PPGEE UFMG - EEE933 - Estudo de Caso 03 (SOLUÇÃO COMPLEMENTAR)

Equipe 3
(Verificadora) Amanda Fernandes Vilaça Martins, (Relator) Bruno Marciano Lopes, (Monitor) Igor Almeida Baratta, (Coordenador) Tiago de Sá Ferreira

10 de outubro de 2016

Planejamento do experimento

Características desejadas para os testes estatísticos

```
alpha <- 0.05
PI <- 0.8
beta <- 1 - PI
dt <- 1
delta_a <- 0.03
n_rep <- 30
n_t <- 8
```

Definição do número de amostras necessárias para o teste da acurácia

Para o teste referente à acurácia, para definir o número de instâncias n_a considerando a potência desejada de $\Pi = 0.8$, é necessário conhecer σ_{aS} e σ_{aP}. As variâncias (e os desvios padrões) das acurácias dos algoritmos são parâmetros desconhecidos. No entanto, uma vez que já se possui um número de instâncias calculado para o teste do tempo, os dados gerados com n_t instâncias podem ser utilizados para estimar as variâncias das acurácias.

Dessa forma, uma primeira execução do aplicativo (Campelo 2016) gerou os dados consolidados no arquivo "*1991-09-15_8_30.csv*".

```
dados_preteste <- read.table("1991-09-15_8_30.csv",sep = ",", header = TRUE)

# Avalia as oito instâncias definidas para o teste do tempo
dados_inicial <- head(dados_preteste,2*n_t*n_rep)
dados_ac_pre <- aggregate(Accuracy~Algorithm:Instance, data=dados_inicial, FUN=mean)
summary(dados_ac_pre)
```

```
##     Algorithm    Instance     Accuracy
##  Proposed:8   Inst01 :2   Min.   :0.8185
##  Standard:8   Inst02 :2   1st Qu.:0.8543
##               Inst03 :2   Median :0.8766
##               Inst04 :2   Mean   :0.8767
##               Inst05 :2   3rd Qu.:0.8909
##               Inst06 :2   Max.   :0.9339
##               (Other):4
```

A partir dessa base de dados, estima-se a variância amostral das acurácias dos algoritmos. Utilizando uma abordagem mais conservadora, ao invés de utilizar os valores calculados como variâncias das acurácias, serão considerados nos testes os maiores valores de variância assumindo um intervalo de confiança utilizando o mesmo nível de significância já mencionado. Assim, assumindo que as distribuições das variâncias são normais, tem-se por (Nordheim, Clayton, and Yandell 2003) que:

```
s2_interval = function(data, significance.level){
  df <- length(data) - 1
  chilower <- qchisq(significance.level/2, df)
  chiupper <- qchisq(significance.level/2, df, lower.tail = FALSE)
  v = var(data)
  c(df*v/chiupper,df*v/chilower)
}
```

A maior variância considerada da acurácia $S^2_{aPcon,max}$ do algoritmo simplificado proposto é:

```
s2_aPcon_max <- max(s2_interval(dados_ac_pre$Accuracy[dados_ac_pre$Algorithm=="Proposed"],
                                alpha))
cat("s2_aPcon_max  =",s2_aPcon_max)
```

```
## s2_aPcon_max  = 0.001463391
```

A maior variância considerada da acurácia $S^2_{aScon,max}$ do algoritmo padrão original é:

```
s2_aScon_max = max(s2_interval(dados_ac_pre$Accuracy[dados_ac_pre$Algorithm=="Standard"],
                               alpha))
cat("s2_aScon_max  =",s2_aScon_max)
```

```
## s2_aScon_max  = 0.002153307
```

Para o teste de hipotése da acurácia será utilizada o método TOST, onde o teste será quebrado em dois testes-t unilaterais, e portanto será utilizada a função calcN_tost2 (Campelo 2015). Para se obter pelo menos a potência $\Pi = 0.8$ no teste referente à acurácia, considerando amostras de tamanhos iguais para ambos os algoritmos, o número de instâncias n_a do teste pode ser determinado através de:

```
# Calcula o desvio padrão amostral das acurácias dos algoritmos
sd_as <- sqrt(s2_aScon_max)
sd_aP <- sqrt(s2_aPcon_max)

# Calcula o tamanho de amostras necessário
n_a<-ceiling(calcN_tost2(alpha = alpha, beta = 1-PI, diff_mu = (delta_a/2),
                         tolmargin = delta_a, s1 = sd_aP, s2 = sd_as))
cat("n_a =",n_a)
```

```
## n_a = 102
```

Assim, é necessário realizar noventa e quatro ($n_{new} = n_a - n_t$) novas amostragens para que o teste da acurácia seja realizado com o nível de potência desejada. O aplicativo (Campelo 2016) é executado mais uma vez com um número de instâncias n_{new}. Admitindo que a nova execução do aplicativo é independente da execução inicial e que as características dos algoritmos não foi alterada, os resultados das novas instâncias podem ser simplesmente concatenados ao arquivo "*1991-09-15_8_30.csv*" (os novos dados compõem o conjunto de amostras total). Essa concatenação é realizada direto no arquivo *.csv* após a renomeação manual das instâncias para representarem corretamente a sequência já iniciada. Com isso, os dados iniciais são renomeados (*Inst01 - Inst08*) e os novos dados (*Inst1 - Inst94*) são concatenados ao final do arquivo original. A opção pelo ajuste manual ao invés de programar uma rotina computacional para realizar a concatenação de forma automatizada é justificada por se buscar manter um único arquivo *.csv* como base de dados.

```
# Avalia as cento e duas instâncias definidas para o teste do tempo
dados_final <- head(dados_preteste,2*n_a*n_rep)
```

Com isso, o arquivo "*1991-09-15_8_30.csv*" passa a conter dados das cento e duas instâncias necessárias para o teste.

Análise Exploratória dos Dados

Os dados são consolidados em um arquiv.o *.csv*. Tem-se para as n_a instâncias definidas do teste de acurácia:

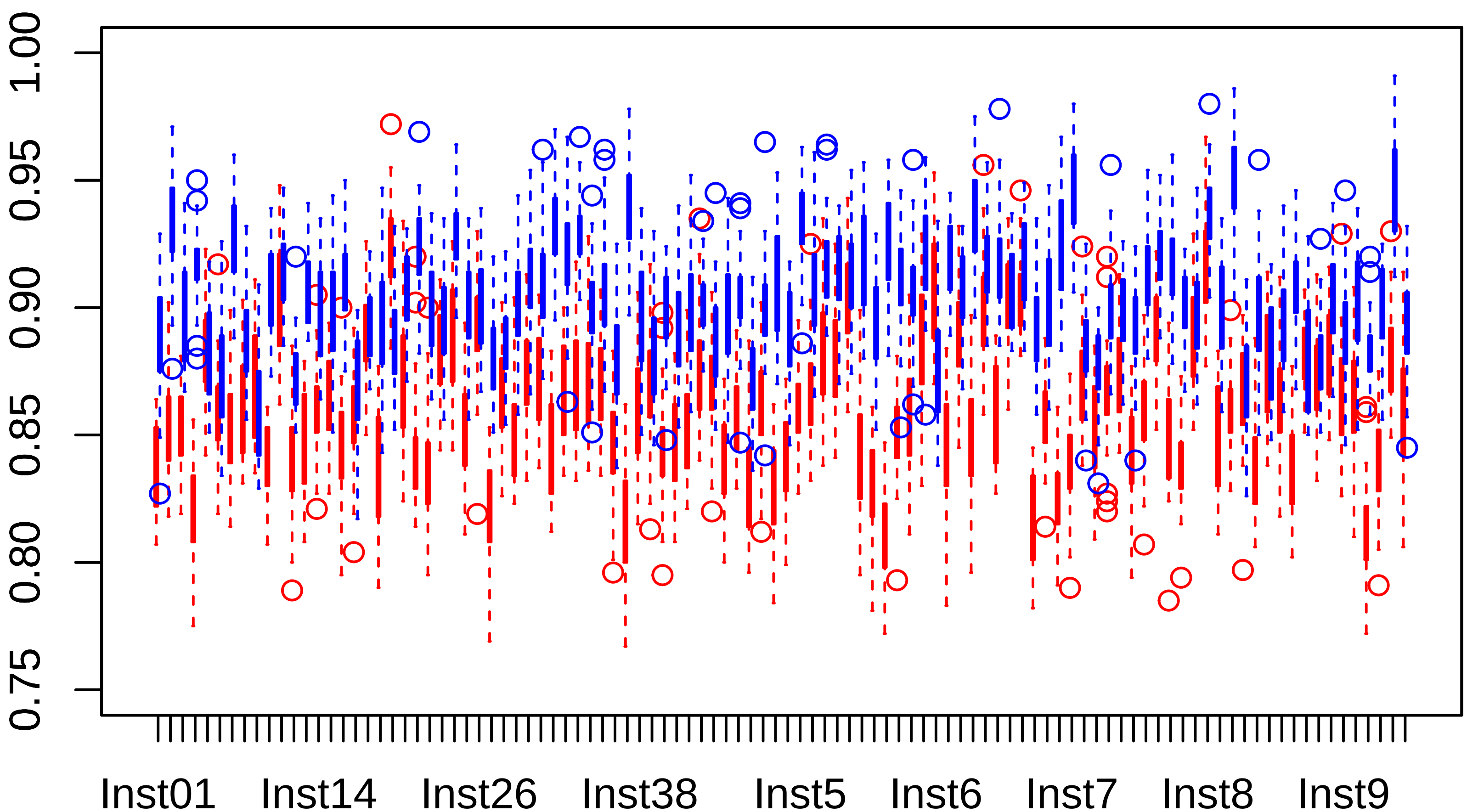
```
dados_ac <- aggregate(Accuracy~Algorithm:Instance, data=dados_final, FUN=mean)
dados_ac <- droplevels(dados_ac)

dados_acP_plot <- dados_final[which(dados_final$Algorithm=="Proposed"),]
dados_acP_plot <- droplevels(dados_acP_plot)

dados_acS_plot <- dados_final[which(dados_final$Algorithm=="Standard"),]
dados_acS_plot <- droplevels(dados_acS_plot)

summary(dados_ac)
```

```
##     Algorithm      Instance      Accuracy
##  Proposed:102   Inst01 :  2   Min.   :0.8095
##  Standard:102   Inst02 :  2   1st Qu.:0.8598
##                 Inst03 :  2   Median :0.8850
##                 Inst04 :  2   Mean   :0.8819
##                 Inst05 :  2   3rd Qu.:0.9023
##                 Inst06 :  2   Max.   :0.9482
##                 (Other):192
```

O *boxplot* das acurácias para as n_{rep} execuções de cada algoritmo em cada instância é:

```
#caixas invisíveis / deslocar caixas esq. em -0.15 / deslocar caixas dir. em +0.15
boxplot(dados_ac$Accuracy~Instance, data = dados_ac, xlim = c(0.5, n_a+0.5),
        ylim = c(0.75, 1.0), boxfill=rgb(1, 1, 1, alpha=1), border=rgb(1, 1, 1, alpha=1))
boxplot(dados_acP_plot$Accuracy~Instance, data = dados_acP_plot, xaxt = "n",
        add = TRUE, boxfill="red", boxwex=0.25, at = 1:n_a - 0.15, border = "red")
boxplot(dados_acS_plot$Accuracy~Instance, data = dados_acS_plot,
        main = "Acurácia: padrão original (azul) X propos. simplificado (vermelho)",
        xaxt = "n", add = TRUE, boxfill="blue", boxwex=0.25, at = 1:n_a + 0.15,
        border = "blue")
```

Acurácia: padrão original (azul) X propos. simplificado (vermelho)



Análise estatística

Teste de hipóteses - acurácia

Como definido anteriormente para o caso da acurácia, optou-se por um teste de equivalência da acurácia dos algoritmos). Será utilizado o método TOST (*two one-sided tests*) para testar a hipótese (definida anteriormente) de não inferioridade.

Assim, para a inspeção da inferioridade do algoritmo simplificado proposto, tem-se que:

$$\begin{cases} H_{a0}^1 : \mu_{aP} - \mu_{aS} = -\delta_a^* \\ H_{a1}^1 : \mu_{aP} - \mu_{aS} < -\delta_a^* \end{cases}$$

```r
with(dados_ac,
     t.test(Accuracy~Algorithm, mu = -delta_a, paired=TRUE, alternative="less",
            conf.level = 1-alpha))
```

```
##
##  Paired t-test
##
## data:  Accuracy by Algorithm
## t = -4.4218, df = 101, p-value = 1.236e-05
## alternative hypothesis: true difference in means is less than -0.03
## 95 percent confidence interval:
##         -Inf -0.03799594
## sample estimates:
## mean of the differences
##             -0.04280229
```

Como ($p_a^1 << \alpha$), é possível rejeitar H_{a0}^1 em detrimento da hipótese alternativa.

Já para a inspeção da superioridade do algoritmo simplificado proposto, tem-se que:

$$\begin{cases} H_{a0}^2 : \mu_{aP} - \mu_{aS} = \delta_a^* \\ H_{a1}^2 : \mu_{aP} - \mu_{aS} > \delta_a^* \end{cases}$$

```r
with(dados_ac,
     t.test(Accuracy~Algorithm, mu = delta_a, paired=TRUE, alternative="greater",
            conf.level = 1-alpha))
```

```
##
##  Paired t-test
##
## data:  Accuracy by Algorithm
## t = -25.145, df = 101, p-value = 1
## alternative hypothesis: true difference in means is greater than 0.03
## 95 percent confidence interval:
##  -0.04760863         Inf
## sample estimates:
## mean of the differences
##             -0.04280229
```

Como ($p_a^2 > \alpha$), não é possível rejeitar a hipótese nula H_{a0}^2.

Validação da premissa de normalidade das médias

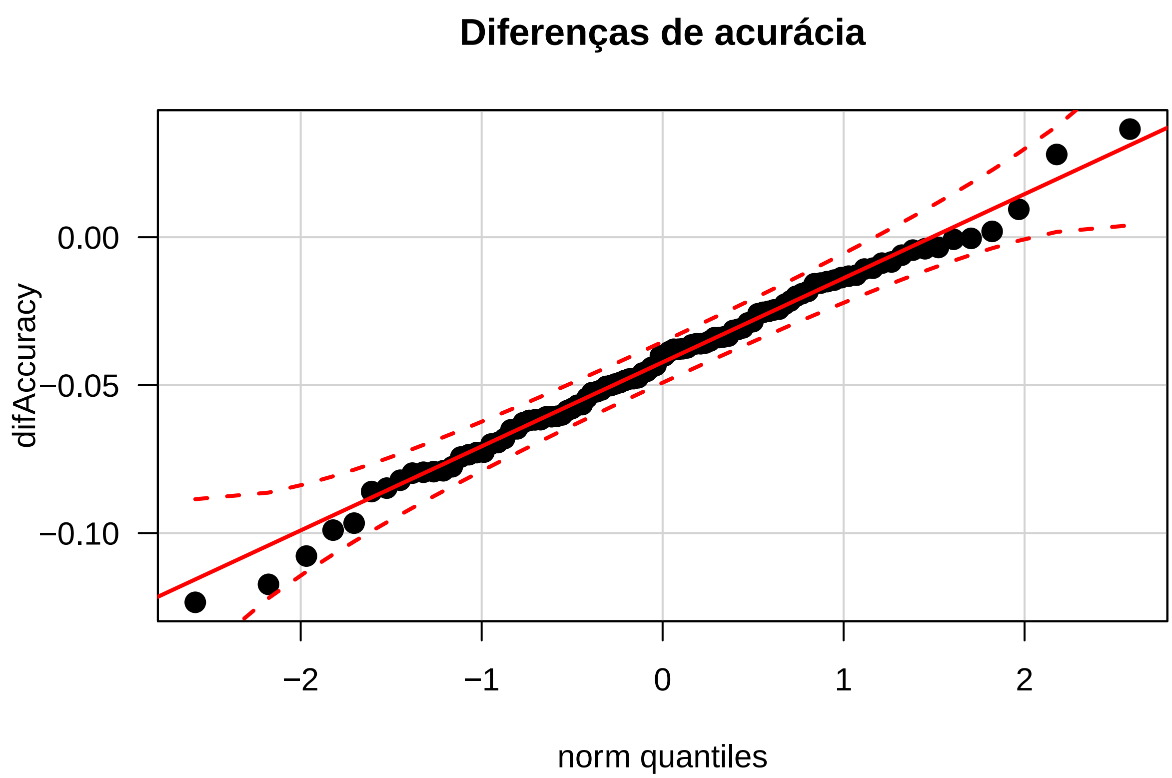
Deseja-se verificar a premissa de normalidade das médias (diferenças) da acurácia através do teste de normalidade de Shapiro-Wilk (Campelo 2015). Para as diferenças de acurácia, tem-se:

```
difAccuracy <- dados_ac$Accuracy[dados_ac$Algorithm=="Proposed"] -
                        dados_ac$Accuracy[dados_ac$Algorithm=="Standard"]
shapiro.test(difAccuracy)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  difAccuracy
## W = 0.99268, p-value = 0.8604
```

Para que a validação da premissa seja mais compreensiva, também serão apresentados os *qqplot* dessas diferenças.

```
qqPlot(difAccuracy, pch=16, cex=1.5, las=1, main = "Diferenças de acurácia")
```



Considerando os *qqplot* apresentados e que o *valor-p* encontrado foi superior ao α_{norm} determinado para o teste, acredita-se que não há nenhum forte indício para rejeição da premissa de normalidade das médias da acurácia.

Conclusões

É possível concluir com um nível de confiança de 95% que o algoritmo proposto não é equivalente ao algoritmo original. De fato, os testes realizados levam à conclusão de que há uma degradação considerável de acurácia (resultado do teste de não-inferioridade). O intervalo de confiança para a diferença das médias da acurácia se encontra na região de rejeição afastado da região crítica no caso do teste de não-inferioridade. O tamanho de efeito prático foi $\delta_a^* = 0.03$ e o número de instâncias considerado no teste foi de $n_t = 102$. Uma análise conservadora foi utilizada na variância amostral considerada, sendo que se utilizou como parâmetro dos testes a maior variância amostral do intervalo de confiança para $\alpha = 0.05$.

Referências bibliográficas

Campelo, Felipe. 2015. "Lecture Notes on Design and Analysis of Experiments (Version 2.11; Creative Commons BY-NC-SA 4.0)." Website, Acesso em 08/set/2016. http://git.io/v3Kh8.

———. 2016. "Classification Algorithms Experiment - Simulator." Website, Acesso em 04/out/2016. http://orcslab.cpdee.ufmg.br:3838/classdata/.

Nordheim, EV, MK Clayton, and BS Yandell. 2003. "7.6.2 Appendix - Using R to Find Confidence Intervals." Website, Acesso em 04/out/2016. https://www.stat.wisc.edu/~yandell/st571/R/append7.pdf.