

Mini Homework

Nora Alquézar

Complex Data Analysis, 2023-2024

Introduction

This report outlines the analysis of insect counts across various sprays using the `InsectSprays` dataset in R. The study employs bootstrap methods to compute the dispersion index, fits a Poisson GLM model to the data, and evaluates the hypothesis of equidispersion.

Step 1: Load the Dataset

The `InsectSprays` dataset is loaded using the `data()` function in R.

```
# Load the InsectSprays dataset  
data(InsectSprays)  
head(InsectSprays)
```

Dataset output:

	count	spray
1	10	A
2	7	A
3	20	A
4	14	A
5	14	A
6	12	A

Step 2: Calculate the Dispersion Index Using Bootstrap

The dispersion index is calculated for each spray group using bootstrap methods.

- Spray A:
 - Percentile method: (0.564, 2.437)
 - Basic method: (0.635, 2.508)
 - Bootstrap-t method: (0.854, 3.016)
- Spray B:
 - Percentile method: (0.371, 1.939)
 - Basic method: (0.440, 2.008)
 - Bootstrap-t method: (0.600, 2.655)
- Spray C:
 - Percentile method: (0.545, 3.007)
 - Basic method: (0.738, 3.200)
 - Bootstrap-t method: (0.822, 3.842)
- Spray D:
 - Percentile method: (0.134, 2.316)
 - Basic method: (0.232, 2.414)
 - Bootstrap-t method: (0.222, 3.009)
- Spray E:
 - Percentile method: (0.3355, 1.2991)
 - Basic method: (0.4152, 1.3788)
 - Bootstrap-t method: (0.4757, 1.5424)
- Spray F:
 - Percentile method: (0.992, 3.128)
 - Basic method: (1.505, 3.641)
 - Bootstrap-t method: (1.591, 3.606)

Step 3: Fit a Poisson GLM Model

A Poisson GLM model is fitted, and the dispersion index is calculated.

```
# Fit a Poisson GLM model
poisson_glm <- glm(count ~ spray, data = InsectSprays,
family = poisson)

# Calculate the dispersion index
pearson_residuals <- residuals(poisson_glm, type = "pearson")
dispersion_index_glm <- sum(pearson_residuals^2)
/ poisson_glm$df.residual
dispersion_index_glm
```

Dispersion Index from the Poisson GLM Model:

1.507713

Step 4: Bootstrap Test for Equidispersion

A bootstrap test is performed to evaluate the null hypothesis of equidispersion.

```
# Define a function to test for overdispersion
test_overdispersion <- function(formula, data, R) {
  fit <- glm(formula, data = data, family = poisson)
  observed_dispersion <- sum(residuals(fit, type = "pearson")^2)
  / fit$df.residual

  boot_fn <- function(data, indices) {
    d <- data[indices, ]
    fit <- glm(formula, data = d, family = poisson)
    sum(residuals(fit, type = "pearson")^2) / fit$df.residual
  }

  boot_results <- boot(data, boot_fn, R = R)
  p_value <- mean(boot_results$t >= observed_dispersion)

  list(observed_dispersion = observed_dispersion, p_value = p_value,
boot_results = boot_results)
}

# Apply the test to the InsectSprays dataset
```

```
test_results <- test_overdispersion(count ~ spray, InsectSprays,
R = 1000)
```

```
# Print results
test_results
```

Bootstrap Test Results:

- Observed Dispersion: 1.507713
- P-value: 0.254
- Bootstrap Statistics:
 - Original: 1.507713
 - Bias: -0.150367
 - Std. Error: 0.2189001

Step 5: Comparison with the `dispersiontest` Function from the AER Package

The results are compared with the `dispersiontest` function from the AER package.

```
# Load necessary library
library(AER)
```

```
# Perform the dispersion test
dispersion_test_results <- dispersiontest(poisson_glm)
```

```
# Print results
dispersion_test_results
```

Results from the `dispersiontest` Function:

Overdispersion test

```
data: poisson_glm
z = 1.7371, p-value = 0.04119
alternative hypothesis: true dispersion is greater than 1
sample estimates:
dispersion
1.38207
```

Conclusion

In this analysis, we examined the distribution of insect counts for different sprays using both bootstrap methods and a Poisson GLM model. We calculated dispersion indices and tested the hypothesis of equidispersion. The results were also cross-validated with the `dispersiontest` function from the AER package, which confirmed the findings.

R Code

```
#STEP 1-----

data(InsectSprays)
head(InsectSprays)

library(boot)

# STEP 2-----
dispersion_index <- function(data, indices) {
  d <- data[indices]
  var(d) / mean(d)
}

# Function to each spray group
results <- lapply(split(InsectSprays$count, InsectSprays$spray), function(x) {
  boot_result <- boot(x, statistic = dispersion_index, R = 1000)
  boot_result
})

ci_results <- lapply(results, function(res) {
  list(
    percentile = boot.ci(res, type = "perc"),
    basic = boot.ci(res, type = "basic"),
    t = boot.ci(res, type = "bca")
  )
})

ci_results

# STEP 3-----
```

```

poisson_glm <- glm(count ~ spray, data = InsectSprays, family = poisson)

# Calculate the dispersion index
pearson_residuals <- residuals(poisson_glm, type = "pearson")
dispersion_index_glm <- sum(pearson_residuals^2) / poisson_glm$df.residual
dispersion_index_glm

# STEP 4-----
test_overdispersion <- function(formula, data, R) {
  fit <- glm(formula, data = data, family = poisson)
  observed_dispersion <- sum(residuals(fit, type = "pearson")^2) /
  fit$df.residual

  boot_fn <- function(data, indices) {
    d <- data[indices, ]
    fit <- glm(formula, data = d, family = poisson)
    sum(residuals(fit, type = "pearson")^2) / fit$df.residual
  }

  boot_results <- boot(data, boot_fn, R = R)
  p_value <- mean(boot_results$t >= observed_dispersion)

  list(observed_dispersion = observed_dispersion, p_value = p_value,
  boot_results = boot_results)
}

test_results <- test_overdispersion(count ~ spray, InsectSprays, R = 1000)

test_results

# STEP 5-----
library(AER)

dispersion_test_results <- dispersiontest(poisson_glm)
dispersion_test_results

```