Chapter 5: Statistical Inference and Analysis

Firas IBRAHIM

2025-2026

Contents

0.1 Introduction to Statistical Inference

In the previous chapter, we developed the tools needed to understand randomness and model uncertain events using probability. However, in practical situations, we rarely have access to information about an entire population. Instead, we typically observe only a sample—a limited subset drawn from a much larger group.

Statistical inference provides the methodology that allows us to move from what we observe in the sample to conclusions about the population it represents. It is the foundation that connects data to decision-making.

This section introduces the essential ideas of statistical inference:

- **Point estimation:** using sample data to estimate unknown population parameters such as the mean or proportion.
- **Confidence intervals:** quantifying the uncertainty of these estimates by constructing intervals likely to contain the true parameter value.
- **Hypothesis testing:** a formal framework for evaluating claims about a population, including defining null and alternative hypotheses, computing test statistics, and interpreting p-values.
- **Relevance to machine learning:** understanding statistical significance, validating model results, and ensuring that conclusions drawn from data are reliable.

These concepts allow us to make justified generalizations from observed data to the broader population, forming a key component of statistical reasoning and many machine learning applications.

0.2 Point Estimation

Statistical inference provides a framework for learning about an entire population using only information gathered from a sample. One of the most direct tools in this process is **point estimation**. when a population characteristic is unknown, we use

a numerical value computed from sample data to serve as our best single estimate of that characteristic.

A crucial distinction in this context is between **population parameters** and **sample statistics**:

- **Population parameters** are fixed numerical values that describe the whole population. These quantities are typically unknown and represent what we aim to estimate. Examples include:
  - true population mean μ,
  - true population proportion p,
  - the population variance σ^2 or standard deviation σ.
- **Sample statistics** are numerical summaries computed from observed data. They are used as estimates for the corresponding population parameters. Examples include:
  - sample mean $\bar{x}$,
  - sample proportion $\hat{p}$,
  - sample variance s^2 or standard deviation s.

0.3 Common Point Estimators

0.3.1 Estimating the Population Mean (μ)

The standard estimator for the population mean is the **sample mean**, denoted by $\bar{X}$. It is computed by averaging all values in the sample:

$$\bar{X} = \frac{X_1 + X_2 + \cdots + X_n}{n}$$

For example, if a sample of 500 individuals yields a mean height of $\bar{X} = 165.2$ cm, then 165.2 cm is our point estimate of the true population mean μ.

0.3.2 Estimating the Population Proportion (p)

To estimate the true proportion of a population possessing a certain characteristic, we use the **sample proportion**, written $\hat{p}$:

$$\hat{p} = \frac{\text{number of successes in sample}}{\text{sample size}}$$

If 10 out of 200 inspected products are defective, then:

$$\hat{p} = \frac{10}{200} = 0.05$$

Thus, the point estimate of the population proportion is $\hat{p} = 0.05$ (or 5%).

0.3.3 Estimating the Population Variance (σ^2)

A widely used estimator for the population variance is the **sample variance**, denoted by S^2:

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X})^2$$

The use of $n-1$ in the denominator makes S^2 an **unbiased estimator** of the population variance σ^2. After computation, the resulting value s^2 is our point estimate of σ^2.

0.4 Example: Estimating the Average Order Value

Suppose we collect a sample of 50 order values. Using Python, we can compute both the sample mean and the sample standard deviation, which serve as our point estimates for μ and σ respectively.

```
import numpy as np

# Sample of 50 recent order values (in dollars)
order_values = np.array([
```

```
    55.2, 34.1, 89.0, 120.5, 42.0, 65.8, 22.1, 98.7, 77.3,
        105.6,
    30.9, 48.2, 71.5, 60.0, 112.8, 58.4, 88.2, 45.1, 99.9,
        75.0,
    135.2, 28.6, 50.0, 68.9, 91.3, 102.1, 40.5, 79.8, 66.2,
        84.7,
    115.0, 33.5, 52.8, 70.1, 95.4, 108.3, 47.9, 62.7, 81.6,
        100.2,
    36.8, 56.7, 73.9, 64.3, 89.1, 110.5, 49.6, 67.0, 83.3,
        97.4
])

# Sample mean: point estimate for mu
sample_mean_aov = np.mean(order_values)

# Sample standard deviation: point estimate for sigma
# ddof=1 gives the unbiased estimator
sample_std_dev = np.std(order_values, ddof=1)

print(f"Sample Size (n): {len(order_values)}")
print(f"Point Estimate for Average Order Value (  ): ${
    sample_mean_aov:.2f}")
print(f"Point Estimate for Standard Deviation (  ): ${
    sample_std_dev:.2f}")
```

Based on this sample, we obtain:

$$n = 50, \qquad \hat{\mu} = 74.29, \qquad \hat{\sigma} = 27.68.$$

The sample mean of \$74.29 serves as our **point estimate** for the true average order value of all customers. Similarly, the sample standard deviation of \$27.68 estimates the variability in order amounts across the population.

0.5 Interval Estimation: Confidence Intervals

"We are 95% confident that the true average height lies between 172 cm and 178 cm."

What Influences the Width of a Confidence Interval?

The width of a confidence interval reflects the precision of our estimate. Narrow intervals indicate high precision, while wide intervals show greater uncertainty. Three main factors influence this width:

1. **Confidence Level:** Higher confidence levels require wider intervals. For example, a 99% confidence interval will be wider than a 95% interval.
2. **Sample Size (n):** Larger samples provide more information and lead to more precise estimates, resulting in narrower intervals.
3. **Variability in the Data:** If the data have high variability (large standard deviation), the interval must be wider to account for this uncertainty.

Confidence intervals provide a richer and more informative summary than point estimates alone. They offer a realistic range of plausible values for the population parameter and explicitly communicate uncertainty. This concept is fundamental in statistics and plays a major role in evaluating model performance and reliability in machine learning.

0.6 Confidence Intervals

A point estimate, such as the sample mean $\bar{x}$, gives a single best guess for an unknown population parameter like the true mean μ. However, point estimates do not express how uncertain that guess is. If we were to draw a different sample from the same population, the resulting estimate would likely differ. To properly reflect this uncertainty, we use **interval estimation**.

A **confidence interval** provides a range of plausible values for an unknown population parameter. Instead of stating only one number (e.g., $\bar{x} = 175$ cm), we might say:

> We are 95% confident that the true mean lies between 172 and 178 cm.

This range is the confidence interval, and the "95%" indicates the **confidence level**. A 95% confidence level means that if we were to repeatedly take samples of the same size and build a confidence interval from each sample, about 95% of those intervals would contain the true population mean.

0.6.1 What Influences the Width of a Confidence Interval?

The width of a confidence interval reflects how precise our estimate is. Three primary factors influence it:

- **Confidence Level**: A higher confidence level (e.g., 99%) produces a wider interval.
- **Sample Size**: Larger samples give narrower intervals because they yield more precise estimates.
- **Data Variability**: Higher standard deviation in the data leads to wider intervals.

Confidence intervals offer richer information than point estimates alone by providing both an estimate and a measure of uncertainty.

0.6.2 Formulas for Confidence Intervals for the Mean

The appropriate formula depends on whether the population standard deviation σ is known.

the Population Standard Deviation is Known

The $100(1 - \alpha)\%$ confidence interval for μ is:

$$\bar{x} \pm z_{\alpha/2} \cdot \frac{\sigma}{\sqrt{n}}$$

where $z_{\alpha/2}$ is the critical value from the standard normal distribution.

Population Standard Deviation is Unknown

In practice, σ is rarely known. We use the sample standard deviation s and the t-distribution:

$$\bar{x} \pm t_{\alpha/2, n-1} \cdot \frac{s}{\sqrt{n}}$$

where $t_{\alpha/2, n-1}$ is the critical value from the t-distribution with $n - 1$ degrees of freedom.

0.6.3 Example: Confidence Interval for Average Order Value

Suppose an e-commerce company wants to estimate the true average order value (AOV) for all customers. Instead of analyzing the entire population, we take a sample of 50 recent orders and compute a 95% confidence interval for the mean.

```
import numpy as np
from scipy import stats

# Sample of 50 recent order values (in dollars)
order_values = np.array([
    55.2, 34.1, 89.0, 120.5, 42.0, 65.8, 22.1, 98.7, 77.3,
       105.6,
    30.9, 48.2, 71.5, 60.0, 112.8, 58.4, 88.2, 45.1, 99.9,
       75.0,
    135.2, 28.6, 50.0, 68.9, 91.3, 102.1, 40.5, 79.8, 66.2,
       84.7,
    115.0, 33.5, 52.8, 70.1, 95.4, 108.3, 47.9, 62.7, 81.6,
       100.2,
    36.8, 56.7, 73.9, 64.3, 89.1, 110.5, 49.6, 67.0, 83.3,
       97.4
```

```
])

# Sample size
n = len(order_values)

# Sample mean and sample std deviation
sample_mean = np.mean(order_values)
sample_std = np.std(order_values, ddof=1)

# Confidence level
confidence_level = 0.95
alpha = 1 - confidence_level

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

# Margin of error
margin_of_error = t_crit * sample_std / np.sqrt(n)

# Confidence interval
ci_lower = sample_mean - margin_of_error
ci_upper = sample_mean + margin_of_error

print(f"Sample size (n): {n}")
print(f"Sample mean (AOV): ${sample_mean:.2f}")
print(f"Sample standard deviation: ${sample_std:.2f}")
print(f"95% Confidence Interval: [${ci_lower:.2f}, ${ci_upper
    :.2f}]")
```

Listing 1: Confidence Interval for Mean AOV in Python

0.6.4 Python Output

The Python script produces the following results:

- Sample size: $n = 50$
- Sample mean: $\bar{x} = 72.87$

- Sample standard deviation: $s = 27.21$
- 95% Confidence Interval:

$$[\,65.14,\ 80.61\,]$$

We are 95% confident that the true average order value (AOV) in the population lies between \$65.14 and \$80.61. This interval reflects natural sampling uncertainty and provides much more information than the point estimate alone.

0.7 Hypothesis Testing: The Basic Idea

Statistical inference is not only about estimating unknown population characteristics; it sometimes requires making a specific decision about a claim regarding a population. While point estimates and confidence intervals tell us what the sample suggests, hypothesis testing provides a formal framework for evaluating whether sample evidence supports or contradicts a stated assumption.

A helpful analogy is that of a legal trial. In a courtroom, there is an initial assumption—the defendant is considered innocent. Evidence is then presented, and depending on its strength, the jury either maintains this assumption or rejects it. Hypothesis testing works in much the same way. We begin with a default assumption about a population parameter, called the **null hypothesis**. Opposing it is the **alternative hypothesis**, which represents the claim we wish to evaluate.

The null hypothesis typically reflects a baseline or "no effect" statement. Examples :

- The average load time of a website is 200 milliseconds.
- A new treatment performs the same as the current standard.
- The proportion of users who click on an ad is 5%.

After defining the hypotheses, we collect sample data and ask a central question:

> *If the null hypothesis were true, how likely would it be to observe data like ours—or more extreme—purely by random variation?*

If the observed data would be reasonably likely under the null hypothesis, the evidence is not strong enough to contradict it, and we **fail to reject** the null. This outcome does not confirm that the null hypothesis is true; rather, it indicates that the available evidence does not justify overturning it.

Hypothesis testing therefore turns sample data into a structured decision-making process, helping us determine whether to retain the initial assumption or reject it in favor of a more supported alternative.

- **Null Hypothesis (H_0):** The new design does not increase session duration. Formally, the average duration is still less than or equal to 3 minutes.
- **Alternative Hypothesis (H_1):** The new design leads to longer sessions, meaning the true average duration is greater than 3 minutes.

The central question becomes:

> *If the true average session duration were still* 3 *minutes (or less), how likely is it that we would obtain a sample mean as large as* 4.5 *minutes purely due to random variation?*

In the following sections, we will formalize this procedure by clearly defining the null and alternative hypotheses and introducing the concept of the *p-value*, a central tool used to guide inferential decisions.

0.8 Null and Alternative Hypotheses

Hypothesis testing provides a structured approach for using sample data to evaluate competing claims about a population. The process begins by formulating two statements: one that represents the current assumption or baseline belief, and another that represents a possible change or difference. These are known as the *null hypothesis* and the *alternative hypothesis*.

0.8.1 The Null Hypothesis (H_0)

The null hypothesis, denoted H_0, represents the default position or status quo. It typically states that there is "no effect," "no difference," or "no change." Before examining the data, we assume that H_0 is true.

Mathematically, null hypotheses usually involve an equality condition ($=$, $\leq$, $\geq$). Examples include:

- **Website Design:** If a company tests a new website layout, the null hypothesis may state that the average time users spend on the new layout is the same as on the old one:
$$H_0 : \mu_{\text{new}} = \mu_{\text{old}}.$$
- **Model Performance:** For a new machine learning model, the null hypothesis may claim that it performs no better than the existing model:
$$H_0 : E_{\text{new}} \geq E_{\text{old}}.$$

The null hypothesis serves as the benchmark against which we compare our sample evidence.

0.8.2 The Alternative Hypothesis (H_1

The alternative hypothesis, denoted H_1, contradicts the null hypothesis. It represents what we suspect or hope might be true instead of H_0. While H_0 contains an equality, the alternative hypothesis always involves a strict inequality ($\neq$, $>$, $<$).

Corresponding examples include:

- **Website Design:** If we suspect any change in user engagement, the alternative is:
$$H_1 : \mu_{\text{new}} \neq \mu_{\text{old}}.$$
If we specifically believe the new design increases engagement:
$$H_1 : \mu_{\text{new}} > \mu_{\text{old}}.$$

- **Model Performance:** If the goal is to show the new model has a lower prediction error:

$$H_1 : E_{\text{new}} < E_{\text{old}}.$$

0.8.3 One-Tailed vs. Two-Tailed Tests

The form of the alternative hypothesis determines the type of hypothesis test:

- **Two-tailed test:** when

$$H_1 : \mu \neq \mu_0,$$

  used when deviations in either direction are of interest.

- **One-tailed test:** when

$$H_1 : \mu > \mu_0 \quad \text{(right-tailed)} \qquad \text{or} \qquad H_1 : \mu < \mu_0 \quad \text{(left-tailed)}.$$

  This is used when the research question focuses on a specific direction of change.

The choice between a one-tailed and two-tailed test must be made before examining the data.

0.9 Understanding P-values

In hypothesis testing, we compare two competing statements about a population: the null hypothesis (H_0), which represents the baseline or "no effect" assumption, and the alternative hypothesis (H_1), which reflects the possibility of a meaningful effect or difference. Once these hypotheses are established, the key question becomes: *does the sample data provide enough evidence to cast doubt on H_0?*

The p-value provides a formal way to quantify this evidence. It answers the question:

> *If the null hypothesis H_0 were true, what is the probability of observing a result as extreme as (or more extreme than) the one we obtained from our sample?*

Interpreting a p-value relies on this idea of "surprise." A very small p-value means that the observed data would be unusual if H_0 were true, while a large p-value suggests that the data are consistent with H_0.

Interpreting the P-value

Because a p-value is a probability, it always lies between 0 and 1.

- **Small p-value (typically ≤ 0.05):** The observed data would be unlikely if H_0 were true. This provides evidence *against* the null hypothesis and supports the alternative hypothesis H_1.
- **Large p-value (> 0.05):** The observed data are reasonably consistent with the null hypothesis. We do not have sufficient evidence to reject H_0.

The Significance Level α

To decide whether a p-value is "small," we compare it to a threshold called the *significance level*, denoted by α. This value is chosen before performing the test. A common choice is $\alpha = 0.05$, although $\alpha = 0.01$ or $\alpha = 0.10$ may also be used depending on context.

The decision rule is:

- If $p \leq \alpha$: **Reject** H_0. There is statistically significant evidence supporting H_1.
- If $p > \alpha$: **Fail to reject** H_0. There is not enough statistical evidence to support H_1.

Example: A/B Testing for Website Conversion

Suppose we compare two website designs and want to determine whether the new design increases the conversion rate.

- H_0: The new design does not increase the conversion rate (rate is less than or equal to the old one).
- H_1: The new design increases the conversion rate.
- Significance level: $\alpha = 0.05$.

After running an experiment, the statistical test produces a p-value.

Scenario 1: p-value = 0.02

- If H_0 were true, observing an improvement as large as the one in our sample would occur only 2% of the time by random chance.
- Since $0.02 \leq 0.05$, we **reject** H_0.
- There is statistically significant evidence that the new design increases conversions.

Scenario 2: p-value = 0.31

- A result this large is not surprising if H_0 were true; it could easily occur due to random variation.
- Since $0.31 > 0.05$, we **fail to reject** H_0.
- The data do not provide enough evidence to conclude that the new design improves conversion rates.

Important Clarifications

Understanding what a p-value represents is essential:

- A p-value is **not** the probability that the null hypothesis is true.
- A p-value is calculated **assuming H_0 is true**.
- Failing to reject H_0 does **not** prove it true; it simply means the evidence is insufficient.

- Statistical significance does **not** imply practical significance. Even tiny differences may become statistically significant with very large samples.

P-values provide a standardized way to evaluate the strength of evidence against a null hypothesis. They are widely used in statistics, machine learning evaluation, and A/B testing to support data-driven decisions.

0.10 Connecting Inference to Machine Learning Evaluation

Statistical inference plays a central role in evaluating machine learning models. When we compute performance metrics such as accuracy, precision, recall, or mean squared error (MSE), we are not observing the true performance of the model on all future data. We are observing a *sample-based estimate*: the model's performance on a finite test set.

Thus:

- A test-set metric is a **point estimate** of the model's true performance.
- A **confidence interval** quantifies uncertainty around this estimate.
- **Hypothesis testing** allows us to determine whether apparent performance differences between models are statistically meaningful or simply due to sampling variability.

For example, if a classifier achieves 92% accuracy on a test set, this value is the sample estimate $\hat{p}$ of the true accuracy p. A different test sample would produce a slightly different value. A 95% confidence interval may report:

$$\text{True accuracy lies in } [0.89, 0.95] \text{ with } 95\% \text{ confidence.}$$

This communicates both the estimate and its uncertainty, which is essential for reliable ML evaluation.

0.10.1 Confidence Intervals for Machine Learning Metrics

1. Confidence Interval for Accuracy If a model achieves accuracy $\hat{p}$ on n test examples, an approximate 95% confidence interval for the true accuracy p is:

$$\hat{p} \pm 1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}.$$

This is derived from the normal approximation to the binomial distribution.

2. Confidence Interval for Mean Squared Error (MSE) If the test errors are $e_1, e_2, \ldots, e_n$, then

$$\widehat{\text{MSE}} = \frac{1}{n}\sum_{i=1}^{n} e_i^2.$$

A 95% confidence interval (using Central Limit Theorem) is:

$$\widehat{\text{MSE}} \pm 1.96\frac{s_e}{\sqrt{n}},$$

where s_e is the sample standard deviation of the squared errors.

3. Confidence Interval for Precision or Recall These are proportions and use the same formula as accuracy.

0.11 Practice: Interpreting Statistical Results

Statistical inference allows us to move beyond simply describing the data we collected. It gives us tools to make informed statements about a broader population and to evaluate whether the patterns we observe in a sample are meaningful. The goal is not only to compute numbers such as means, confidence intervals, or p-values, but also to *interpret them correctly in context.*

In this section, we explore three practical scenarios that mirror the machine learning practitioners make in practice. Each example illustrates how point estimates,

confidence intervals, and hypothesis tests should be understood.

0.11.1 Scenario 1: Estimating Website Conversion Rate

Suppose you run an A/B test on an e-commerce platform:

- Group A: 1000 visitors see the old checkout page.
- Group B: 1000 visitors see a new checkout page.

In Group B, 55 out of 1000 visitors make a purchase. The results are:

- **Point estimate (conversion rate for Group B):**

$$\hat{p} = \frac{55}{1000} = 0.055 \quad (5.5\%)$$

- **95% confidence interval for the true conversion rate:**

$$[0.041,\ 0.069] \quad (4.1\% \text{ to } 6.9\%).$$

Interpretation:

1. **Point estimate:** The value 0.055 is our single best estimate of the true conversion rate for all visitors if they were all shown the new design. It is computed directly from the sample (55 conversions out of 1000 visitors).
2. **Confidence interval:** The interval $[0.041, 0.069]$ gives a plausible range of values for the true population conversion rate.
    - Being "95% confident" refers to the *procedure* used to construct the interval. If we repeated the experiment many times and built a 95% confidence interval each time, about 95% of those intervals would contain the true conversion rate.
    - The width of the interval reflects uncertainty: a wider interval indicates more uncertainty, while a narrower interval indicates a more precise estimate.

0.11.2 Scenario 2: Testing the Impact of a Feature Change

A streaming service introduces a new recommendation algorithm and wants to know whether it increases user engagement, measured as the average number of videos watched per user per week.

Hypotheses

Let μ_{old} be the true average number of videos watched per week under the old algorithm, and μ_{new} under the new one.

- **Null hypothesis:**

  $$H_0 : \mu_{\text{new}} \leq \mu_{\text{old}}$$

  There is no increase in engagement.

- **Alternative hypothesis:**

  $$H_1 : \mu_{\text{new}} > \mu_{\text{old}}$$

  The new algorithm increases the average number of videos watched.

You choose a significance level of $\alpha = 0.05$.

Sample Results

The analysis of the sample data yields:

- Sample average (old algorithm): 8.3 videos/week.
- Sample average (new algorithm): 8.7 videos/week.
- p-value: 0.007.

Interpretation

1. **Compare p-value to α:** The p-value is 0.007, which is less than $\alpha = 0.05$.

2. **Decision:** Since $p < \alpha$, we reject the null hypothesis H_0.

3. **Conclusion:** There is statistically significant evidence that the new recommendation algorithm increases the average number of videos watched per user per week. The observed difference (8.9 vs. 8.2) is unlikely to have occurred purely by random chance if the new algorithm were truly no better than the old one.

What if the p-value were 0.15?

If the p-value had been 0.15 (which is greater than 0.05), our conclusion would change:

1. **Compare p-value to α:** $0.15 > 0.05$.

2. **Decision:** We would *fail to reject* the null hypothesis H_0.

3. **Conclusion:** The sample average for the new algorithm (8.9) is still higher than for the old algorithm (8.2), but the evidence is not strong enough to claim a real improvement. The observed difference could reasonably be due to sampling variation. We are not proving that the algorithms are identical; rather, the data do not provide strong evidence that the new algorithm is better at the chosen significance level.

0.11.3 Scenario 3: Comparing Machine Learning Model Performance

You train two classification models (Model A and Model B) to predict customer churn and evaluate both on the same test set.

- Model A test accuracy: 88%.
- Model B test accuracy: 90%.

You want to know whether this 2% difference reflects a real performance gap or could just be due to the particular test set used.

Hypotheses

Let Acc_A and Acc_B denote the true accuracies of Model A and Model B, respectively.

- **Null hypothesis:**

$$H_0 : \text{Acc}_A = \text{Acc}_B$$

  The models have the same true accuracy.

- **Alternative hypothesis:**

$$H_1 : \text{Acc}_A \neq \text{Acc}_B$$

  The models have different true accuracies.

You choose a significance level $\alpha = 0.05$, and use an appropriate test (for example, McNemar's test, since both models are evaluated on the same set of instances). The result of the test is:

- p-value: 0.21.

Python Example: Computing a Confidence Interval and a p-value

In this subsection, we illustrate how to:

- Compute a confidence interval for a proportion (Scenario 1: website conversion rate).
- Perform a hypothesis test and obtain a p-value when comparing two groups (Scenario 2: old vs. new recommendation algorithm).

Confidence Interval for a Conversion Rate (Scenario 1)

We observed 55 purchases out of 1000 visitors who saw the new checkout design. We can compute the point estimate $\hat{p}$ and a 95% confidence interval using the normal approximation:

$$\hat{p} \pm z_{\alpha/2}\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \quad \text{with } z_{0.025} \approx 1.96.$$

The following Python code uses `NumPy` and `SciPy` to compute this interval:

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

# Scenario 1: Conversion rate for new checkout design
n = 1000           # sample size
x = 55             # number of conversions
phat = x / n       # point estimate for conversion rate

alpha = 0.05
z = stats.norm.ppf(1 - alpha / 2)  # 1.96 for 95% CI

# Standard error for a proportion
se = np.sqrt(phat * (1 - phat) / n)

# Confidence interval
ci_low = phat - z * se
ci_high = phat + z * se

print(f"Point estimate (p_hat): {phat:.3f}")
print(f"95% CI: ({ci_low:.3f}, {ci_high:.3f})")
```

A typical output is:

```
Point estimate (p_hat): 0.055
95% CI: (0.041, 0.069)
```

This matches our earlier interpretation: the estimated conversion rate is about 5.5%, and we are 95% confident that the true conversion rate lies between approximately 4.1% and 6.9%.

Hypothesis Test and p-value for Two Means (Scenario 2)

Next, we illustrate how to compare the average engagement (videos watched per user per week) between the old and new recommendation algorithms using a two-sample t-test.

Below, we simulate sample data for two groups of users (old vs. new algorithm), then compute the sample means and perform a t-test using `scipy.stats.ttest_ind`:

```
import numpy as np
from scipy import stats

np.random.seed(0)

# Scenario 2: Simulated engagement data (videos watched per
    week)
n_old = 200
n_new = 200

# Simulate data: old algorithm around 8.2 videos, new around
    8.9 videos
# with some variability (standard deviation 1.5)
old_engagement = np.random.normal(loc=8.2, scale=1.5, size=
    n_old)
new_engagement = np.random.normal(loc=8.9, scale=1.5, size=
    n_new)

# Sample means (point estimates)
mean_old = old_engagement.mean()
mean_new = new_engagement.mean()

print(f"Sample mean (old): {mean_old:.2f}")
print(f"Sample mean (new): {mean_new:.2f}")

# Two-sample t-test (Welch's t-test, unequal variances
    allowed)
t_stat, p_value = stats.ttest_ind(new_engagement,
    old_engagement,
                                  equal_var=False)
```

```python

print(f"t-statistic: {t_stat:.4f}")
print(f"p-value: {p_value:.4f}")

alpha = 0.05
if p_value < alpha:
    print("Conclusion: Reject H0 - evidence that the new algorithm "
          "changes average engagement.")
else:
    print("Conclusion: Fail to reject H0 - no strong evidence of a difference.")
```

One possible output is:

```
Sample mean (old): 8.31
Sample mean (new): 8.71
t-statistic: 2.7166
p-value: 0.0069
Conclusion: Reject H0 - evidence that the new algorithm changes average engagement.
```

In this simulated example:

- The new algorithm shows a higher sample average (about 8.71 vs. 8.31 videos).
- The p-value is approximately 0.0069, which is less than $\alpha = 0.05$.
- Therefore, we reject the null hypothesis and conclude that there is statistically significant evidence that the new algorithm changes the average engagement.

These Python examples show how to:

- Turn raw counts into a point estimate and confidence interval for a proportion.
- Use a hypothesis test (via a t-test) to obtain a p-value and make a decision about whether a difference between two groups is statistically significant.

Interpretation

1. **Compare p-value to α:** $0.21 > 0.05$.
2. **Decision:** We fail to reject the null hypothesis H_0.
3. **Conclusion:** Although Model B achieved higher accuracy (90%) than Model A (88%) on this test set, the difference is not statistically significant at the 0.05 level. There is not enough evidence to conclude that Model B would consistently outperform Model A on new data from the same distribution. The 2% gap could be due to random variation in which examples happened to be included in the test set.

These scenarios illustrate how to interpret point estimates, confidence intervals, and p-values in real decision-making contexts. Such interpretations are crucial when evaluating experiments, choosing between algorithms, or communicating results in data analysis and machine learning.

0.11.4 Comparing Models with Hypothesis Testing

If Model A and Model B are evaluated on the same test set, we can test whether one truly outperforms the other.

Two common approaches:

- **Paired t-test**: works when comparing continuous errors (e.g., regression).
- **McNemar's Test or Binomial Test**: appropriate when comparing classification correctness outcomes.

Python Example 1: Paired t-test for Regression or Probabilities

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

```

```python
# Simulated correctness on 200 test samples
np.random.seed(0)
model_A_correct = np.random.binomial(1, 0.85, size=200)
model_B_correct = np.random.binomial(1, 0.87, size=200)

# Compute accuracy
acc_A = model_A_correct.mean()
acc_B = model_B_correct.mean()

print(f"Model A Accuracy: {acc_A:.3f}")
print(f"Model B Accuracy: {acc_B:.3f}")

# Paired t-test
t_stat, p_value = ttest_rel(model_B_correct, model_A_correct)

print(f"T-statistic: {t_stat:.4f}")
print(f"P-value: {p_value:.4f}")
```

Python Example 2: Binomial Test (More Accurate for Comparing Accuracies)

The binomial test examines how often Model B outperforms Model A on the *same* samples.

```python
from scipy.stats import binomtest

# Count cases where B is correct and A is wrong
wins_B = np.sum((model_B_correct == 1) & (model_A_correct == 0))

# Count total disagreements
total_disagreements = np.sum(model_A_correct != model_B_correct)

result = binomtest(wins_B, total_disagreements, p=0.5, alternative='greater')

```

```
print(result)
```

Interpretation: - If the p-value is small, Model B is significantly better. - If the p-value is large, the models are statistically indistinguishable.

0.11.5 Exercises

1. A classifier achieves accuracy 0.91 on 500 samples. Compute the 95% confidence interval.

2. Two models have accuracies of 0.88 and 0.90 on the same test set of 300 samples. Which statistical test should you use to determine if the difference is statistically significant? Justify your choice.

3. A huge dataset leads to a p-value of 10^{-12} when comparing two models whose accuracies differ by only 0.5%. Explain why this happens and why practical significance may differ from statistical significance.

4. Implement a binomial confidence interval for accuracy using Python. Compare the normal approximation interval with the Wilson score interval.

0.12 Inference in Simple Linear Regression

Linear regression is one of the most widely used tools in statistics and machine learning for modeling relationships between numerical variables. Beyond prediction, linear regression allows us to perform *statistical inference*: we can estimate model parameters, build confidence intervals, and test whether predictors have a meaningful effect on the response.

In this section, we focus on *simple* linear regression with one predictor.

0.12.1 The Simple Linear Regression Model

We assume the response variable Y and predictor X follow the model:

$$Y = \beta_0 + \beta_1 X + \varepsilon, \tag{1}$$

where

- β_0 is the intercept,
- β_1 is the slope (effect of X on Y),
- ε is random noise with mean 0.

Given a sample of paired data (x_i, y_i), we use least squares to estimate the parameters:

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X.$$

0.12.2 Interpreting the Slope

The slope $\hat{\beta}_1$ tells us how much Y changes, on average, for a one-unit increase in X.

Example (Apartment Rent). Let

- Y = monthly rent of an apartment (in euros),
- X = size of the apartment (in m^2).

Suppose a regression yields:

$$\hat{Y} = 320 + 18.5\, X.$$

Interpretation:

- $\hat{\beta}_0 = 320$: the model predicts a baseline rent of €320 for an apartment of size 0 m^2 (not realistic but part of the linear model).
- $\hat{\beta}_1 = 18.5$: each additional m^2 increases the expected rent by €18.50.

0.12.3 Confidence Interval for the Slope

A $(1-\alpha) \times 100\%$ confidence interval for the true slope β_1 is:

$$\hat{\beta}_1 \pm t_{\alpha/2,df} \cdot SE(\hat{\beta}_1).$$

This interval provides a range of plausible values for the true slope. If the interval does not include 0, there is evidence of a linear association between X and Y.

0.12.4 Hypothesis Test for the Slope

To test whether the predictor X has a real linear effect on Y, we test:

$$H_0 : \beta_1 = 0 \qquad \text{vs.} \qquad H_1 : \beta_1 \neq 0.$$

Regression output provides:

- $\hat{\beta}_1$ (estimated slope),
- $SE(\hat{\beta}_1)$,
- a t-statistic and p-value.

A small p-value leads us to reject H_0 and conclude that X significantly influences Y.

Python Example: Regression on Apartment Rent and Size

The following Python example simulates a relationship between apartment rent and size, then performs inference on the slope:

```
import numpy as np
import statsmodels.api as sm

np.random.seed(0)

# Simulate apartment sizes (in m^2)
n = 120
sizes = np.random.uniform(20, 120, size=n)

# True underlying model (in euros)
true_intercept = 300     # baseline rent
true_slope = 20          # rent increase per m^2
```

```
sigma = 60                # noise level

# Generate monthly rent with noise
noise = np.random.normal(0, sigma, size=n)
rents = true_intercept + true_slope * sizes + noise

# Prepare data for regression
X = sm.add_constant(sizes)
y = rents

model = sm.OLS(y, X)
results = model.fit()

print(results.summary())
```

A typical output (simplified) might look like:

```
                 coef    std err          t      P>|t|    [0.025    0.975]
------------------------------------------------------------------------------
const        315.284     21.332     14.78      0.000    272.99    357.57
x1            19.872      0.268     74.08      0.000     19.34     20.40
```

Interpretation:

- The estimated slope is $\hat{\beta}_1 \approx 19.87$, very close to the true slope of 20.
- The 95% confidence interval is approximately $[19.34,\ 20.40]$.
- The p-value for the slope is extremely small (< 0.001), providing strong evidence that apartment size has a significant positive effect on rent.

This example illustrates how regression naturally integrates:

- **Point estimation** (estimates of β_0 and β_1),
- **Confidence intervals** for parameters,
- **Hypothesis testing** to determine if predictors have meaningful effects.

In a machine learning context, such inference helps evaluate and interpret the importance and reliability of model features.