Distributions

Arif_Furkan

2024-06-29

Functions for distributions

In this project, we see how statistical distributions are implemented in R and finally we apply the central limit theorem.

Uniform Distribution

```
x<-dunif(0.5, min = 0, max = 1)
dunif(x, min = 0, max = 1, log = FALSE)
```

```
## [1] 1
```

```
q<-punif(0.5, min = 0, max = 1)
punif(q, min = 0, max = 1, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0.5
```

```
p<-qunif(0.5, min = 0, max = 1)
qunif(p, min = 0, max = 1, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0.5
```

```
n<-runif(10, min = 0, max = 1)
runif(n, min = 0, max = 1)
```

```
##  [1] 0.09951053 0.17927398 0.88938915 0.43771345 0.73940020 0.47462938
##  [7] 0.77872455 0.28526973 0.13050311 0.27091801
```

Binomial Distribution

These probability and size values were chosen randomly.

```
size<-5
prob<-0.5
dbinom(x, size, prob, log = FALSE)
```

```
## [1] 0.15625
```

```r
pbinom(q, size, prob, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0.03125
```

```r
qbinom(p, size, prob, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 2
```

```r
rbinom(n, size, prob)
```

```
##  [1] 1 3 1 4 1 1 2 1 2 3
```

Normal Distribution

This mean value was chosen randomly.

```r
mean<-0
dnorm(x, mean = 0, sd = 1, log = FALSE)
```

```
## [1] 0.2419707
```

```r
pnorm(q, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0.6914625
```

```r
qnorm(p, mean = 0, sd = 1, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0
```

```r
rnorm(n, mean = 0, sd = 1)
```

```
##  [1] -0.2017029  0.5652956  1.4270631 -1.4291748 -1.1630644 -0.2729076
##  [7] -0.9731021  0.5677152  1.7712150  1.2320155
```

Hypergeometric Distribution

These m, k and e values were chosen randomly.

```r
m<-10
k<-7
e<-5
dhyper(x, m, e, k, log = FALSE)
```

```
## [1] 0
```

```r
phyper(q, m, e, k, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0
```

```r
qhyper(p, m, e, k, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 5
```

```r
rhyper(n, m, e, k)
```

```
##  [1] 5 5 3 5 4 3 4 3 4 5
```

Poisson Distribution

This lambda value was chosen randomly.

```r
lambda<-2
dpois(x, lambda, log = FALSE)
```

```
## [1] 0.2706706
```

```r
ppois(q, lambda, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0.1353353
```

```r
qpois(p, lambda, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 2
```

```r
rpois(n, lambda)
```

```
##  [1] 1 1 1 1 0 1 1 2 3 2
```

Geometric Distribution

```r
dgeom(x, prob, log = FALSE)
```

```
## [1] 0.25
```

```r
pgeom(q, prob, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0.5
```

```r
qgeom(p, prob, lower.tail = TRUE, log.p = FALSE)
```

```
## [1] 0
```

```r
rgeom(n, prob)
```

```
##  [1] 0 1 0 0 0 1 0 3 2 0
```

Central Limit Theorem

Load necessary packages

```r
require(ggplot2)
```

```
## Zorunlu paket yükleniyor: ggplot2
```

Create a population from a Poisson distribution

```r
lambda <- 5  # Parameter for Poisson distribution (mean)
population <- rpois(10000, lambda)  # Population data
```

Sample size and number of simulations

```r
n <- 30  # Sample size
num_simulations <- 1000  # Number of simulations
```

Calculate the sample means

```r
sample_means <- replicate(num_simulations, {
sample <- sample(population, n, replace = TRUE)
mean(sample)})
```

Save the sample means as a data frame

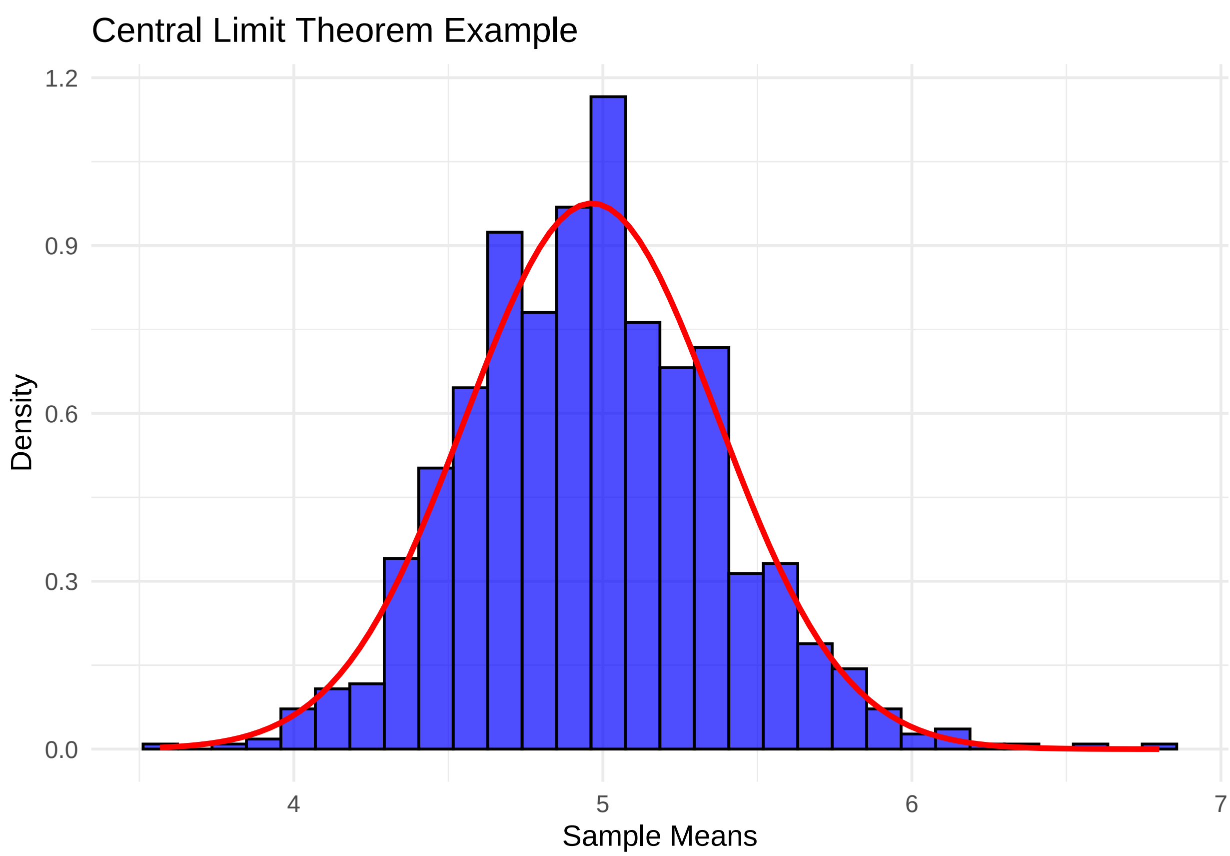
```r
sample_means_df <- data.frame(sample_means)
```

Plot the histogram of sample means

```
ggplot(sample_means_df, aes(x = sample_means)) +
geom_histogram(aes(y = ..density..), bins = 30, fill = "blue", color = "black", alpha = 0.7) +
stat_function(fun = dnorm, args = list(mean = mean(sample_means), sd = sd(sample_means)), color = "red"
labs(title = "Central Limit Theorem Example",
x = "Sample Means",
y = "Density") +
theme_minimal()
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```

```
## Warning: The dot-dot notation ('..density..') was deprecated in ggplot2 3.4.0.
## i Please use 'after_stat(density)' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```



This example demonstrates the Central Limit Theorem, showing how the distribution of sample means approaches a normal distribution as the sample size increases.