UNIVERSIDADE FEDERAL DO PARANÁ
Especialização em Inteligência Artificial Aplicada
IAA004 - Linguagem R
Prof. Dr. Razer A N R Montaño

TRABALHO DA DISCIPLINA

Estudante: Bruno Moreira Ribas

Este trabalho pode ser realizado em equipes de no máximo 5 integrantes

O que deve ser entregue:

- Um arquivo compactado com os documentos e arquivos
- A lista de comandos R que foi executada, com suas respectivas saídas
- Um texto com o resultado e justificativa do porque
- Outros arquivos pedidos (ex, modelo gerado)

1 Pesquisa com Dados de Satélite (Satellite)

O banco de dados consiste nos valores multi-espectrais de pixels em vizinhanças 3x3 em uma imagem de satélite, e na classificação associada ao pixel central em cada vizinhança. O objetivo é prever esta classificação, dados os valores multi-espectrais.

Um quadro de imagens do Satélite Landsat com MSS (Multispectral Scanner System) consiste em quatro imagens digitais da mesma cena em diferentes bandas espectrais. Duas delas estão na região visível (correspondendo aproximadamente às regiões verde e vermelha do espectro visível) e duas no infravermelho (próximo). Cada pixel é uma palavra binária de 8 bits, com 0 correspondendo a preto e 255 a branco. A resolução espacial de um pixel é de cerca de 80m x 80m. Cada imagem contém 2340 x 3380 desses pixels. O banco de dados é uma subárea (minúscula) de uma cena, consistindo de 82 x 100 pixels. Cada linha de dados corresponde a uma vizinhança quadrada de pixels 3x3 completamente contida dentro da subárea 82x100. Cada linha contém os valores de pixel nas quatro bandas espectrais (convertidas em ASCII) de cada um dos 9 pixels na vizinhança de 3x3 e um número indicando o rótulo de classificação do pixel central.

As classes são: solo vermelho, colheita de algodão, solo cinza, solo cinza úmido, restolho de vegetação, solo cinza muito úmido.

Os dados estão em ordem aleatória e certas linhas de dados foram removidas, portanto você não pode reconstruir a imagem original desse conjunto de dados. Em cada linha de dados, os quatro valores espectrais para o pixel superior esquerdo são dados primeiro, seguidos pelos quatro valores espectrais para o pixel superior central e, em seguida, para o pixel superior direito, e assim por diante, com os pixels lidos em sequência, da esquerda para a direita e de cima para baixo. Assim, os quatro valores espectrais para o pixel central

são dados pelos atributos 17, 18, 19 e 20. Se você quiser, pode usar apenas esses quatro atributos, ignorando os outros. Isso evita o problema que surge quando uma vizinhança 3x3 atravessa um limite.

O banco de dados se encontra no pacote **mlbench** e é completo (não possui dados faltantes).

Tarefas:

1. Treine modelos RandomForest, SVM e RNA para predição destes dados.
2. Escolha o melhor modelo com base em suas matrizes de confusão.
3. Treine o modelo final com todos os dados e faça a predição na base completa.
4. Analise o resultado.
5. Salve este modelo final

2 Estimativa de Volumes de Árvores

Modelos de aprendizado de máquina são bastante usados na área da engenharia florestal (mensuração florestal) para, por exemplo, estimar o volume de madeira de árvores sem ser necessário abatê-las.

O processo é feito pela coleta de dados (dados observados) através do abate de algumas árvores, onde sua altura, diâmetro na altura do peito (dap), etc, são medidos de forma exata. Com estes dados, treina-se um modelo de AM que pode estimar o volume de outras árvores da população.

Os modelos, chamados **alométricos**, são usados na área há muitos anos e são baseados em regressão (linear ou não) para encontrar uma equação que descreve os dados. Por exemplo, o modelo de Spurr é dado por:

$$\text{Volume} = b_0 + b_1 * dap^2 * H_t$$

Onde **dap** é o diâmetro na altura do peito (1,3metros), $\mathbf{H_t}$ é a altura total. Tem-se vários modelos alométricos, cada um com uma determinada característica, parâmetros, etc. Um modelo de regressão envolve aplicar os dados observados e encontrar b_0 e b_1 no modelo apresentado, gerando assim uma equação que pode ser usada para prever o volume de outras árvores.

Dado o arquivo **Volumes.csv**, que contém os dados de observação, escolha um modelo de aprendizado de máquina com a melhor estimativa, a partir da estatística de correlação.

Tarefas

1. Carregar o arquivo **Volumes.csv**
   (**http://www.razer.net.br/datasets/Volumes.csv**)
2. Eliminar a coluna NR, que só apresenta um número sequencial
3. Criar partição de dados: treinamento 80%, teste 20%
4. Usando o pacote "caret", treinar os modelos: Random Forest (rf), SVM (svmRadial), Redes Neurais (neuralnet) e o modelo alométrico de SPURR
5. O modelo alométrico é dado por: Volume = b_0 + b_1 * dap^2 * H_t

```
alom <- nls(VOL ~ b0 + b1*DAP*DAP*HT, dados, start=list(b0=0.5, b1=0.5))
```

6. Efetue as predições nos dados de teste
7. Crie funções e calcule as seguintes métricas entre a predição e os dados observados
   - Coeficiente de determinação: R^2

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

onde y_i é o valor observado, $\widehat{y_i}$ é o valor predito e $\bar{y}$ é a média dos valores y_i observados. Quanto mais perto de 1 melhor é o modelo;

   - Erro padrão da estimativa: S_{yx}

$$S_{yx} = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \widehat{y_i})^2}{n-2}}$$

esta métrica indica erro, portanto quanto mais perto de 0 melhor é o modelo;

   - S_{yx}%

$$S_{yx}\% = \frac{S_{yx}}{\bar{y}} * 100$$

esta métrica indica porcentagem de erro, portanto quanto mais perto de 0 melhor é o modelo;

8. Escolha o melhor modelo

EXERCÍCIO 1(Satélites)

```
library("mlbench")
data(Satellite)
dataset <- Satellite
head(dataset)
```

```
  x.1 x.2 x.3 x.4 x.5 x.6 x.7 x.8 x.9 x.10 x.11 x.12 x.13 x.14
1  92 115 120  94  84 102 106  79  84  102  102   83  101  126
2  84 102 106  79  84 102 102  83  80  102  102   79   92  112
3  84 102 102  83  80 102 102  79  84   94  102   79   84  103
4  80 102 102  79  84  94 102  79  80   94   98   76   84   99
5  84  94 102  79  80  94  98  76  80  102  102   79   84   99
6  80  94  98  76  80 102 102  79  76  102  102   79   76   99
  x.15 x.16 x.17 x.18 x.19 x.20 x.21 x.22 x.23 x.24 x.25 x.26
1  133  103   92  112  118   85   84  103  104   81  102  126
2  118   85   84  103  104   81   84   99  104   78   88  121
3  104   81   84   99  104   78   84   99  104   81   84  107
4  104   78   84   99  104   81   76   99  104   81   84   99
5  104   81   76   99  104   81   76   99  108   85   84   99
6  104   81   76   99  108   85   76  103  118   88   84  103
  x.27 x.28 x.29 x.30 x.31 x.32 x.33 x.34 x.35 x.36   classes
1  134  104   88  121  128  100   84  107  113   87 grey soil
2  128  100   84  107  113   87   84   99  104   79 grey soil
3  113   87   84   99  104   79   84   99  104   79 grey soil
4  104   79   84   99  104   79   84  103  104   79 grey soil
5  104   79   84  103  104   79   79  107  109   87 grey soil
6  104   79   79  107  109   87   79  107  109   87 grey soil
```

```
> tail(dataset)
     x.1 x.2 x.3 x.4 x.5 x.6 x.7 x.8 x.9 x.10 x.11 x.12 x.13 x.14 x.15 x.16 x.17 x.18 x.19 x.20 x.21 x.22 x.23 x.24 x.25
6430  84 116 128 103  92 116 133 103  84  112  122   96   79  103  123  100   84  111  128  100   84  103  118   92   66
6431  60  83  96  85  64  87 100  88  64   83  104   88   59   83  100   83   63   87  100   87   63   83  104   87   66
6432  64  79 100  85  56  71  96  85  56   68   91   81   63   79  100   87   59   75   96   87   59   72   96   83   63
6433  56  68  91  81  56  64  91  81  53   64   83   78   59   72   96   83   59   75   96   75   59   75   89   75   63
6434  56  68  87  74  60  71  91  81  60   64  104   99   59   79   89   71   63   79   93   75   63   68  109   92   59
6435  60  71  91  81  60  64 104  99  56   64  108   96   63   79   93   75   63   68  109   92   59   75  109   96   59
     x.26 x.27 x.28 x.29 x.30 x.31 x.32 x.33 x.34 x.35 x.36            classes
6430   71  100   85   74   83  104   92   78   96  112   96           red soil
6431   91  104   92   66   87  108   89   63   83  104   85           red soil
6432   83  100   85   66   83  100   85   63   83  100   81           red soil
6433   83  100   81   59   87   96   81   63   83   92   74 vegetation stubble
6434   83   96   74   59   83   92   74   59   83   92   70 vegetation stubble
6435   83   92   74   59   83   92   70   63   79  108   92 vegetation stubble
```

Random forest

```
> indices <- createDataPartition(dataset$classes, p=0.8, list=F)
> treino <- dataset[indices,]
> teste <- dataset[-indices,]
```

```
> set.seed(1)
>
> rf <- train(classes ~ x.17 + x.18 + x.19 + x.20, data=treino, method="rf")
1 package is needed and is not installed. (randomForest). Would you like to try to install it now?
1: yes
2: no

Selection:
Enter an item from the menu, or 0 to exit
Selection: rf <- train(classes ~ x.17 + x.18 + x.19 + x.20, data=treino, method="rf")
Enter an item from the menu, or 0 to exit
Selection:
Enter an item from the menu, or 0 to exit
Selection: 1
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/randomForest_4.7-1.1.zip'
Content type 'application/zip' length 222290 bytes (217 KB)
downloaded 217 KB

package 'randomForest' successfully unpacked and MD5 sums checked
```

```
> predicao.rf <- predict(rf, teste)
>
> confusionMatrix(predicao.rf, teste$classes)
Confusion Matrix and Statistics

                    Reference
Prediction           red soil cotton crop grey soil damp grey soil vegetation stubble very damp grey soil
  red soil                300           0         2              2                   7                    0
  cotton crop               0         133         0              0                   4                    0
  grey soil                 3           0       253             28                   0                    3
  damp grey soil            1           0        12             60                   0                   40
  vegetation stubble        2           6         1              1                 115                    5
  very damp grey soil       0           1         3             34                  15                  253

Overall Statistics

               Accuracy : 0.8676
                 95% CI : (0.8478, 0.8857)
    No Information Rate : 0.2383
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8361

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: red soil Class: cotton crop Class: grey soil Class: damp grey soil Class: vegetation stubble
Sensitivity               0.9804            0.9500           0.9336               0.48000                   0.81560
Specificity               0.9888            0.9965           0.9664               0.95427                   0.98688
Pos Pred Value            0.9646            0.9708           0.8815               0.53097                   0.88462
Neg Pred Value            0.9938            0.9939           0.9819               0.94449                   0.97747
Prevalence                0.2383            0.1090           0.2111               0.09735                   0.10981
Detection Rate            0.2336            0.1036           0.1970               0.04673                   0.08956
Detection Prevalence      0.2422            0.1067           0.2235               0.08801                   0.10125
Balanced Accuracy         0.9846            0.9733           0.9500               0.71714                   0.90124
                     Class: very damp grey soil
Sensitivity                              0.8405
Specificity                              0.9461
Pos Pred Value                           0.8268
Neg Pred Value                           0.9509
Prevalence                               0.2344
Detection Rate                           0.1970
Detection Prevalence                     0.2383
Balanced Accuracy                        0.8933
```

Avaliação do resultado: precisão do modelo foi de 86,76%, o que é considerado positivo, pois indica que o modelo não está sofrendo de *overfitting*. No entanto, o modelo enfrentou dificuldades em reconhecer a classe de solo cinza úmido.

SVM

```
>
> svm <- train(classes ~ x.17 + x.18 + x.19 + x.20, data=treino, method="svmRadial")
>
> predicao.svm <- predict(svm, teste)
>
> confusionMatrix(predicao.svm, teste$classes)
Confusion Matrix and Statistics

                    Reference
Prediction           red soil cotton crop grey soil damp grey soil vegetation stubble very damp grey soil
  red soil                301           0         0              2                   8                   0
  cotton crop               0         129         0              0                   3                   0
  grey soil                 3           0       266             32                   0                   4
  damp grey soil            1           0         5             59                   0                  43
  vegetation stubble        1           9         0              1                 114                   5
  very damp grey soil       0           2         0             31                  16                 249

Overall Statistics

               Accuracy : 0.8707
                 95% CI : (0.8511, 0.8886)
    No Information Rate : 0.2383
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8398

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: red soil Class: cotton crop Class: grey soil Class: damp grey soil Class: vegetation stubble
Sensitivity                   0.9837             0.9214           0.9815                0.47200                    0.80851
Specificity                   0.9898             0.9974           0.9615                0.95772                    0.98600
Pos Pred Value                0.9678             0.9773           0.8721                0.54630                    0.87692
Neg Pred Value                0.9949             0.9905           0.9949                0.94388                    0.97660
Prevalence                    0.2383             0.1090           0.2111                0.09735                    0.10981
Detection Rate                0.2344             0.1005           0.2072                0.04595                    0.08879
Detection Prevalence          0.2422             0.1028           0.2375                0.08411                    0.10125
Balanced Accuracy             0.9867             0.9594           0.9715                0.71486                    0.89726
                     Class: very damp grey soil
Sensitivity                              0.8272
Specificity                              0.9502
Pos Pred Value                           0.8356
Neg Pred Value                           0.9473
Prevalence                               0.2344
Detection Rate                           0.1939
Detection Prevalence                     0.2321
Balanced Accuracy                        0.8887
```

Avaliação do resultado: O modelo obteve acurácia de 87,07% o que é algo positivo, pois indica que o modelo não está sofrendo de overfitting. O modelo encontrou menor dificuldade em reconhecer a classe damp grey soil.

RNA

```
set.seed(1)
```

```
rna <- train(classes ~ x.17 + x.18 + x.19 + x.20, data=treino, method="nnet")
```

```
iter  10 value 8366.158041
iter  20 value 6299.915271
iter  30 value 5234.190352
iter  40 value 4843.733474
iter  50 value 4487.058982
iter  60 value 4221.942090
iter  70 value 4033.316695
iter  80 value 3769.913802
iter  90 value 3539.180427
iter 100 value 3210.422102
final  value 3210.422102
stopped after 100 iterations

> predicao.rna <- predict(rna, teste)
>
> confusionMatrix(predicao.rna, teste$classes)
Confusion Matrix and Statistics

                    Reference
Prediction           red soil cotton crop grey soil damp grey soil vegetation stubble very damp grey soil
  red soil                295           1         1              0                   5                   0
  cotton crop               2         130         0              0                  10                   0
  grey soil                 5           0       239             35                   0                  11
  damp grey soil            0           0         0              0                   0                   0
  vegetation stubble        3           7         0              2                 110                   9
  very damp grey soil       1           2        31             88                  16                 281

Overall Statistics

               Accuracy : 0.8217
                 95% CI : (0.7996, 0.8422)
    No Information Rate : 0.2383
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7761

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: red soil Class: cotton crop Class: grey soil Class: damp grey soil Class: vegetation stub
Sensitivity                   0.9641             0.9286           0.8819              0.00000                   0.78
Specificity                   0.9928             0.9895           0.9497              1.00000                   0.98
Pos Pred Value                0.9768             0.9155           0.8241                  NaN                   0.83
Neg Pred Value                0.9888             0.9912           0.9678              0.90265                   0.97
Prevalence                    0.2383             0.1090           0.2111              0.09735                   0.10
Detection Rate                0.2298             0.1012           0.1861              0.00000                   0.08
Detection Prevalence          0.2352             0.1106           0.2259              0.00000                   0.10
Balanced Accuracy             0.9784             0.9590           0.9158              0.50000                   0.88
                     Class: very damp grey soil
Sensitivity                              0.9336
Specificity                              0.8596
Pos Pred Value                           0.6706
Neg Pred Value                           0.9769
Prevalence                               0.2344
Detection Rate                           0.2188
Detection Prevalence                     0.3263
Balanced Accuracy                        0.8966
> .
```

Avaliação do resultado: O modelo obteve acurácia de 82,17% o que é algo positivo, pois indica que o modelo não está sofrendo de overfitting. O modelo encontrou maior dificuldade em reconhecer a classe damp grey soil.

Resultado: Melhor modelo apresentado foi o SVM devido sua acurácia.

```
> print(svm)
Support Vector Machines with Radial Basis Function Kernel

5151 samples
   4 predictor
   6 classes: 'red soil', 'cotton crop', 'grey soil', 'damp grey soil', 'vegetation stubble', 'very damp grey soil'

No pre-processing
Resampling: Bootstrapped (25 reps)
Summary of sample sizes: 5151, 5151, 5151, 5151, 5151, 5151, ...
Resampling results across tuning parameters:

  C     Accuracy   Kappa
  0.25  0.8573089  0.8230716
  0.50  0.8595635  0.8258808
  1.00  0.8615429  0.8283218

Tuning parameter 'sigma' was held constant at a value of 0.8970709
Accuracy was used to select the optimal model using the largest value.
The final values used for the model were sigma = 0.8970709 and C = 1.
```

```
> library(kernlab)
warning message:
package 'kernlab' was built under R version 4.2.2
>
> modelo_final <- ksvm(classes~x.17+x.18+x.19+x.20,
+                      data=dataset,
+                      type="C-svc",
+                      kernel="rbfdot",
+                      C=1.0,
+                      kpar=list(sigma=0.8788543)
+ )
```

```
> confusionMatrix(predicao_final.svm, dataset$classes)
Confusion Matrix and Statistics

                    Reference
Prediction           red soil cotton crop grey soil damp grey soil vegetation stubble very damp grey soil
  red soil               1499          2        10              5                48                   0
  cotton crop               2        637         0              1                 9                   2
  grey soil                19          0      1302            161                 3                  49
  damp grey soil            0          6        41            309                 4                 165
  vegetation stubble       12         48         0              4               571                  29
  very damp grey soil       1         10         5            146                72                1263

Overall Statistics

               Accuracy : 0.8673
                 95% CI : (0.8588, 0.8755)
    No Information Rate : 0.2382
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.8355

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: red soil Class: cotton crop Class: grey soil Class: damp grey soil Class: vegetation stubble
Sensitivity                   0.9778            0.90612           0.9588              0.49361                   0.80764
Specificity                   0.9867            0.99756           0.9543              0.96282                   0.98376
Pos Pred Value                0.9584            0.97849           0.8488              0.58857                   0.85994
Neg Pred Value                0.9930            0.98859           0.9886              0.94636                   0.97643
Prevalence                    0.2382            0.10925           0.2110              0.09728                   0.10987
Detection Rate                0.2329            0.09899           0.2023              0.04802                   0.08873
Detection Prevalence          0.2430            0.10117           0.2384              0.08159                   0.10319
Balanced Accuracy             0.9823            0.95184           0.9565              0.72821                   0.89570
                     Class: very damp grey soil
Sensitivity                              0.8375
Specificity                              0.9525
Pos Pred Value                           0.8437
Neg Pred Value                           0.9504
Prevalence                               0.2343
Detection Rate                           0.1963
Detection Prevalence                     0.2326
Balanced Accuracy                        0.8950
```

Avaliação do resultado: O modelo obteve acurácia de 86,73% o que é algo positivo pois indica que o modelo não está sofrendo de overfitting. Foi o melhor modelo obtido até o momento, mas isso deve ser devido a utilizar toda a base para treino e teste.

```
saveRDS(modelo_final, "satelites_svm.rds")
```

EXERCÍCIO 2(Arvores)

```
> library("caret")
Carregando pacotes exigidos: ggplot2
Carregando pacotes exigidos: lattice
Warning message:
package 'caret' was built under R version 4.2.3
>
> df <- read.csv("http://www.razer.net.br/datasets/Volumes.csv",sep=";", dec=",")
> df$NR <- NULL
> head(df)
   DAP    HT   HP       VOL
1 34.0 27.00 1.80 0.8971441
2 41.5 27.95 2.75 1.6204441
3 29.6 26.35 1.15 0.8008181
4 34.3 27.15 1.95 1.0791682
5 34.5 26.20 1.00 0.9801112
6 29.9 27.10 1.90 0.9067022
>
```

```
> tail(df)
     DAP    HT   HP       VOL
95  31.5 23.50 2.50 0.9221653
96  33.1 25.75 2.65 1.0966956
97  31.0 25.70 2.60 1.0514350
98  43.0 27.90 2.70 2.0090605
99  40.0 24.20 1.10 1.7411209
100 38.0 27.65 2.45 1.5336724
```

```
> indices <- createDataPartition(df$VOL, p=0.8, list=F)
> treino <- df[indices,]
> teste <- df[-indices,]
```

random forest

```
> set.seed(1)
>
> rf <- train(VOL ~ ., data=treino, method="rf")
note: only 2 unique complexity parameters in default grid. Truncating the grid to 2 .
```

```
> predicao.rf <- predict(rf, teste)
>
> summary(predicao.rf)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.8508  1.0936  1.2212  1.3690  1.6106  2.3515
```

SVM

```
> set.seed(1)
>
> svm <- train(VOL ~ ., data=treino, method="svmRadial")
> predicao.svm <- predict(svm, teste)
```

Redes neurais

```
> set.seed(1)
>
> install.packages(neuralnet)
Error in install.packages : object 'neuralnet' not found
> install.packages("neuralnet")
WARNING: Rtools is required to build R packages but is not currently installed. Please downlo
ersion of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
also installing the dependency 'Deriv'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/Deriv_4.1.3.zip'
Content type 'application/zip' length 148896 bytes (145 KB)
downloaded 145 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/neuralnet_1.44.2.zip'
Content type 'application/zip' length 123914 bytes (121 KB)
downloaded 121 KB

package 'Deriv' successfully unpacked and MD5 sums checked
package 'neuralnet' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\bruno\AppData\Local\Temp\RtmpMfSTTd\downloaded_packages
> library(neuralnet)
Warning message:
package 'neuralnet' was built under R version 4.2.3
>
> rna <- neuralnet(VOL ~ ., data=treino, hidden=c(2,1), linear.output=FALSE, threshold=0.01)
>
> rna <- neuralnet(VOL ~ ., data=treino, hidden=c(2,1), linear.output=FALSE, threshold=0.01)
> predicao.rna <- predict(rna, teste)
```

Modelo alométrico de SPURR

```
> set.seed(1)
>
> alom <- nls(VOL ~ b0 + b1*DAP*DAP*HT, data=treino, start=list(b0 = 0.5, b1=0.5))
> predicao.alom <- predict(alom, teste)
```

Criando funções para calcular métricas entre as predições dos dados observados e Erro padrão da estimativa com e sem porcentagem

```
> r2 <- function (valor_observado, valor_predito){
+     return(1-(sum((valor_observado-valor_predito)^2)/sum((valor_observado-mean(valor_observado))^2)))
+ }
>
> syx_porc <- function(valor_observado, valor_predito){
+     return(syx(valor_observado, valor_predito)/mean(valor_observado)*100)
+ }
> syx <- function(valor_observado, valor_predito){
+     return(sqrt(sum((valor_observado-valor_predito)^2)/(length(valor_observado)-2)))
+ }
```

Utilizando funções criadas para calcular métricas entre as predições dos dados observados

R^2

```
> r2(teste$VOL, predicao.rf)
[1] 0.8879647
> r2(teste$VOL, predicao.svm)
[1] 0.7985581
>
> r2(teste$VOL, predicao.rna)
[1] -0.8725625
>
> r2(teste$VOL, predicao.alom)
[1] 0.8654416
```

Avaliação: O modelo Random Forest se saiu melhor que os outros modelos, pois está mais próximo de 1. Nota: ainda não consegui interpretar o sinal negativo em RNA.

Erro padrão da estimativa:

```
> syx(teste$VOL, predicao.rf)
[1] 0.1491092
>
> syx(teste$VOL, predicao.svm)
[1] 0.199941
>
> syx(teste$VOL, predicao.rna)
[1] 0.6096002
>
> syx(teste$VOL, predicao.alom)
[1] 0.1634114
```

Avaliação: O Random Forest apresentou melhor desempenho em comparação aos demais, com um valor mais próximo de 0. Observação: A RNA obteve o pior resultado.

Erro padrão da estimativa em %

```
> syx_porc(teste$VOL, predicao.rf)
[1] 10.69305
>
> syx_porc(teste$VOL, predicao.svm)
[1] 14.33835
>
> syx_porc(teste$VOL, predicao.rna)
[1] 43.7162
>
> syx_porc(teste$VOL, predicao.alom)
[1] 11.71871
```

Após avaliar os resultados obtidos foi escolhido o modelo Random forest