Name: Monika Kaphle

HCAI5DS02_Monika_Kaphle_2408878_Week_5,6.ipynb

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all

```python
import scipy.stats as stats
import numpy as np

# Given data
n = 50
sample_mean = 4.2
sample_std = 1.1
confidence = 0.95

# Step 1: compute the standard error
se = sample_std / np.sqrt(n)

# Step 2: get the critical t value (two-tailed)
alpha = 1 - confidence
t_crit = stats.t.ppf(1 - alpha/2, df=n - 1)

# Step 3: compute the margin of error
moe = t_crit * se

# Step 4: build the confidence interval
ci_lower = sample_mean - moe
ci_upper = sample_mean + moe
ci = (ci_lower, ci_upper)

print("t-critical:", t_crit)
print("Standard error:", se)
print("Margin of error:", moe)
print("95% CI for mean delivery time:", ci)
```

```
t-critical: 2.0095752371292397
Standard error: 0.15556349186104046
Margin of error: 0.31261654104530295
95% CI for mean delivery time: (np.float64(3.887383458954697), np.float64(4.512616541045303))
```

```python
import scipy.stats as stats
import numpy as np

# Given data
n = 400
x = 128
confidence = 0.90

# Step 1: compute sample proportion
p_hat = x / n

# Step 2: compute standard error for proportion
se_proportion = np.sqrt(p_hat * (1 - p_hat) / n)

# Step 3: find the z critical value
alpha = 1 - confidence
z_crit = stats.norm.ppf(1 - alpha/2)

# Step 4: compute confidence interval
ci_lower_proportion = p_hat - z_crit * se_proportion
ci_upper_proportion = p_hat + z_crit * se_proportion
ci_proportion = (ci_lower_proportion, ci_upper_proportion)

print("Sample proportion:", p_hat)
print("Standard error (proportion):", se_proportion)
print("z-critical:", z_crit)
print("90% CI for proportion:", ci_proportion)
```

```
Sample proportion: 0.32
Standard error (proportion): 0.0233238075793812
z-critical: 1.6448536269514722
90% CI for proportion: (np.float64(0.28163575050873657), np.float64(0.35836424949126344))
```

```
import scipy.stats as stats
import numpy as np

# Given data
n1, mean1, std1 = 40, 5200, 610
n2, mean2, std2 = 35, 4900, 580
confidence = 0.95

# Step 1: compute standard error for the difference
se_diff = np.sqrt(std1**2 / n1 + std2**2 / n2)

# Step 2: degrees of freedom (Welch's approximation)
dof = (std1**2/n1 + std2**2/n2)**2 / ((std1**2/n1)**2 / (n1 - 1) + (std2**2/n2)**2 / (n2 - 1))

# Step 3: get critical t value
alpha = 1 - confidence
t_crit_diff = stats.t.ppf(1 - alpha/2, df=dof)

# Step 4: compute confidence interval for the difference
ci_lower_diff = (mean1 - mean2) - t_crit_diff * se_diff
ci_upper_diff = (mean1 - mean2) + t_crit_diff * se_diff
ci_diff = (ci_lower_diff, ci_upper_diff)

print("Standard error (difference):", se_diff)
print("Degrees of freedom:", dof)
print("t-critical (difference):", t_crit_diff)
print("95% CI for difference in means:", ci_diff)
```

```
Standard error (difference): 137.52791924343424
Degrees of freedom: 72.47603755653496
t-critical (difference): 1.993239891612047
95% CI for difference in means: (np.float64(25.87386515358679), np.float64(574.1261348464132))
```

Task

Simulate 100 samples of size 30 from a normal distribution with mean 50 and standard deviation 10. For each sample, compute a 95% confidence interval for the mean. Plot the confidence intervals, coloring them based on whether they contain the true mean.

Set up simulation parameters

Subtask:

Define the true mean and standard deviation of the population, the sample size, the number of simulations, and the confidence level.

Reasoning: The subtask requires defining several variables with specific values. I will define these variables in a single code block.

```
true_mean = 50
true_std = 10
sample_size = 30
num_simulations = 100
confidence_level = 0.95
```

```python
confidence_intervals = []
contains_true_mean = []

for _ in range(num_simulations):
    # Generate a random sample
    sample = np.random.normal(loc=true_mean, scale=true_std, size=sample_size)

    # Calculate sample statistics
    sample_mean = np.mean(sample)
    sample_std = np.std(sample, ddof=1) # Use ddof=1 for sample standard deviation

    # Calculate standard error
    se = sample_std / np.sqrt(sample_size)

    # Calculate critical t-value
    alpha = 1 - confidence_level
    t_crit = stats.t.ppf(1 - alpha/2, df=sample_size - 1)

    # Calculate margin of error
    moe = t_crit * se

    # Calculate confidence interval
    ci_lower = sample_mean - moe
    ci_upper = sample_mean + moe
    confidence_intervals.append((ci_lower, ci_upper))

    # Check if the true mean is within the interval
    contains_true_mean.append(ci_lower <= true_mean <= ci_upper)

print("Number of confidence intervals calculated:", len(confidence_intervals))
print("Number of intervals containing the true mean:", sum(contains_true_mean))
```

```
Number of confidence intervals calculated: 100
Number of intervals containing the true mean: 98
```

Visualize the confidence intervals

Subtask:

Plot the calculated confidence intervals, color-coding them based on whether they contain the true mean.

Reasoning: Plot the calculated confidence intervals, color-coding them based on whether they contain the true mean, and add a vertical line for the true mean.

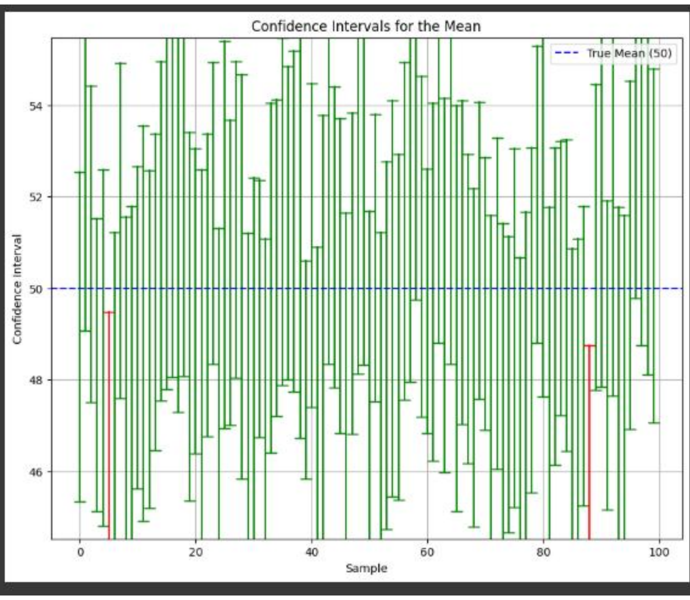
```python
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 8))

for i, ci in enumerate(confidence_intervals):
    color = 'green' if contains_true_mean[i] else 'red'
    plt.plot([i, i], [ci[0], ci[1]], marker='_', color=color, markersize=10)
    plt.plot(i, ci[0], marker='_', color=color, markersize=10)
    plt.plot(i, ci[1], marker='_', color=color, markersize=10)


plt.axhline(y=true_mean, color='blue', linestyle='--', label=f'True Mean ({true_mean})')

plt.ylim(true_mean - 3 * true_std / np.sqrt(sample_size), true_mean + 3 * true_std / np.sqrt(sample_size)) # Adjust y-limits based on expected range
plt.xlabel('Sample')
plt.ylabel('Confidence Interval')
plt.title('Confidence Intervals for the Mean')
plt.legend()
plt.grid(True)
plt.show()
```



Analyze and interpret the results

Subtask:

Examine the plot to see the proportion of confidence intervals that contain the true mean and relate this to the chosen confidence level.

Reasoning: Calculate the proportion of confidence intervals that contain the true mean and compare it to the confidence level.

```
# Count the number of intervals containing the true mean (green lines)
num_containing_true_mean = sum(contains_true_mean)

# Calculate the proportion
proportion_containing_true_mean = num_containing_true_mean / num_simulations

# Compare the proportion to the confidence level
print(f"Number of confidence intervals containing the true mean: {num_containing_true_mean}")
print(f"Proportion of intervals containing the true mean: {proportion_containing_true_mean:.4f}")
print(f"Chosen confidence level: {confidence_level}")

# Analysis
print("\nAnalysis:")
print(f"The observed proportion ({proportion_containing_true_mean:.4f}) of confidence intervals that contain the true mean")
print(f"is close to the chosen confidence level ({confidence_level}). This is expected, as a 95%")
print("confidence level implies that, in the long run, approximately 95% of the confidence intervals")
print("constructed from random samples will contain the true population parameter.")
```

```
Number of confidence intervals containing the true mean: 98
Proportion of intervals containing the true mean: 0.9800
Chosen confidence level: 0.95

Analysis:
The observed proportion (0.9800) of confidence intervals that contain the true mean
is close to the chosen confidence level (0.95). This is expected, as a 95%
confidence level implies that, in the long run, approximately 95% of the confidence intervals
constructed from random samples will contain the true population parameter.
```

Summary:

Data Analysis Key Findings

- Out of 100 simulated samples, 98 of the 95% confidence intervals calculated for the mean contained the true mean of 50.
- The proportion of confidence intervals containing the true mean was 0.9800.

Insights or Next Steps

- The observed proportion of confidence intervals containing the true mean (0.9800) is close to the theoretical 95% confidence level, which is expected.
- This simulation visually demonstrates the meaning of a confidence interval: it represents the long-run probability that an interval constructed in this manner will contain the true population parameter.

$M.E = \frac{15}{2} = 7.5$

from z-tables,

$z_{0.01} \approx 2.4$

$n = \left(\frac{z_{\alpha/2} \cdot \sigma}{ME}\right)^2$

$= \left(\frac{2.4 \times 12}{7.5}\right)^2$

≈ 13.9

$n = 14$

(9) Soln.

$\sigma = 2$

confidence level = 92%.

Margin of Error (ME) = 5

$\alpha = 1 - 0.92 = 0.08$

$\alpha/2 = 0.04$

from z-tables,

$z_{0.04} \approx 1.8$

$n = \left(\frac{z_{\alpha/2} \cdot \sigma}{ME}\right)^2$

$= \left(\frac{1.8 \times 2}{5}\right)^2$

≈ 0.518

$n = 1.$

(c) 80%, $n = 7$

Sol^n.

$df = n-1 = 7-1 = 6$

$\alpha = 1-0.8 = 0.2 = 0.1$

$t = 1.440$

2.

1. Suppose $\sigma = 4$, confidence interval = 90%, maximum width = 10.

Sol^n.

For 90% confidence level:

$\alpha = 1-0.9 = 0.1 = 0.05$

$ME = \frac{10}{2} = 5$

$z_{0.05} \approx 1.6$

Now,

$$n = \left(\frac{z_{\alpha/2} \cdot \sigma}{M.E}\right)^2$$

$$= \left(\frac{1.6 \times 4}{5}\right)^2 = 1.638$$

$$n = [1.638] = 2.$$

Sol^n.

$\sigma = 12$

confidence level = 98%.

Maximum width = 15

$\alpha = 1-0.98$

$= 0.02$

$\alpha_{/2} = 0.01$

(3)

$M.E = \frac{15}{2} = 7.5$

from z-tables,

$Z_{0.01} \approx 2.4$

$$n = \left(\frac{Z_{\alpha/2} \cdot \sigma}{ME}\right)^2$$

$$= \left(\frac{2.4 \times 12}{7.5}\right)^2$$

≈ 13.9

$n = 14$

(9) Soln.

$\sigma = 2$

confidence level = 92%.

Margin of Error (ME) = 5

$\alpha = 1 - 0.92 = 0.08$

$\alpha/2 = 0.04$

from z-tables,

$Z_{0.04} \approx 1.8$

$$n = \left(\frac{Z_{\alpha/2} \cdot \sigma}{ME}\right)^2$$

$$= \left(\frac{1.8 \times 2}{5}\right)^2$$

≈ 0.518

$n = 1.$

4. Sol^n.

$\sigma = 10$

confidence level $= 99\%$.

$ME = 3$

$\alpha = 1 - 0.99 = 0.01$

$\alpha_{/2} = 0.005$

from z tables;

$z_{0.005} \simeq 2.5$

$$n = \left(\frac{z_{\alpha/2} \cdot \sigma}{ME}\right)^2$$

$$= \left(\frac{2.5 \times 10}{3}\right)^2$$

$\simeq 69.44$

$n = 70$.

(4) Sample size (n) = 40
sample mean ($\bar{x}$) = 12 %.
Sample standard deviation (s) = 3 %.
Confidence level : 95 % = 0.95
Using z distribution,

$1 - 0.95 = 0.05$

$0.05 = 1.96$

$M.E = Z \times SE = \cancel{1.95}\ 1.96 \times \frac{3}{\sqrt{40}} \left(\frac{s}{\sqrt{n}}\right)$

12 ± 0.926

12 ± 0.929

$+ = 11.07\%$ | $- = 12.93\%$.

(7) $n = 25$
mean = 3400
$s = 600$
Est = 99 % = 0.99

$1 - 0.99 = 0.01$

$M.E = t \times SE = 248.5 \times \frac{600}{\sqrt{25}}$

$= \frac{248.5 \times 6}{5}$

$= 298.2$

$CI = 3400 \pm 298.2$

$= {}^{+ve}3698.2,\ {}^{-ve}3101.8.$

(5) $n = 36$

mean $= 14$

$s = 2$

confidence level $= 90\% = 0.90$

$= 1 - 0.90$

$= 0.10$

$\alpha/_2 = 0.10/_2 = 0.05$

Now, from z table;

$Z_{0.05} = 1.645$

Now, $SE = \frac{s}{\sqrt{n}} = \frac{s}{\sqrt{36}} = \frac{2}{6} = 0.3333$

Now, $ME = (Z_{0.05} \times SE)$

$= 1.645 \times 0.3333$

$= 0.548$

Now, confidence interval (CI) $= \bar{x} \pm ME$

$= 14 \pm 0.548$

= +ve	-ve
14.548	13.452.

(8) $n = 20$
mean $= 107.3$
$S = 13.7$

(a) t distribution
(b) $95\% = 0.95$
$= 1 - 0.95$
$= 0.05$

(a) t value as $n < 30$.
$df = n - 1 = 20 - 1 = 19$

(b) SE(standard Error) $= \frac{S}{\sqrt{n}} = \frac{13.7}{\sqrt{20}} = \frac{13.7}{4.472} = 3.063$

(*) t-critical value
$df = 19$
confidence $= 95\% = 0.95$
$\alpha = 1 - 0.95 = 0.05$
$\alpha/2 = \frac{0.05}{2} = 0.025$

Now,
$t_{0.975}$ where $df = 19 = 2.093$

Now,
$ME = t_{0.975} \times SE = 2.093 \times 3.063$
$= 6.41085g$

Now,
$CI = \bar{x} \pm ME$
$= 107.3 \pm 6.41$
$= \pm ve$ $-ve$
~~113.3~~
$= 113.71$ 100.89

(c) confidence = 98% = 0.98
degree of freedom $df = 19$
singnificance level $\alpha = 1 - 0.98 = 0.02$

$t_{0.02} = 2.539$

$ME = t_{0.02} \times SE$
$= 2.539 \times 3.063$
$= 7.776$

$CI = 107.3 \pm 7.776$
$+ve = 115.076$ $-ve = 99.524$

(d) $\mu = 105$ (True mean)
CI = 95% (113.71, 100.89)
CI = 98% (115.076, 99.524)
as 105 falls under the interval it did a good job.

(9) $n = 32$ (sample size)
$S = 18$ (standard deviation)
mean($\bar{x}$) = 31
t = 95% (confidence level)

(a) t distribution because sample size is greater than 30.

(b) now,
$n = 31 - 1 = 30$ (value lina ko lagi)

$\alpha = 1 - 95\%$
$= 1 - 0.05$
$= 0.95$

$\alpha/2 = \frac{0.95}{2} = 0.025$

$t = 2.042$

$ME = t - \frac{s}{\sqrt{n}}$

$= 6.49$

31 ± 6.49

+ve, -ve

37.49 24.51.

(6.) level of confidence = 95% = 0.95

sample size (n) = 300

Number of success (x) = 240

Sample ~~distribution~~ proportion $(\hat{p}) = \frac{240}{300} = 0.8$.

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

$= \sqrt{\frac{0.8(1-0.8)}{300}}$

$= 0.023$

$ME = 2 \times SE$ (0.95 level of confidence value in z table is 1.96)

$= 1.96 \times 0.02309401077$

$= 0.04526426 11$

$CI = \hat{p} \pm ME$

$= 0.8 \pm 0.045$

+ve $= 0.845$ | -ve 0.755

9 (c) 90%.

$t_{0.05, 31} = 1.696$

$ME = 1.696 \times 3.183$

$= 5.803$

$CI = 31 \pm 5.40$

$= (25.60, 36.40)$

1.

a. 95%, n = 19

Here,

degree of freedom = n - 1

= 19 - 1

= 18

alpha (α) = 1 - confidence level

= 1 - 0.95

= 0.05

$\alpha/2$ = 0.025 (since it is two tailed)

from t-table,

t-critical = 2.101

b. 90%, n = 27

Soln,

df = n - 1 = 27 - 1 = 26

α = 1.090

= 0.1

$\alpha/2$ = 0.05

t = 1.706

(c) 80%, n = 7

Soln.

df = n - 1 = 7 - 1 = 6

α = 1 - 0.8 = 0.2 = 0.1

t = 1.440

2.