

# Estudo de Caso 02: Comparação entre Algoritmos de Classificação

*Equipe 04*

*15 de maio de 2017*

Coordenador: Danny Tonidandel

Relator: Alessandro Cardoso Verificador: Gustavo Vieira Monitor: Bernardo Marques

## O Experimento

Este estudo de caso consiste na comparação de algoritmos de classificação em que um deles consiste no *padrão atual* e o outro em um novo algoritmo proposto, que utiliza uma técnica de simplificação baseada em inferência estatística.

Segundo os pesquisadores responsáveis pelo algoritmo proposto, este apresenta uma melhora significativa frente ao padrão atual com relação ao tempo requerido para a classificação e que essa nova abordagem não resulta em grandes perdas de desempenho em termos de acurácia da classificação.

O experimento deste trabalho foi desenvolvido no intuito de verificar as afirmações acima. Este consiste na comparação dos dois algoritmos citados baseando-se nos questionamentos abaixo:

## Questionamentos

1. O método proposto realmente apresenta ganhos em relação ao tempo de execução, quando comparado ao método padrão?
2. O método proposto realmente não resulta em variações consideráveis de acurácia?

## Características desejadas

Para que sejam investigados os questionamentos acima são desejadas as seguintes características para os testes estatísticos:

- Nível de significância:  $\alpha = 0.05$ ;
- Tamanho de efeito de interesse prático para os ganhos de tempo:  $d_{tempo}^* = 1.5$ ;
- Margem de não-inferioridade para acurácia:  $\delta_{acuracia}^* = 0.05$ .
- Potência desejada:  $\pi = 0.8$ ;

## Planejamento experimental

### Caracterização do experimento

Os dados experimentais para os testes foram gerados por meio de uma aplicação disponibilizada na web em [http : //orcslab.cpdee.ufmg.br/3838/classdata](http://orcslab.cpdee.ufmg.br/3838/classdata). Os dados são referentes ao tempo necessário para classificação (*Time.s*) e acurácia (*Accuracy*) por cada um dos algoritmos em cada instância e em cada

execução. Para geração dos dados o grupo de trabalho deve fornecer a data de nascimento do membro mais jovem da equipe e selecionar um número de instâncias e execuções por instâncias.

Estes dados são apresentados conforme seleção de número de instâncias e número de repetições das mesmas. Segundo [1] quando cada par de observação é coletado sobre condições homogêneas, ainda assim estas podem variar de um par para outro. Conforme apresentado em [2], a variabilidade devida aos diferentes problemas de teste é uma forte fonte de variação espúria que pode e deve ser controlada e que uma solução elegante para eliminar a influência deste inconveniente é o pareamento das medições por instância. O número de execuções por instância escolhido pelo grupo foi 30, por se tratar de um número comumente utilizado.

Conforme a apresentação das amostras, verificou-se a necessidade de efetuar os testes estatísticos com as amostras pareadas. Para [1] o procedimento experimental mais adequado quando os dados são coletados aos pares é o *t-test pareado*. Para tal, seja  $(X_{11}, X_{21}), (X_{12}, X_{22}), \dots, (X_{1n}, X_{2n})$  um conjunto de  $n$  observações pareadas onde assumimos que a média e a variância da população representada por  $X_1$  são  $\mu_1$  e  $\sigma_1^2$ , e a média e a variância da população representada por  $X_2$  são  $\mu_2$  e  $\sigma_2^2$ . Defini-se as diferenças entre cada par de observações como  $D_j = X_{1j} - X_{2j}$ ,  $j = 1, 2, p, \dots, n$ . Os  $D_j$ 's são assumidos como sendo normalmente distribuídos na média.

## Definição de hipóteses

Os testes de hipótese elaborados são:

- Referente ao questionamento 1 definiu-se o teste de hipótese unilateral convencional, onde a hipótese nula é de que o algoritmo proposto apresenta tempo de execução similar ao algoritmo atual. Já a hipótese alternativa é de que o algoritmo proposto é mais rápido que o padrão atual. Esta formulação é apresentada abaixo.

$$\begin{cases} H_0 : \mu_{pt} - \mu_{at} = 0 \\ H_1 : \mu_{pt} - \mu_{at} < 0 \end{cases}$$

Os valores de  $\mu_{pt}$  e  $\mu_{at}$  são as médias de tempo de execução dos algoritmos *proposto* e *padrão atual* respectivamente.

- Referente ao questionamento 2 definiu-se o teste de não-inferioridade onde a hipótese nula é que o algoritmo proposto apresenta acurácia média inferior ao algoritmo atual. A hipótese alternativa é que o algoritmo proposto não apresenta acurácia média inferior ao padrão atual como abaixo:

$$\begin{cases} H_0 : \mu_{pa} - \mu_{aa} = -\delta_{acurácia}^* \\ H_1 : \mu_{pa} - \mu_{aa} > -\delta_{acurácia}^* \end{cases}$$

Os valores de  $\mu_{pa}$  e  $\mu_{aa}$  são os valores arbitrários de acurácia dos algoritmos *proposto* e *padrão atual* respectivamente e  $\delta_{acurácia}^*$  é a margem de não-inferioridade para a acurácia.

## Tamanho amostral

Afim de se atingir as características exigidas para o experimento é necessário definir o tamanho amostral a ser utilizado nos testes.

No caso de amostras pareadas, entende-se por amostras, o números de instâncias conforme exposto por [3], ou problemas a serem resolvidos pelos algoritmos analisados neste trabalho. Outra consideração é o número de repetições a serem adotadas por instância. Para este estudo adotou-se o valor de 30 repetições por instância seguindo recomendação expressas em [2] de que um valor heurístico para esta finalidade deve ser maior ou igual a 30.

Para a determinação do tamanho amostral foi realizado um *Estudo Piloto*, de forma a determinar uma estimativa inicial para os testes atendendo aos parâmetros desejados.

Como não há informação histórica da variância dos dados, a equipe decidiu adotar o método iterativo dado pela equação abaixo para determinar o número de amostras para o estudo piloto. Como  $d_{tempo}^*$  é conhecido, este foi utilizado para o cálculo.

$$n = 2\left(\frac{1}{d_{tempo}^*}\right)^2(t_{\alpha/2} + t_{\beta})^2$$

,

Os valores de  $t_{\alpha/2}$  e  $t_{\beta}$  foram substituídos para  $z_{\alpha/2}$  e  $z_{\beta}$  na equação acima considerando-se a primeira iteração, de forma a fazê-los independentes de  $n$ . As iterações seguintes convergiram rapidamente para um valor de 17 amostras a serem utilizadas no *Estudo Piloto* e foram geradas conforme abaixo:

- Nascimento: 21/11/1992;
- Número de instâncias de cada problema: 17;
- Número de execuções: 30;

```
#dados <- read.csv("1992-11-21_17_30.csv", header = T)
#head(dados)
```

Observou-se que os dados gerados continham inconsistências por apresentarem valores negativos para o tempo de execução do algoritmos como se pode observar abaixo.

Inserir código para verificar dado negativo

```
err = which (v_data < 0)
if (length(err)) v_data <- v_data[-err]
#Não sei bem se isto está certo, mas baseei no que o professor fez para checar inconsistências no estud
```

```
neg = dados$Time.s < 0
neg2 = dados$Accuracy < 0 & dados$Accuracy > 1 \ dados[neg,]
```

Os dados negativos de tempos de execução e respectivas acurácias foram retirados para se evitar contaminação do resultados. Para tal, considerou-se que a retirada de 2 valores não seria representativa na massa amostral adotada.

Inserir código substituindo valores negativos por NA

```
dados[neg,]$Time.s = NA
dados[neg,]$Accuracy = NA \ dados[neg,]
```

Por meio dos dados do *Estudo Piloto* obteve-se o desvio padrão das amostras dos algoritmos. Estes valores foram utilizados para se obter o tamanho amostral adequado às características exigidas para os testes. Utilizando-se a função “calcN\_tost2” fornecida por [2], foram verificados os tamanhos amostrais mínimos necessários considerando-se o tempo de execução e acurácia dos algoritmos.

```
v_nTime = calcN_tost2(alpha = 0.05,
  beta = 0.2,
  diff_mu = 1,
  tolmargin = min(v_dataSD$Time.s),
  s1 = v_dataSD$Time.s[1],
  s2 = v_dataSD$Time.s[2]
)

v_nAcc = calcN_tost2(alpha = 0.05,
  beta = 0.2,
  diff_mu = 0.01,
```

```

        tolmargin = 0.05,
        s1 = v_dataSD$Accuracy[1],
        s2 = v_dataSD$Accuracy[2]
    )

v_n = ceiling(max(v_nTime, v_nAcc))

```

Para a geração e coleta dos dados foi adotado o maior tamanho amostral verificado para o tempo de execução e acurácia. Neste caso, quem apresentou o maior valor foi o tempo de execução. Este valor atende às características exigidas para ambos os testes sem necessidade de uma nova coleta de dados para o caso da acurácia. Os dados foram gerados conforme abaixo:

- Nascimento: 21/11/1992;
- Número de instâncias de cada problema: 33;
- Número de execuções: 30;

As amostras geradas para os testes também apresentaram tempo de execução negativo e estas foram tratadas conforme as amostras do *Estudo de Piloto*.

Inserir código para verificar dado negativo

```

neg = dados$Time.s < 0
neg2 = dados$Accuracy < 0 & dados$Accuracy > 1 \ dados[neg,]

```

Inserir código substituindo valores negativos por NA

```

dados[neg,]$Time.s = NA
dados[neg,]$Accuracy = NA \ dados[neg,]

```

## Teste das Hipóteses

### Resultados

O teste de hipótese referente ao questionamento 1 foi implementado como abaixo e seu resultado indica a rejeição a hipótese nula.

```
# hypothesis test Time
```

```

v_diffTime = subset(v_data, Algorithm=='Proposed')$Time.s - subset(v_data, Algorithm=='Standard')$Time.s
v_tTestTime = t.test(v_diffTime,
                     conf.level = 0.05,
                     mu=0,
                     alternative = 'less'
                     )
v_pTime = v_tTestTime$p.value

```

O gráfico bloxplot abaixo evidência uma diferença entre as médias dos tempos de execução. Podemos ver que o algoritmo proposto possui média de tempo de execução menor que a média do algoritmo padrão.

Inserir chunk

```

boxplot(Time.s ~ Algorithm, data = mean.Time, col = c("green", "blue"), main = "Tempo - Média das
Instâncias", names = c("Proposed", "Standard"), xlab = "Algoritmo", ylab = "Tempo de Execução")

```

O teste de hipótese referente ao questionamento 2 foi implementado como abaixo e seu resultado indica que não se pode rejeitar a hipótese nula com o nível de confiança desejado.

```
# hypothesis test acc
v_diffAcc = subset(v_data, Algorithm=='Proposed')$Accuracy - subset(v_data, Algorithm=='Standard')$Accuracy
v_tTestAcc = t.test(v_diffAcc,
                    mu = -0.05,
                    conf.level = 0.05,
                    alternative = 'greater'
                    )
v_pAcc = v_tTestAcc$p.value
```

Refazer o texto abaixo

O gráfico bloxplot abaixo evidência uma diferença entre as médias das acurácias. Podemos ver que a média das acurácias para algoritmo proposto é menor que a do algoritmo padrão.

Inserir chunk

```
boxplot(Accuracy ~ Algorithm, data = mean.Time, col = c("green", "blue"), main = "Tempo - Média das
Instâncias", names = c("Proposed", "Standard"), xlab = "Algoritmo", ylab = "Tempo de Execução")
```

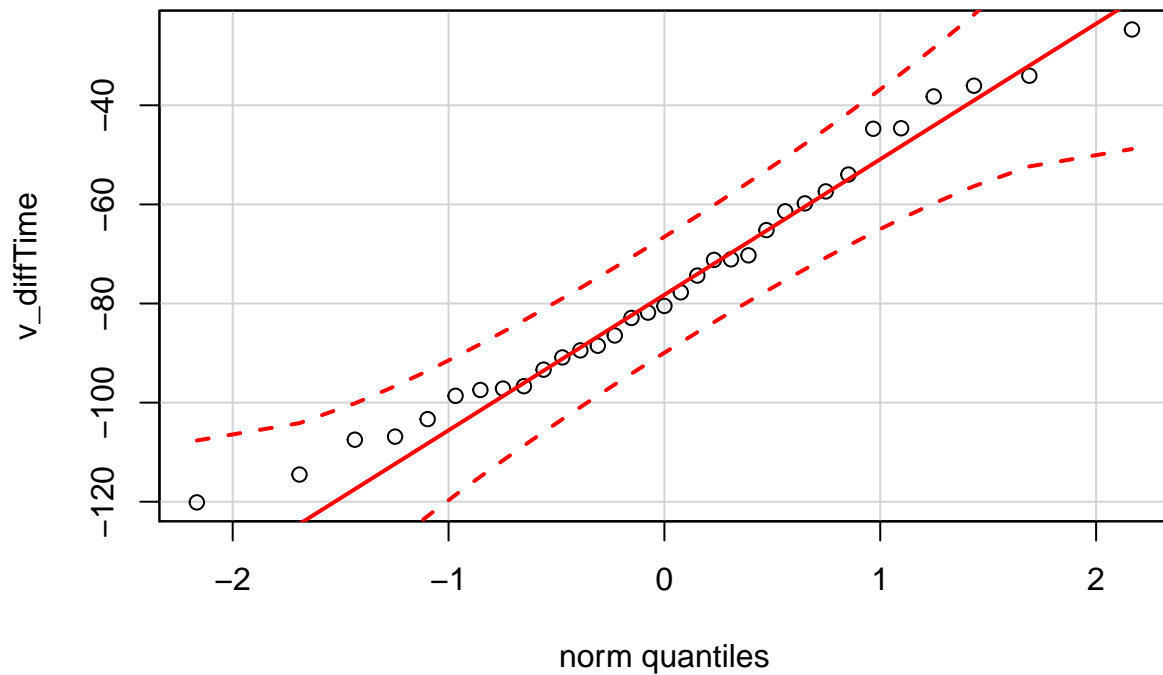
## Validação das Premissas

Como não se tem informações sobre a variância da população, o grupo adotou o teste de t student, assumindo a premissa de normalidade e independência.

A premissa de normalidade para o teste do questionamento 1 foi testada inicialmente de forma visual a partir do gráfico QQplot

```
## Hypothesis validation - Time

# Normality
qqPlot(v_diffTime)
```



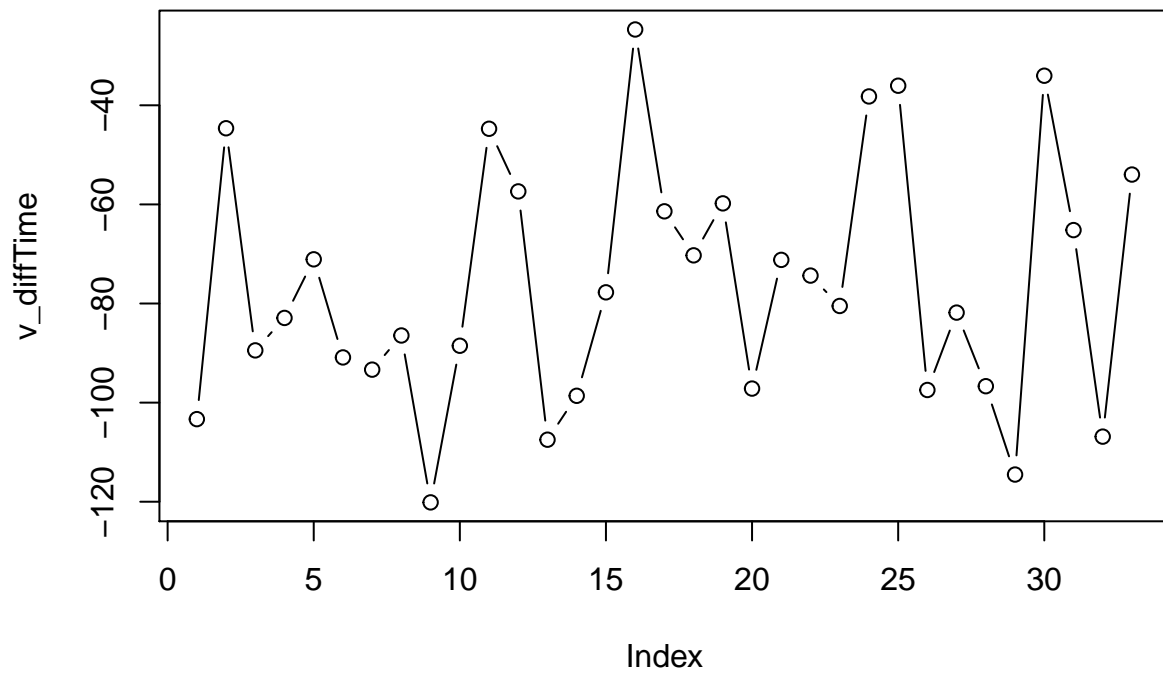
O gráfico QQplot para o tempo de execução sugere normalidade. Sendo assim foi aplicado o teste de Shapiro-Wilk para normalidade

```
v_shapiroTime = shapiro.test(v_diffTime)
```

O resultado sugere normalidade considerando a diferença de tempo médio de execução dos algoritmos analisados neste estudo de caso.

Outro teste utilizado foi o de verificação de independência de Durbin-Watson:

```
# Independence
v_dwTime = dwtest(v_diffTime~1)
plot(v_diffTime, type='b')
```

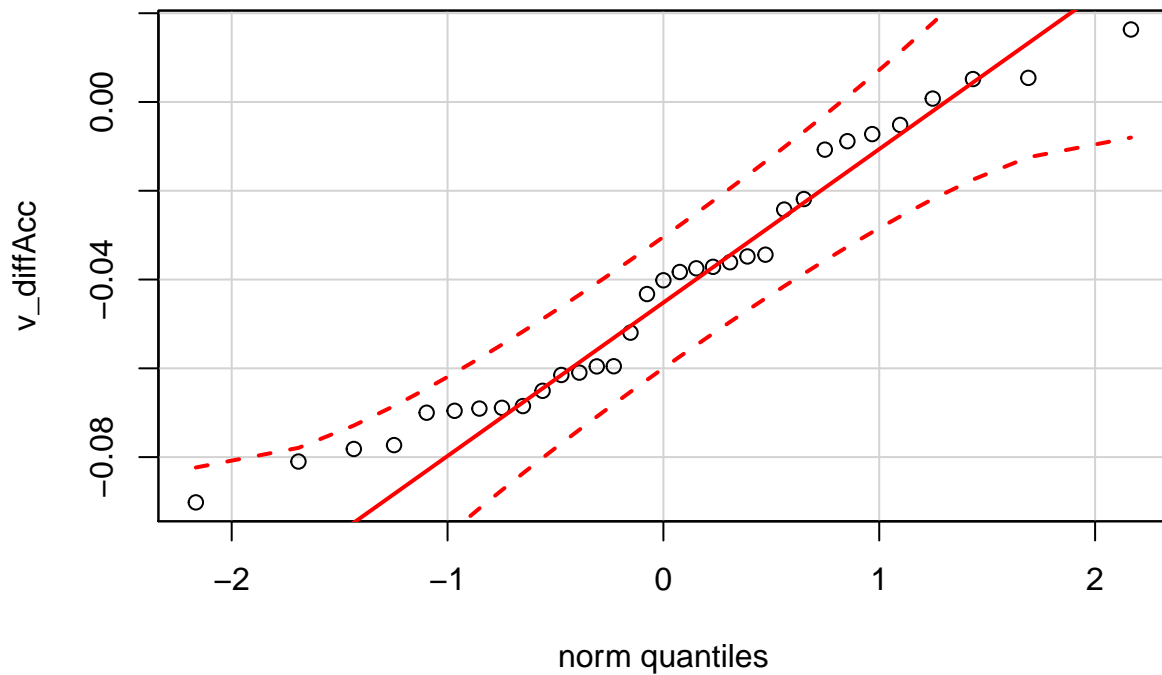


A premissa de normalidade para o teste do questionamento 2 foi testada inicialmente de forma visual a partir do gráfico QQplot.

```
## Hypothesis validation - Acc
```

```
# Normality
```

```
qqPlot(v_diffAcc)
```



O gráfico QQplot para a acurácia sugere normalidade. Sendo assim foi aplicado o teste de Shapiro-Wilk para normalidade.

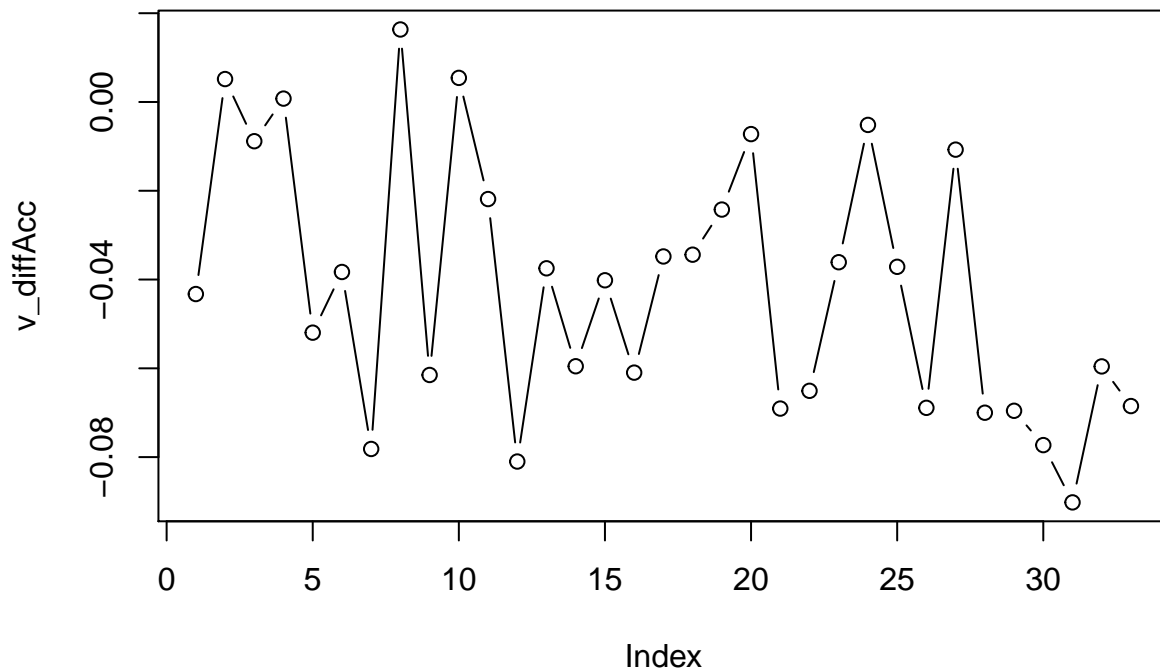
```
v_shapiroAcc = shapiro.test(v_diffAcc)
```

O resultado sugere normalidade considerando a diferença de acurácia média dos algoritmos analisados neste estudo de caso.

Também se verificou a independência por meio do teste Durbin-Watson para a acurácia dos algoritmos deste estudo:

```
# Independence
v_dwAcc = dwtest(v_diffAcc~1)
plot(v_diffAcc, type='b')
```





## Conclusão

O grupo conseguiu realizar as análises da relação entre tempo de execução e acurácia entre os algoritmos padrão e alternativo com sucesso. Os resultados foram: 1. O método proposto apresenta sim ganhos significativos em termos de tempo de execução em relação ao método padrão. 2. O método proposto resulta em variações consideráveis de acurácia.

Em relação ao primeiro resultado, pela rejeição da hipótese nula conclui-se que o método proposto tem ganhos significativos em termos de tempo de execução. Em relação ao segundo resultado, como não foi possível rejeitar a hipótese nula com o nível de significância desejado, não pode-se concluir que o método novo é não-inferior em relação ao método padrão.

Vale ainda notar que o número calculado de amostras necessárias para as características desejadas do teste foram consideravelmente distoantes, 33 para o primeiro teste e 4 para o segundo. Foi então decidido fazer apenas um teste com o número mais alto de amostras, uma vez que isto não implicaria em perdas para nenhum dos casos e não seria necessário fazer duas coletas. Um resultado disto foi que para o segundo teste a potência calculado foi de 1. Isto significa que a chance de ocorrer o erro do tipo *II*, que é a falha em rejeitar a hipótese nula sendo esta falsa, é 0. Por tanto, consideramos que as variações de acurácia são de fato consideráveis.

## Referências

- [1] D. C. Montgomery and G. C. Runger, *Applied statistics and probability for engineers*, vol. 5. John Wiley;

Sons, 2011.

[2] F. Campelo, “Lecture notes on design and analysis of experiments.” <https://github.com/fcampelo/Design-and-Analysis-of-Experiments>, 2015.

[3] E. Walker and A. S. Nowacki, “Understanding equivalence and noninferiority testing.” *Journal of general internal medicine*, vol. 26 2, pp. 192–6, 2011.