# Atividade Avaliativa - Aprendizado de Máquina

Criação de um modelo preditor de classificação

Nomes: Andreza e Lincoln

# DATASET ESCOLHIDO - SEEDS

Os dados são de grãos pertencentes a três variedades diferentes de trigo: Kama, Rosa e Canadian;

Cada classe possui 70 elementos, selecionados aleatoriamente para o experimento.

A visualização de alta qualidade da estrutura interna do grão foi detectada usando uma técnica de raio-X suave que não é destrutiva e consideravelmente mais barata que outras técnicas de imagem mais sofisticadas, como microscopia de varredura ou tecnologia a laser. As imagens foram gravadas em placas KODAK de raios X de 13x18 cm.

Os estudos foram conduzidos usando grãos de trigo provenientes de campos experimentais do Instituto de Agrofísica da Academia Polonesa de Ciências de Lublin. Clique aqui para obter mais informações sobre o dataset.

# Atributos do dataset:

1. **A:** area
2. **P:** perimeter
3. **C:** compactness C = 4*pi*A/P^2,
4. **LK:** length of kernel
5. **WK:** width of kernel
6. **AssC:** asymmetry coefficient
7. **LK_G:** length of kernel groove
8. **Var:** varieties of wheat (Kama, Rosa and Canadian)

```
> percentage <- prop.table(table(df$Var)) * 100
> cbind(freq=table(df$Var), percentage=percentage)
  freq percentage
1   70   33.33333 ← Kama
2   70   33.33333 ← Rosa
3   70   33.33333 ← Canadian
>
```

- Quantidade de amostras iguais para as 3 classes de trigo, 70 de cada.

# ANÁLISE EXPLORATÓRIA DE DADOS

Do dataset de treinamento

```
> percentage <- prop.table(table(training$Var)) * 100
> cbind(freq=table(training$Var), percentage=percentage)
  freq percentage
1   58   34.52381 ← Kama
2   54   32.14286 ← Rosa
3   56   33.33333 ← Canadian
>
```

Amostras no conj de treinamento:

- kama: 58;
- rosa: 54;
- canadian: 56.

# STR

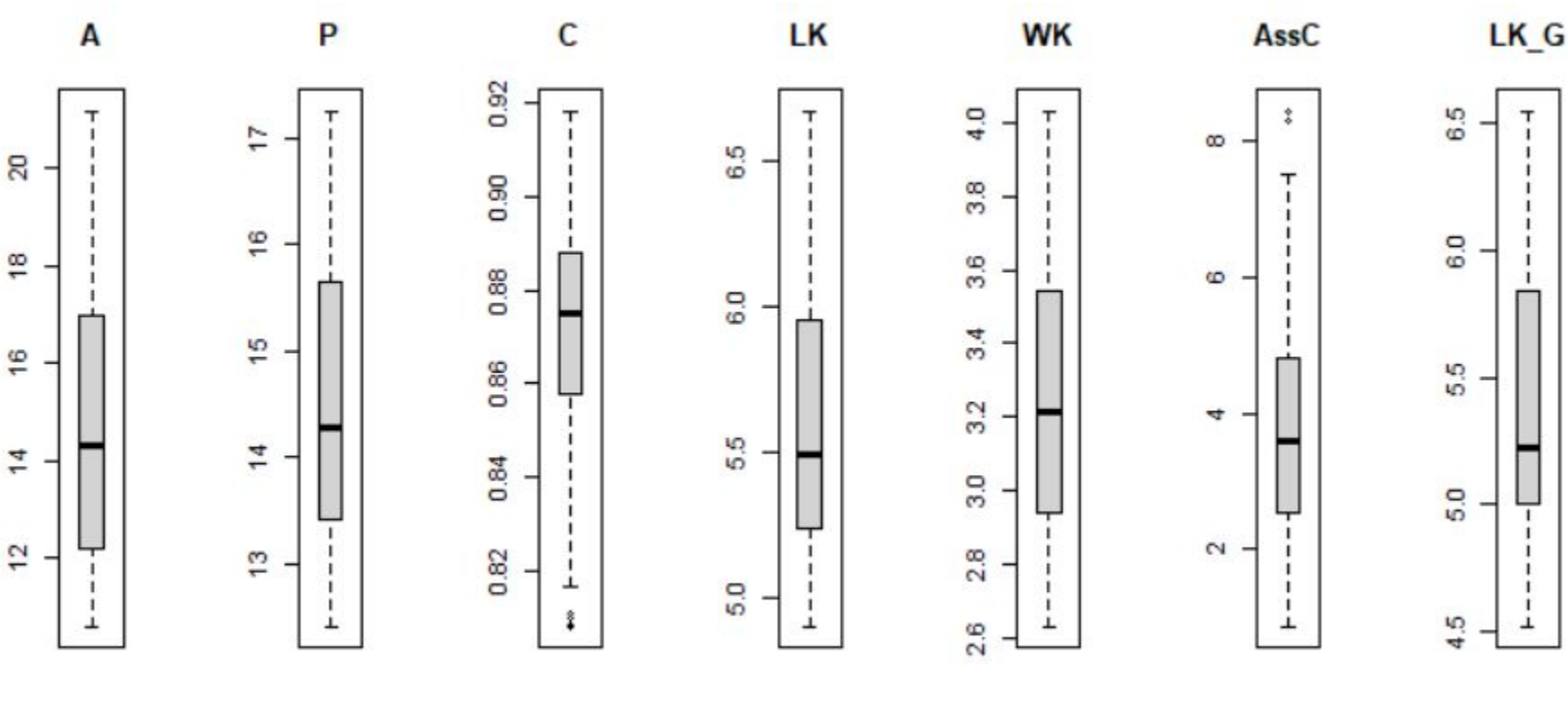
```
> str(training)
'data.frame':   168 obs. of  8 variables:
 $ A    : num  15.3 14.9 14.3 13.8 16.1 ...
 $ P    : num  14.8 14.6 14.1 13.9 15 ...
 $ C    : num  0.871 0.881 0.905 0.895 0.903 ...
 $ LK   : num  5.76 5.55 5.29 5.32 5.66 ...
 $ WK   : num  3.31 3.33 3.34 3.38 3.56 ...
 $ AssC: num  2.22 1.02 2.7 2.26 1.35 ...
 $ LK_G: num  5.22 4.96 4.83 4.8 5.17 ...
 $ Var : int  1 1 1 1 1 1 1 1 1 1 ...
>
```

- 168 instâncias e 8 atributos, sendo 7 de entrada e 1 de saída

# SUMARIZAÇÃO

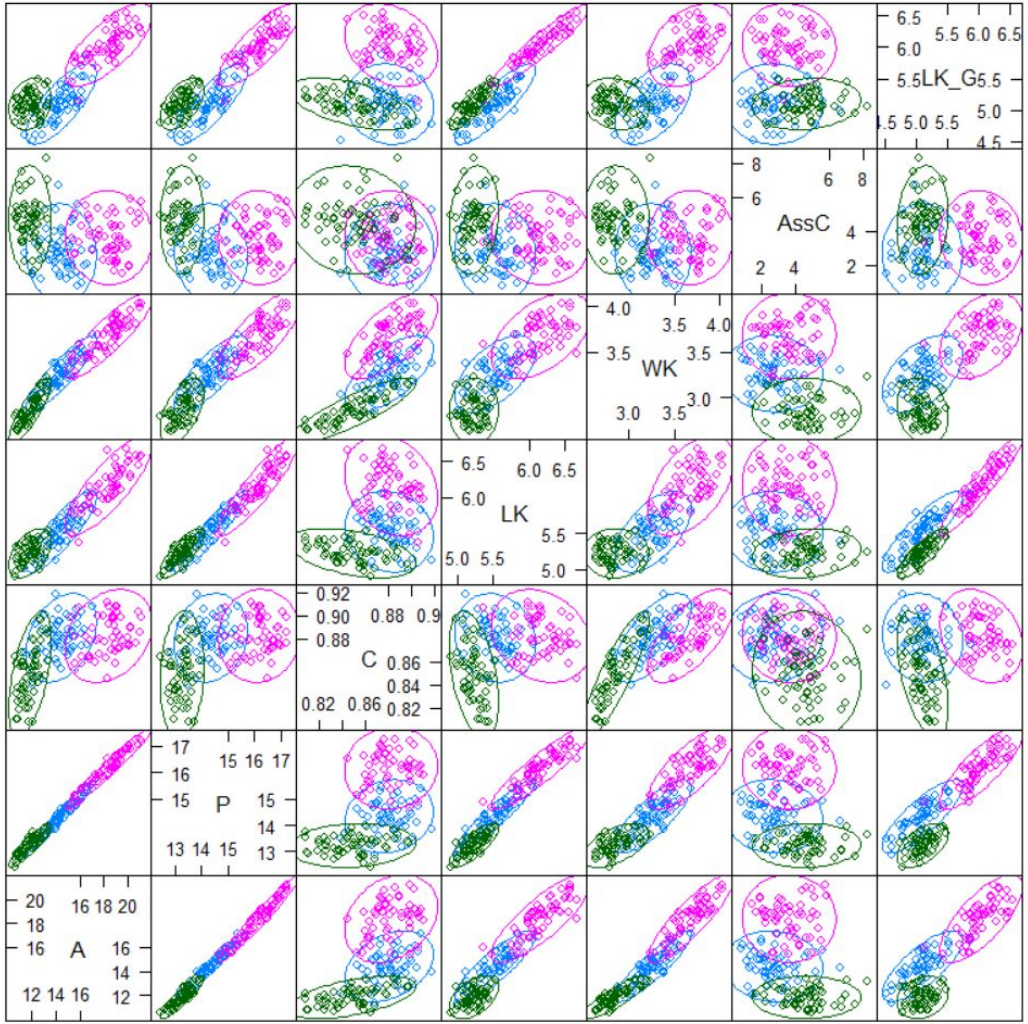
```
> summary(training)
       A               P               C               LK       
 Min.   :10.59   Min.   :12.41   Min.   :0.8081   Min.   :4.899  
 1st Qu.:12.19   1st Qu.:13.41   1st Qu.:0.8578   1st Qu.:5.239  
 Median :14.29   Median :14.27   Median :0.8749   Median :5.493  
 Mean   :14.73   Mean   :14.50   Mean   :0.8712   Mean   :5.608  
 3rd Qu.:16.92   3rd Qu.:15.63   3rd Qu.:0.8880   3rd Qu.:5.940  
 Max.   :21.18   Max.   :17.25   Max.   :0.9183   Max.   :6.675  
       WK              AssC            LK_G            Var       
 Min.   :2.630   Min.   :0.8551   Min.   :4.519   Min.   :1.000  
 1st Qu.:2.940   1st Qu.:2.5408   1st Qu.:5.003   1st Qu.:1.000  
 Median :3.216   Median :3.5920   Median :5.220   Median :2.000  
 Mean   :3.248   Mean   :3.7015   Mean   :5.383   Mean   :1.988  
 3rd Qu.:3.534   3rd Qu.:4.7860   3rd Qu.:5.838   3rd Qu.:3.000  
 Max.   :4.033   Max.   :8.4560   Max.   :6.550   Max.   :3.000  
> 
```

# Distribuição das variáveis independentes:



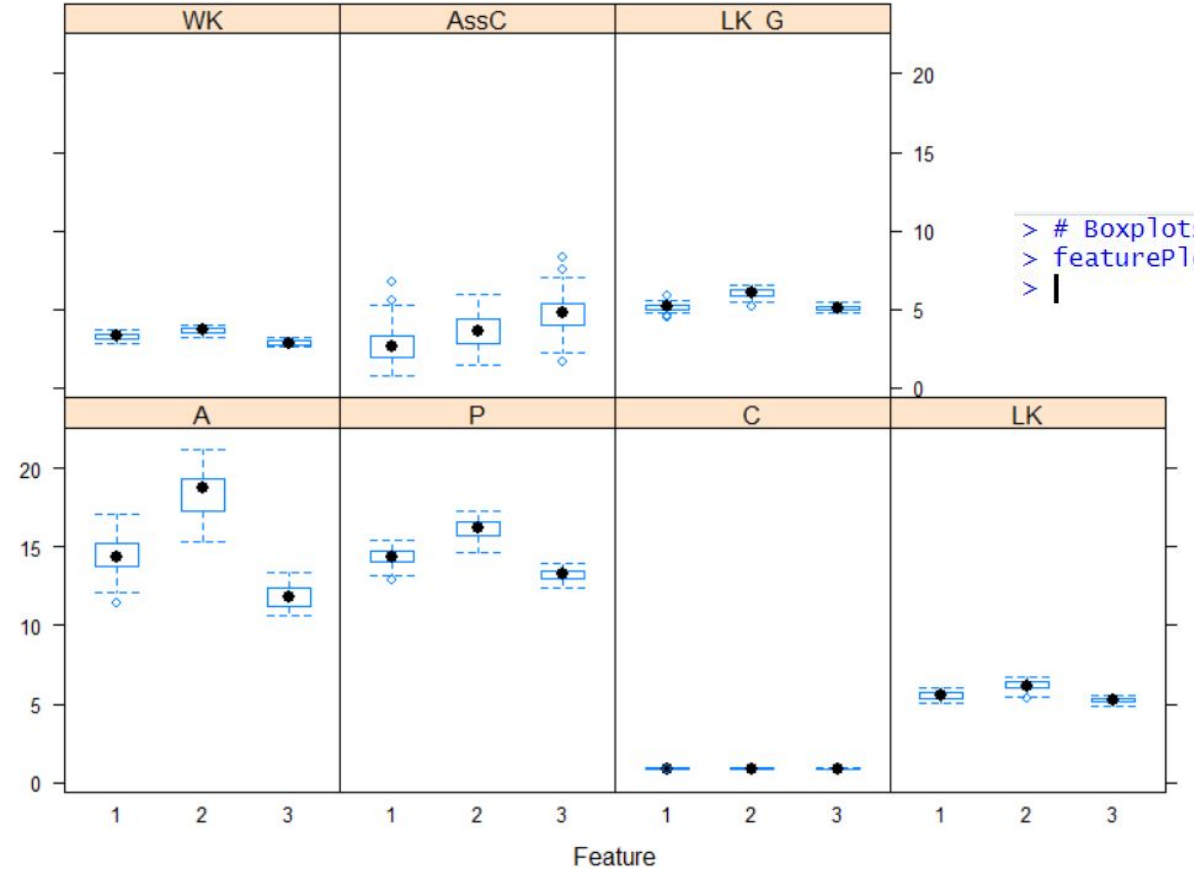
# Visualização dos dados em ScatterPlot:

```
> # Matriz de dispersão considerando atributos de entrada e saída
> featurePlot(x=x, y=y, plot="ellipse")
>
```



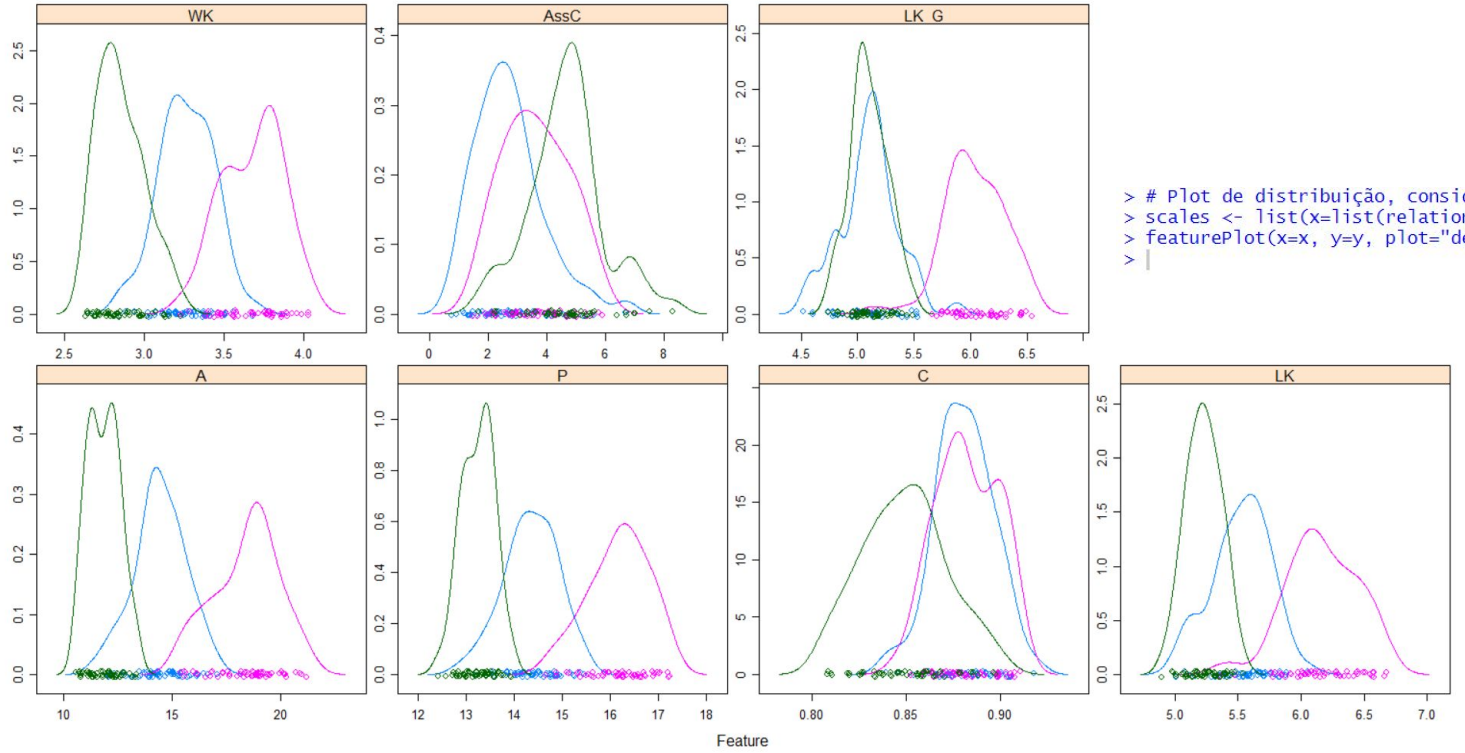

Scatter Plot Matrix

# Visualização dos dados em Boxplot:




```
> # Boxplots considerando atributos de entrada e saída
> featurePlot(x=x, y=y, plot="box")
>
```

# Visualização dos dados em Densidade:




```
> # Plot de distribuição, considerando atributos de entrada e saida
> scales <- list(x=list(relation="free"), y=list(relation="free"))
> featurePlot(x=x, y=y, plot="density", scales=scales)
>
```

# APLICAÇÃO DE ALGORITMOS DE CLASSIFICAÇÃO

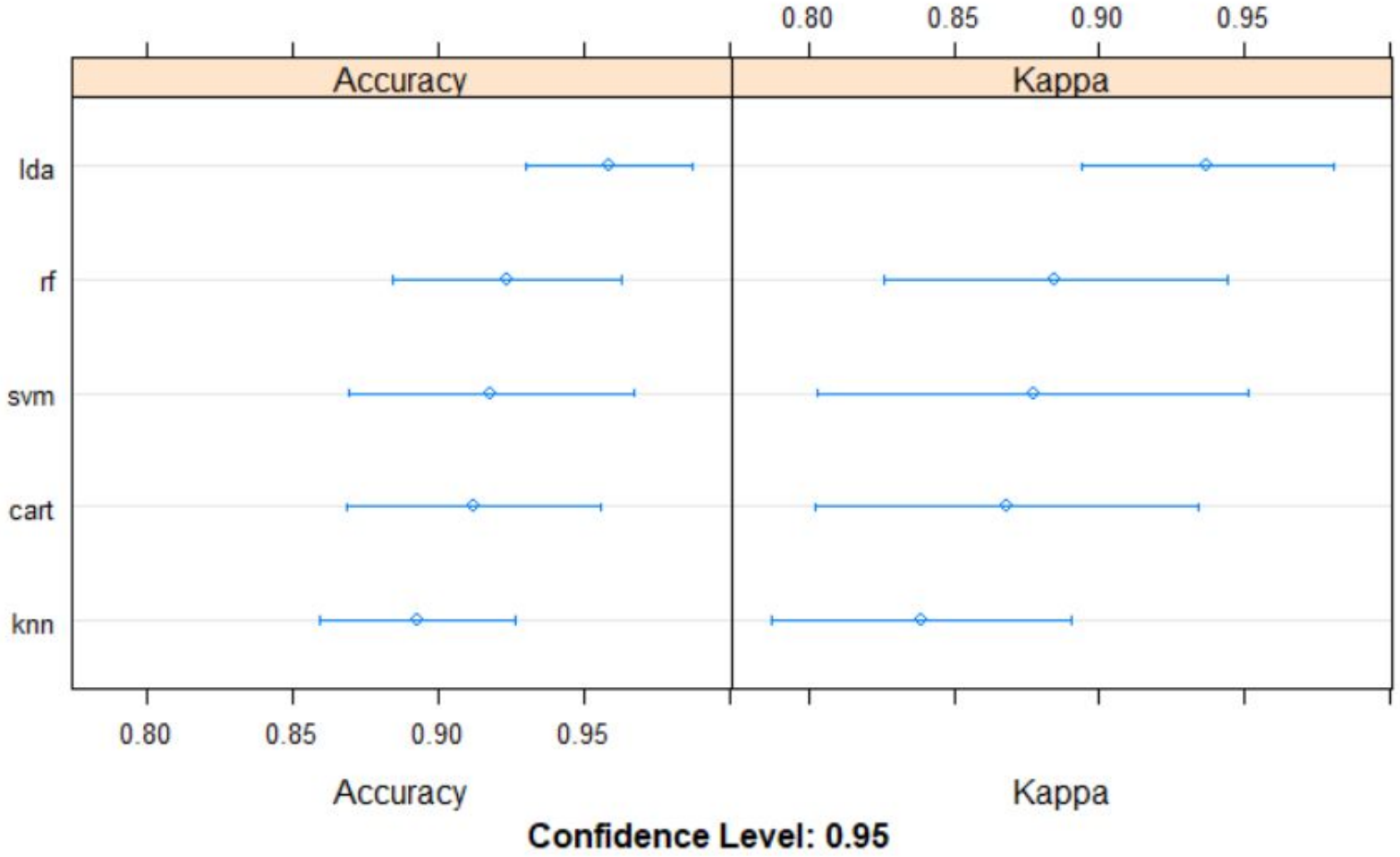
```r
# Uso de cross validation, com k-fold = 10
control <- trainControl(method="cv", number=10)
metric <- "Accuracy"

# --Treinamento e testes de algoritmos--

# Algoritmos lineares
set.seed(7)
fit.lda <- train(Var~., data=training, method="lda",
                 metric=metric, trControl=control)

# Algoritmos não lineares
set.seed(7)
fit.cart <- train(Var~., data=training, method="rpart",
                  metric=metric, trControl=control)
```

```r
# kNN
set.seed(7)
fit.knn <- train(Var~., data=training, method="knn",
                 metric=metric, trControl=control)

# SVM
set.seed(7)
fit.svm <- train(Var~., data=training, method="svmRadial",
                 metric=metric, trControl=control)

# Random Forest
set.seed(7)
fit.rf <- train(Var~., data=training, method="rf",
                metric=metric, trControl=control)
```

# COMPARAÇÃO ENTRE O DESEMPENHO DOS ALGORITMOS

```
> results <- resamples(list(lda=fit.lda, cart=fit.cart,
+                           knn=fit.knn, svm=fit.svm, rf=fit.rf))
> summary(results)

Call:
summary.resamples(object = results)

Models: lda, cart, knn, svm, rf
Number of resamples: 10

Accuracy
          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
lda  0.8823529 0.9384191 0.9428105 0.9584150 1.0000000 1.0000000    0
cart 0.8235294 0.8823529 0.9111111 0.9122467 0.9402574 1.0000000    0
knn  0.8125000 0.8823529 0.8856209 0.8928431 0.9364583 0.9444444    0
svm  0.8235294 0.8784722 0.9150327 0.9182190 0.9852941 1.0000000    0
rf   0.8235294 0.8839869 0.9375000 0.9234477 0.9411765 1.0000000    0

Kappa
          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
lda  0.8219895 0.9068587 0.9140625 0.9372224 1.0000000 1.0000000    0
cart 0.7357513 0.8217480 0.8666667 0.8681214 0.9102692 1.0000000    0
knn  0.7159763 0.8215186 0.8290378 0.8388789 0.9048246 0.9166667    0
svm  0.7343750 0.8171569 0.8721640 0.8771470 0.9779793 1.0000000    0
rf   0.7357513 0.8262090 0.9053254 0.8848471 0.9116865 1.0000000    0

> 
```

Visualização dos dados em DotPlot:



# Sumarização do melhor modelo:

```
> print(fit.lda)
Linear Discriminant Analysis

168 samples
  7 predictor
  3 classes: '1', '2', '3'

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 151, 150, 151, 151, 150, 151, ...
Resampling results:

  Accuracy  Kappa
  0.958415  0.9372224

> |
```

```
> # Testes usando matriz de confusão
> predictions <- predict(fit.lda, validation)
> confusionMatrix(predictions, validation$Var)
Confusion Matrix and Statistics

          Reference
Prediction  1  2  3
         1 13  0  0
         2  0 14  0
         3  1  0 14

Overall Statistics

               Accuracy : 0.9762
                 95% CI : (0.8743, 0.9994)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9643

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 1 Class: 2 Class: 3
Sensitivity            0.9286   1.0000   1.0000
Specificity            1.0000   1.0000   0.9643
Pos Pred Value         1.0000   1.0000   0.9333
Neg Pred Value         0.9655   1.0000   1.0000
Prevalence             0.3333   0.3333   0.3333
Detection Rate         0.3095   0.3333   0.3333
Detection Prevalence   0.3095   0.3333   0.3571
Balanced Accuracy      0.9643   1.0000   0.9821
>
```

Melhor algoritmo para o dataset seeds:

- Análise de Discriminante Linear

Acurácia do Modelo de Classificação:

- 0.9762

Classes:

1. Kama
2. Rosa
3. Canadian