Case Study 01

Frederico Augustos (Verificador), Mariana Pimenta (Relator), Pedro Maia(Coordenador)

09 de setembro de 2019

Resumo

Este artigo apresenta um estudo de caso que, por meio de inferência estatística, avalia o desempenho de uma nova versão de um software. Sendo assim, analisou-se a média e a variância do custo de execução de ambos algoritmos. Utilizando a base de dados da versão atual, que apresenta média populacional μ=50 e variância σ^2=100, aplicou-se testes para determinar se a nova versão apresenta melhor desempenho quando comparado com a anterior. Neste caso, melhor desempenho significa média e variância do custo de execução menores. Para a análise da média do custo de execução, definiu-se a hipóteses nula como a não diferença estatística entre o custo médio de execução dos algoritmos e a alternativa como o custo médio de execução do segundo algoritmo fosse menor. A mesma hipótese nula e alternativa foi adotada para a variância. Após a aplicação do "teste t" para a média do custo e do teste "bootstrap" para a variância do custo, concluímos que a hipótese nula pode ser rejeitada para variância e que a hipótese nula não pode ser rejeitada para a média dos custos.

Teste de média

Definição das hipóteses

Um dos critérios para comparar o desempenho de cada uma das versões é a média do custo de execução. Dessa forma, foi definido o seguinte teste de hipótese:

$$\begin{cases} H_0 : \mu = 50 \\ H_1 : \mu \leq 50 \end{cases} \tag{1}$$

A hipótese nula H_0 assume que a média da nova versão não difere da versão atual, ou seja, sugere que não há melhorias no custo médio. Já a hipótese alternativa H_1 assume que o custo médio da nova versão é menor que o custo atual, o que indica uma melhoria no desempenho. Visto que o objetivo era avaliar somente se o custo atual era menor ou igual do que o custo anterior, foi então definida uma hipótese alternativa unilateral.

Cálculo do tamanho da amostra

Para o teste de média, assumiu-se que o Teorema do Limite Central é aplicável e, portanto, considerou-se a normalidade dos dados. Os seus parâmetros foram definidos como, nível de significância α=0.01, poder de teste π=0.8 e tamanho de efeito δ=4. Assim, conforme código a seguir, calculou-se o tamanho da amostra do experimento por meio do teste "power.t.test".

```
# parâmetros
mu_0 <- 50
sigma_0 <- 10

alpha <- 0.01
beta <- 0.2
pi <- 1 - beta
delta <- 4
powerTest <- power.t.test(power = pi, delta = delta, sd = sigma_0, sig.level = alpha,
type = "one.sample", alternative = "one.sided")
print(powerTest)
```

```
##
##      One-sample t test power calculation
##
##               n = 65.45847
##           delta = 4
##              sd = 10
##       sig.level = 0.01
##           power = 0.8
##     alternative = one.sided
```

Coleta das observações

Por meio dos resultados obtidos no teste anterior definiu-se o tamanho da amostra para a coleta de dados $n = 66$. As rotinas fornecidas da nova versão do software foram executadas e os custos obtidos foram armazenados no arquivo "data.csv".

```
# Set up the data-generating procedure
library(ExpDE)
mre <- list(name = "recombination_bin", cr = 0.9)
mmu <- list(name = "mutation_rand", f = 2)
mpo <- 100
mse <- list(name = "selection_standard")
mst <- list(names = "stop_maxeval", maxevals = 10000)
mpr <- list(name = "sphere", xmin = -seq(1, 20), xmax = 20 + 5 * seq(5, 24))

N <- ceiling(powerTest$n) # tamanho da amostra
custos <- vector()
for (i in 1:N)
{
  custos[i] <- ExpDE(mpo, mmu, mre, mse, mst, mpr,
showpars = list(show.iters = "none"))$Fbest
}

write.csv(custos, "data.csv", row.names=FALSE)
```

Execução do teste

A partir das observações coletadas, o "t.test" foi executado e, como o p-valor > 0.01, o teste falhou em rejeitar a hipótese nula, ou seja, não existem evidências fortes o suficiente para se rejeitar a hipótese nula com um nível de confiança de 99%.

```
mean_test <- t.test(custos, alternative = "less", mu = mu_0, conf.level = 1-alpha)
print(mean_test)
```

```
##
##  One Sample t-test
##
## data:  custos
## t = -0.99292, df = 65, p-value = 0.1622
## alternative hypothesis: true mean is less than 50
## 99 percent confidence interval:
##      -Inf 51.21776
## sample estimates:
## mean of x
##  49.13148
```

O "t.test" fornece o intervalo de confiança para o nível de significância imposto. Como o nível superior do intervalo de confiança está acima do custo médio, há mais indícios de que o teste falhou em rejeitar a hipótese nula. O limite superior do intervalo de confiança U também pode ser calculado pela expressão abaixo, corroborando a conclusão de que não há evidências que o novo algoritmo apresenta melhor desempenho do que o anterior.

```
media <- mean(custos)
variancia <- var(custos)
desvio_padrao <- sqrt(variancia)

t <- (media - mu_0)/(desvio_padrao/sqrt(N))
p <- pt(t, df=N-1)

t_U <- qt(1-alpha, df=N-1)
U <- media + t_U*desvio_padrao/sqrt(N)
cat("t = ", t)
```

```
## t =  -0.9929175
```

```
cat("\np = ", p)
```

```
##
## p =  0.1622161
```

```
cat("\nU = ", U)
```

```
##
## U =  51.21776
```

Validação das premissas de teste

Uma forma quantitativa de avaliar a normalidade dos dados é através do teste de Shapiro-Wilk, que é um teste de hipótese. Neste teste, a hipótese nula H_0 afirma que a população é Normal e a hipótese alternativa H_1 afirmar o contrário. Deste modo, aplicou-se o teste nas amostras coletadas obtendo os seguintes resultados:

```
shapiro.test(custos)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  custos
## W = 0.92716, p-value = 0.0008206
```

Como o p-valor ficou muito baixo, existem fortes indícios que os dados não seguem uma distribuição normal. Mas pelo Teorema do Limite Central, a soma das médias de variáveis aleatórias independentes e igualmente distribuídas assumem normalidade.

Discussão da potência do teste

O tamanho de amostra calculado para a execução do teste considerou que a variância das duas versões era igual. Ao verificar a variância da nova versão, obteve-se um valor inferior ao de referência.

```
var(custos)
```

```
## [1] 50.49866
```

Como a variância do novo algoritmo é menor que a do algoritmo atual, a curva da distribuição é mais estreita que a assumida inicialmente, implicando numa menor interseção entre as curvas de custo de cada versão. Portanto, conclui-se que a potência do teste para este tamanho de amostra é maior.

```r
power.t.test(n = N, delta = delta, sd = sqrt(var(custos)), sig.level = alpha,
type = "one.sample", alternative = "one.sided")
```

```
##
##      One-sample t test power calculation
##
##               n = 66
##           delta = 4
##              sd = 7.106241
##       sig.level = 0.01
##           power = 0.9842277
##     alternative = one.sided
```

Conforme o resultado do "power.t.test", a potência do teste ao usar a variância amostral foi de fato bem maior que a potência original. Analogamente, caso a potência fosse mantida, o tamanho amostral considerando a variância amostral poderia ser bem menor.

```r
power.t.test(power = pi, delta = delta, sd = sqrt(var(custos)), sig.level = alpha,
type = "one.sample", alternative = "one.sided")
```

```
##
##      One-sample t test power calculation
##
##               n = 34.43228
##           delta = 4
##              sd = 7.106241
##       sig.level = 0.01
##           power = 0.8
##     alternative = one.sided
```

Teste da variância

Definição das hipóteses

O segundo critério para comparar o desempenho de cada uma das versões é a variância do custo de execução. Dessa forma, definiu-se o seguinte teste de hipótese:

$$\begin{cases} H_0 : \sigma^2 = 100 \\ H_1 : \sigma^2 \leq 100 \end{cases} \tag{2}$$

A hipótese nula H_0 assume que a variância do custo da nova versão não difere da versão atual, ou seja, sugere que não há melhorias na variância. Já a hipótese alternativa H_1 assume que a variância do custo da nova versão é menor que a variância do custo atual, o que indicaria uma melhoria no desempenho.

Execução do teste

Aplicou-se um teste não paramétrico sobre a variância dos custos visto que elas não seguem normalidade, e, portanto, nada se pode assumir sobre sua distribuição. O teste aplicado foi o bootstrap e o seu parâmetro definido foi o nível de significância α=0.05. Assim, foi possível executar o teste de hipóteses utilizando a mesma amostra do teste de média.

```r
library(boot)
boot.out <- boot(custos, statistic = function(x, i){var(x[i])}, R = 500)
boot.ci(boot.out, conf = 0.95 , type = "bca")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 500 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot.out, conf = 0.95, type = "bca")
##
## Intervals :
## Level       BCa
## 95%   (32.50, 85.86 )
## Calculations and Intervals on Original Scale
## Some BCa intervals may be unstable
```

Após a execução do teste, a hipótese nula foi rejeitada pois o limite superior do intervalo de confiança ficou abaixo do valor da variância do software atual.

Conclusões

Os testes aplicados sobre a média e variância dos custos nos permite rejeitar a hipótese nula somente para a variância. Isso implica que, apesar de não ser constatada uma diminuição do custo médio, o segundo algoritmo apresenta resultados mais " consistentes", ou seja, os valores de custo de cada amostra se desviarão menos da média amostral se comparada com os valores de custo do primeiro algoritmo. Portanto, o segundo algoritmo apresenta a melhor performance se comparado com o atual.

Questão bonus - Intervalo de tolerância

Esse cálculo assume uma distribuição normal dos dados, e indica os limites inferior e superior assumindo que 90% dos valores estão dentro desse intervalo de tolerância.

```
media <- mean(custos)
variancia <- var(custos)
desvio_padrao <- sqrt(variancia)
alpha <- 0.05
gamma <- 0.9

z <- qt(alpha/2, df=N-1)
chi <- qchisq(gamma, df=N-1)

delta <- desvio_padrao*sqrt((N-1)/N*(N+z*z)/chi)
L <- media - delta
U <- media + delta
cat("Lower bound:", L)
```

```
## Lower bound: 42.53417
```

```
cat("\nUpper bound: ", U)
```

```
##
## Upper bound:  55.72878
```

É importante ressaltar que esses limites só são válidos para dados seguindo uma distribuição normal. Os testes anteriores mostraram que os dados não estão normalmente distribuídos, portanto este cálculo do intervalo de tolerância não pode ser aplicado.