

How do you calculate point estimates in R? Can you provide any examples?

Authored by
stats writer

May 4, 2024

RECOMMENDED CITATION

stats writer (2024). *How do you calculate point estimates in R? Can you provide any examples?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=142715>

Point estimates in R are calculated by using statistical methods to estimate a single value or parameter from a larger population. This involves using a sample of data to make an inference about the population. In R, point estimates can be calculated using various functions such as `mean()`, `median()`, and `mode()`. For example, to calculate the mean of a sample data set, the `mean()` function can be used. The resulting value would be the point estimate of the population mean. Other examples of point estimates in R include estimating the population proportion, variance, and standard deviation. Point estimates are commonly used in data analysis to make predictions and draw conclusions about a population based on a sample.

Calculate Point Estimates in R (With Examples)

A point estimate represents a number that we calculate from sample data to estimate some population parameter. This serves as our best possible estimate of what the true population parameter may be.

The following table shows the point estimate that we use to estimate population parameters:

Measurement	Population parameter	Point estimate
Mean	μ (population mean)	\bar{x} (sample mean)
Proportion	π (population proportion)	p (sample proportion)

The following examples explain how to calculate point estimates for a population mean and a population proportion in R.

Example 1: Point Estimate of Population Mean

Suppose we would like to estimate the mean height (in

inches) of a certain type of plant in a certain field. We gather a simple random sample of 13 plants and measure the height of each plant.

The following code shows how to calculate the sample mean:

```
#define data
```

```
data <- c(8, 8, 9, 12, 13, 13, 14, 15, 19, 22, 23, 23, 24)
```

```
#calculate sample mean
```

```
mean(data, na.rm = TRUE)
```

```
15.61538
```

The sample mean is 15.6 inches. This represents our point estimate for the population mean.

We can also use the following code to calculate a 95% confidence interval for the population mean:

```
#find sample size, sample mean, and sample standard deviation
```

```
n <- length(data)
```

```
xbar <- mean(data, na.rm = TRUE)
```

```
s <- sd(data)
```

```
#calculate margin of error
```

```
margin <- qt(0.975,df=n-1)*s/sqrt(n)
```

```
#calculate lower and upper bounds of confidence interval
```

```
low <- xbar - margin
```

```
low
```

```
12.03575
```

```
high <- xbar + margin
```

```
high
```

```
19.19502
```

The 95% confidence interval for the population mean is inches.

Example 2: Point Estimate of Population Proportion

Suppose we would like to estimate the proportion of people in a certain city that support a certain law. We survey a simple random sample of 20 citizens.

The following code shows how to calculate the sample proportion:

```
#define data
```

```
data <- c('Y', 'Y', 'Y', 'N', 'N', 'Y', 'Y', 'Y', 'N', 'Y',  
'N', 'Y', 'Y', 'N', 'N', 'Y', 'Y', 'Y', 'N', 'N')
```

```
#find total sample size
```

```
n <- length(data)
```

```
#find number who responded 'Yes'
```

```
k <- sum(data == 'Y')
```

```
#find sample proportion
```

```
p <- k/n
```

```
p
```

```
0.6
```

The sample proportion of citizens who support the law is 0.6. This represents our point estimate for the population proportion.

```
#find total sample size
```

```
n <- length(data)
```

```
#find number who responded 'Yes'
```

```
k <- sum(data == 'Y')
```

```
#find sample proportion
```

```
p <- k/n
```

```
#calculate margin of error
```

```
margin <- qnorm(0.975)*sqrt(p*(1-p)/n)
```

```
#calculate lower and upper bounds of confidence interval
```

```
low <- p - margin
```

```
low
```

```
0.3852967
```

```
high <- p + margin
```

```
high
```

```
0.8147033
```

The 95% confidence interval for the population proportion is .