("Fail to reject H0: No significant difference among the group means.")

# Degrees of freedom for the F-test in one-way ANOVA
num_groups = 4
```

```
observations_per_group = 10
total_observations = num_groups * observations_per_group
df_numerator = num_groups - 1
df_denominator = total_observations - num_groups
# Calculate critical F-value
critical_f_value = f.ppf(1 - alpha, df_numerator, df_denominator)
# Plotting the F-distribution
x = np.linspace(0, F_statistic + 2, 500) # Adjust upper limit based on F-statistic
y = f.pdf(x, df_numerator, df_denominator)
plt.figure(figsize=(10, 6))
plt.plot(x, y, color='blue', lw=2, label=f'F-distribution (df={df_numerator}, {df_denominator})')
# Shade the rejection region
x_reject = np.linspace(critical_f_value, x[-1], 300) # Shade from critical value to the right
plt.fill_between(x_reject, f.pdf(x_reject, df_numerator, df_denominator), color='red', alpha=0.4,
                label=f'Rejection Region')
# Mark the F-statistic
plt.axvline(F_statistic, color='green', linestyle='--', lw=2, label=f'F-statistic = {F_statistic:.2f}')
# Mark the critical F-value
plt.axvline(critical_f_value, color='red', linestyle='-', lw=2, label=f'Critical F-value = {
    critical_f_value:.2f}')
plt.title('F-Test for One-Way ANOVA with Rejection Region')
plt.xlabel('F-value')
plt.ylabel('Density')
plt.legend()
plt.grid(True)
plt.show()
```




```
F-statistic: 71.655 P-value: 0.0000 Reject
H0: There is a significant difference among
the group means.
```

Figure 3: Sample Output One - Way - ANOVA

—————— The - End ——————