HEADS

PhD PROGRAMME IN HEALTH DATA SCIENCE

Learning tree models

Decision trees and random forests

Machine Learning and Data Mining

PhD Programme in Health Data Science

Pedro Pereira Rodrigues

U.PORTO
FMUP FACULDADE DE MEDICINA
UNIVERSIDADE DO PORTO

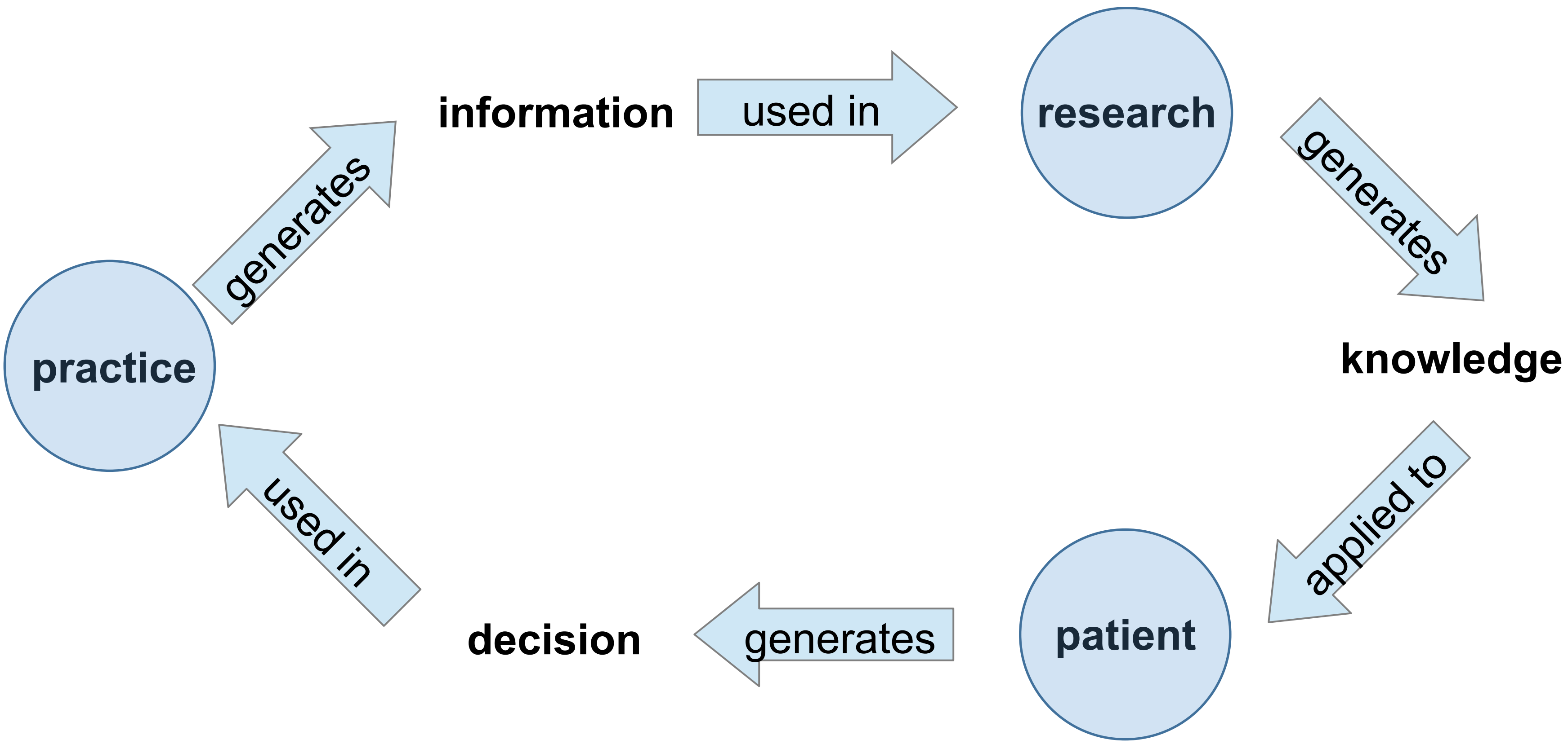
MEDCIDS
DEPARTAMENTO DE MEDICINA
DA COMUNIDADE, INFORMAÇÃO
E DECISÃO EM SAÚDE

CINTESIS
Health. Research.

Evidence Based Medicine

"Conscient, explicit and criterious use of the best available evidence in clinical decision"

Sackett D. (1996)


information
used in
research
generates
knowledge
applied to
patient
generates
decision
used in
practice
generates

Real-World Biomedical Data

“The complicated nature of real-world biomedical data has made it necessary to look beyond traditional biostatistics.”

Lucas P. (2004)

Wealth of Health Data

“The routine operation of modern healthcare systems produces a wealth of data in electronic health records, administrative databases, clinical registries, and other clinical systems.”

Peek & Rodrigues (2018)

Knowledge Discovery

"It is widely acknowledged that there is great potential for utilizing these routine data for health research to derive new knowledge about health, disease, and treatments."

Peek & Rodrigues (2018)

Data Science

"Study on creation, validation and transformation of data to generate meaning."

Data Science Association (2020)

Clinical Knowledge Representation

“Clinical cases are getting more and more complex, yielding the application of modelling techniques likewise increasingly complex.”

Lucas P. (2014)

Machine Learning

“The field of machine learning is concerned with question of how to construct computer programs that automatically improve with experience”

Mitchell (1997)

Supervised Machine Learning Metaphor

“There is a teacher who teaches the system a concept, with which the student is able to classify new cases, and there is an error function for that classification.”

Hastie T., Tibshirani R. & Friedman J. (2001)

Inductive Bias

An algorithm that learns automatically from a set of data looks for a hypothesis, in the space of possible hypotheses, that best fits the training data.

Each algorithm chooses a representation for this hypothesis.

- The chosen representation represents a **representation bias**
- The way the algorithm searches for the hypothesis represents a **search bias**

Inductive Bias

"A learner that makes no a priori assumptions regarding the identity of the target concept has no rational basis for classifying any unseen instances."

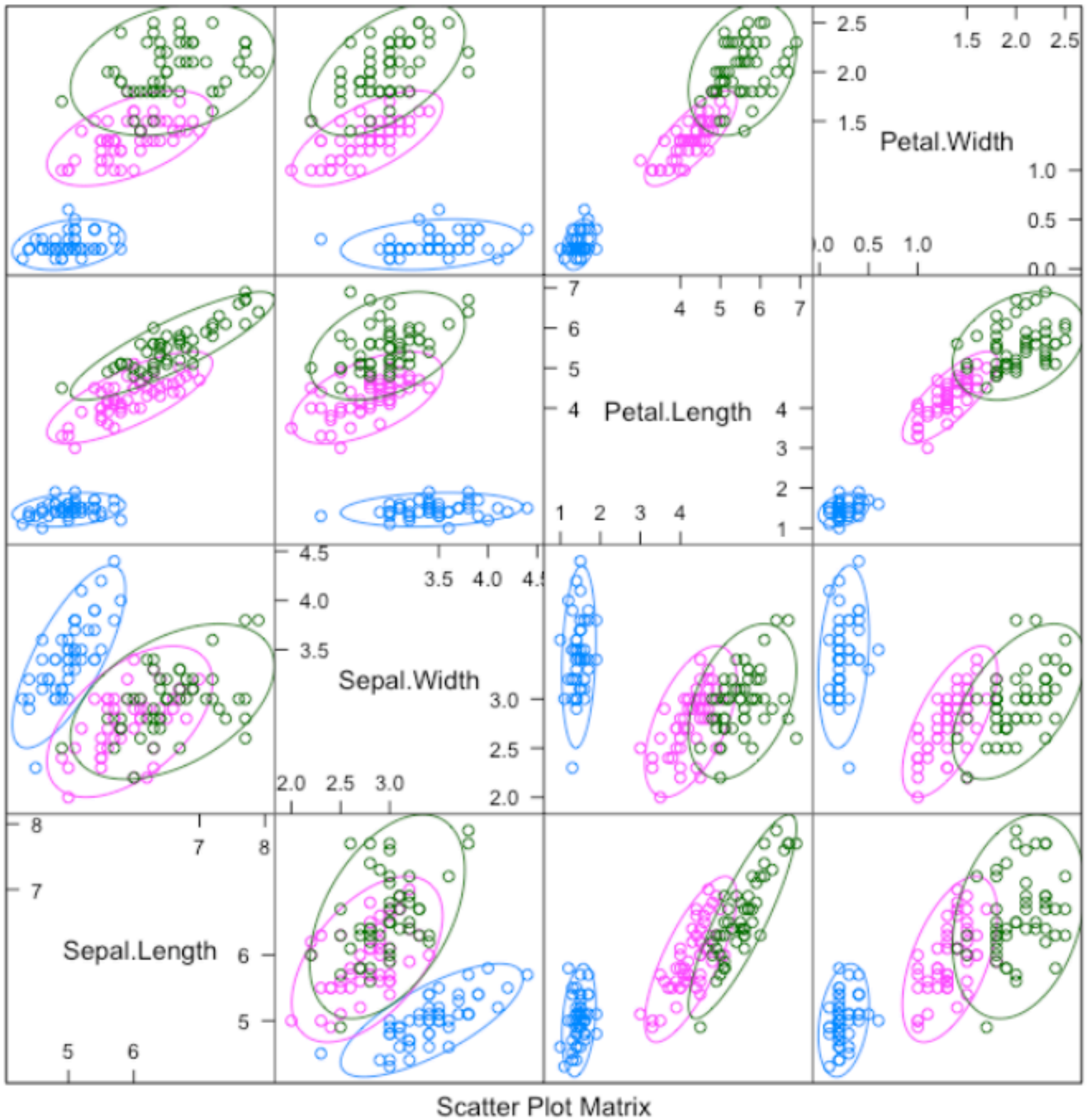
Mitchell (1997)

Black Boxes

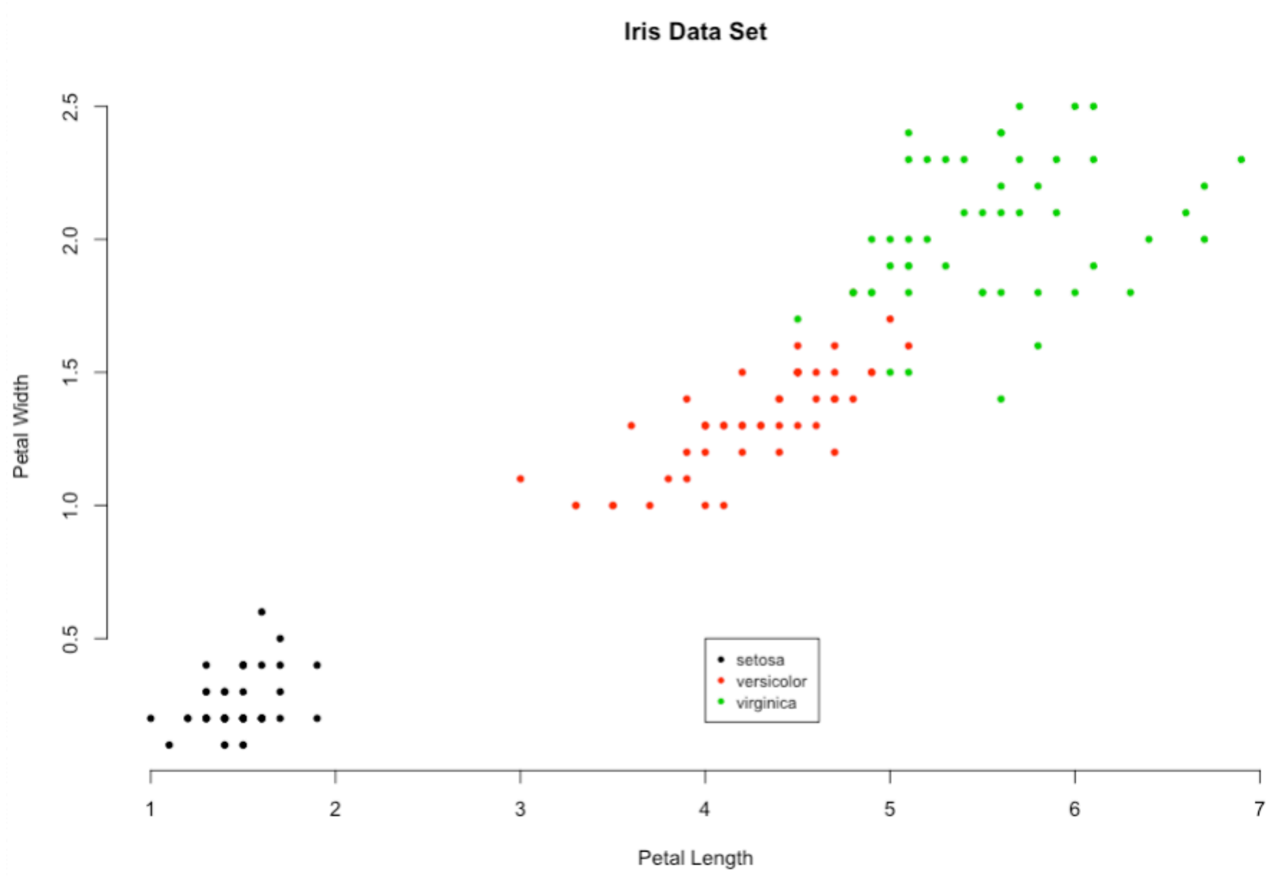
“Some machine learning techniques, although very successful from the accuracy point of view, are very opaque in terms of understanding how they make decisions.”

EU Commission (2019)

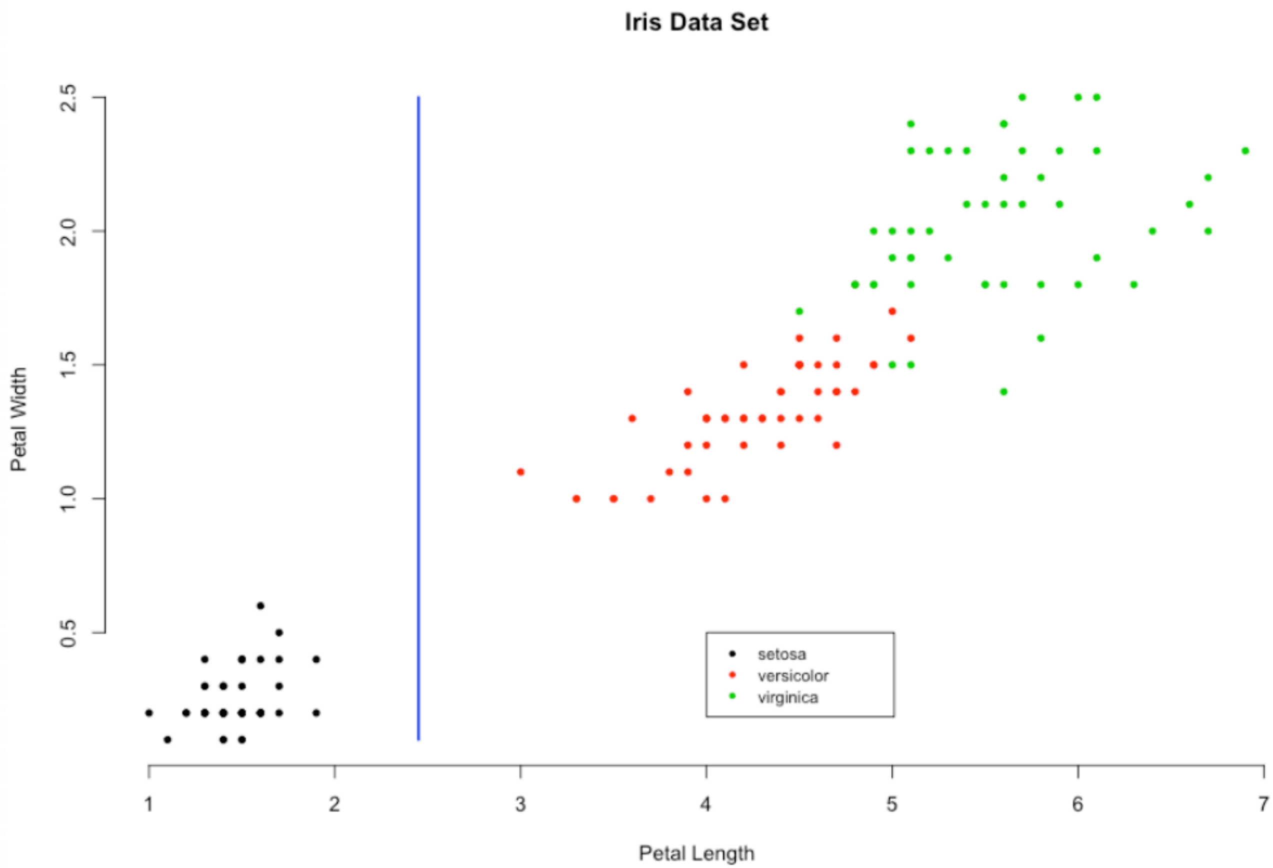
The Iris Data Set



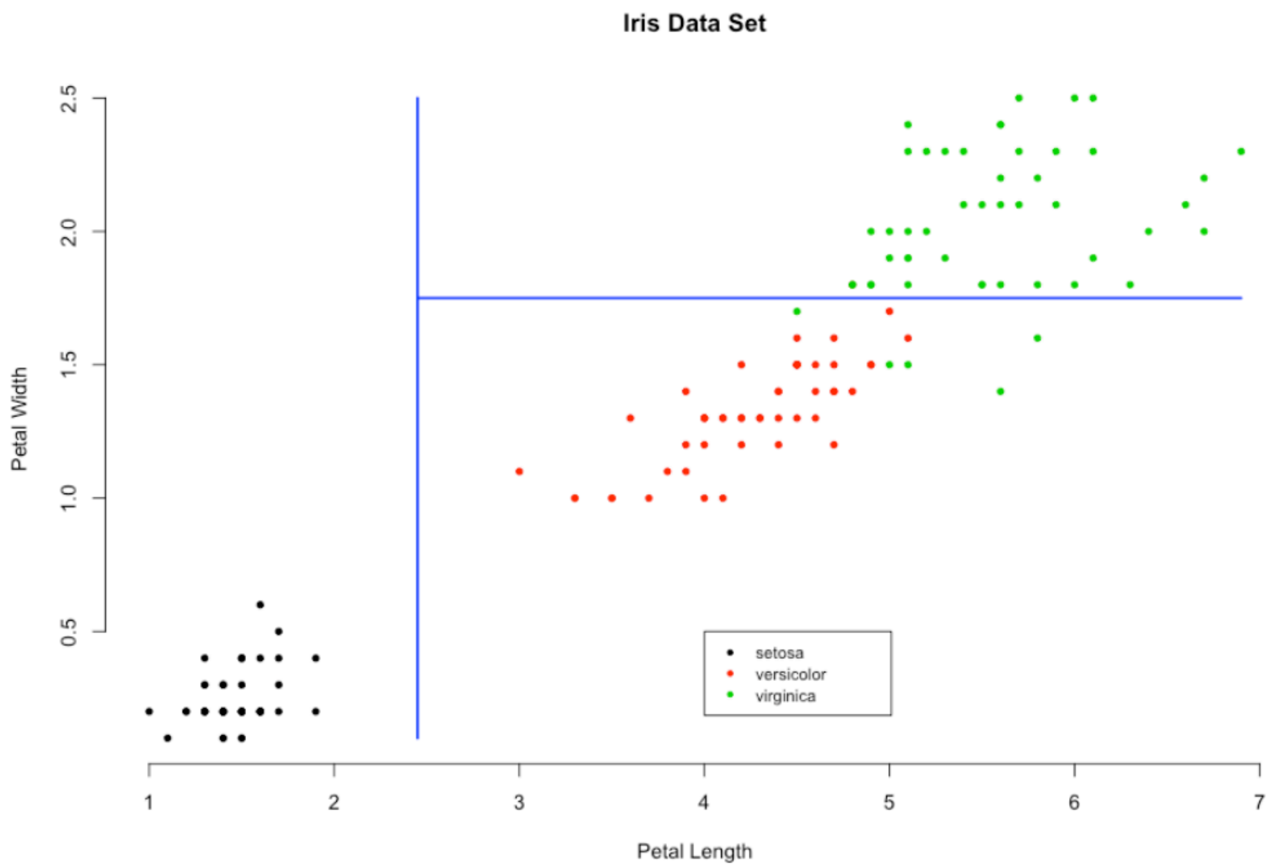
The Iris Data Set



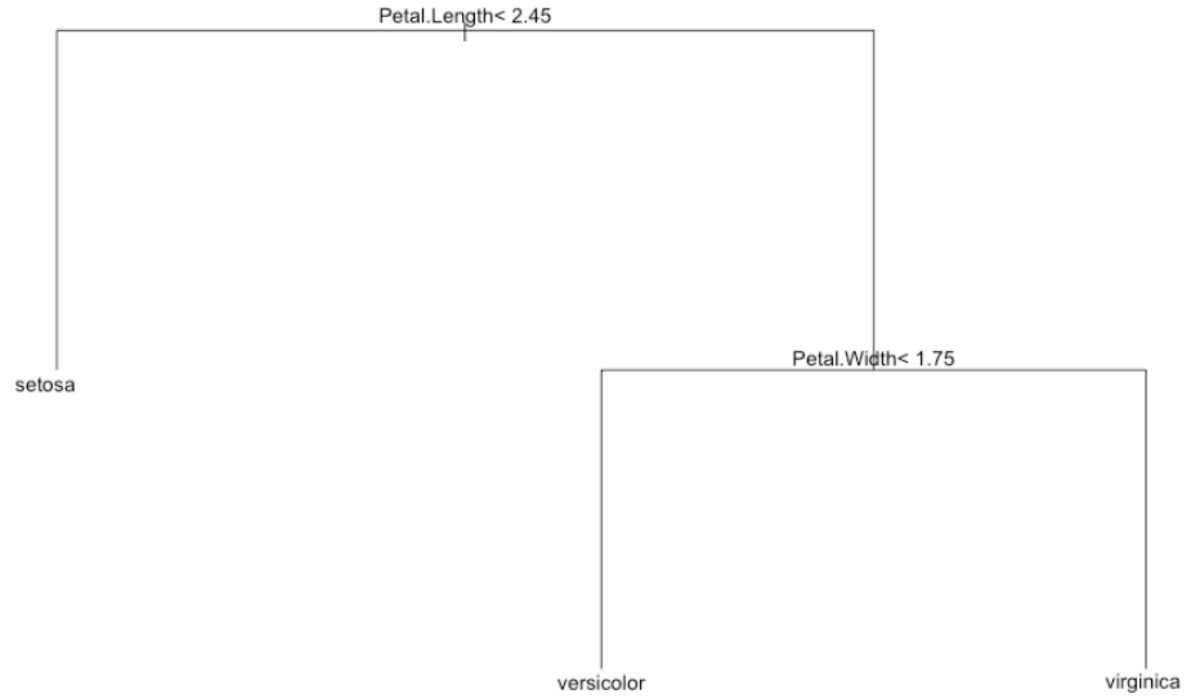
The Iris Data Set



The Iris Data Set

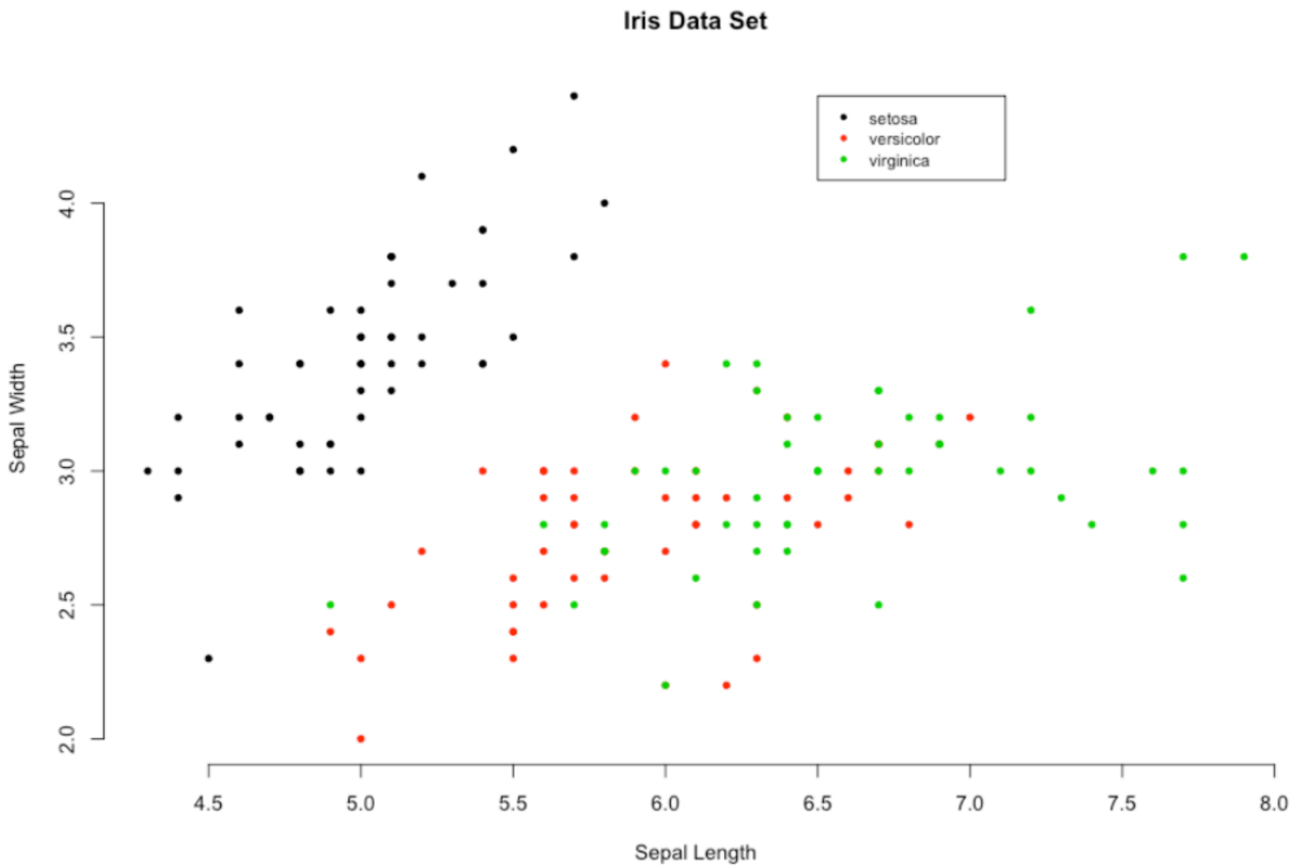


Decision Tree

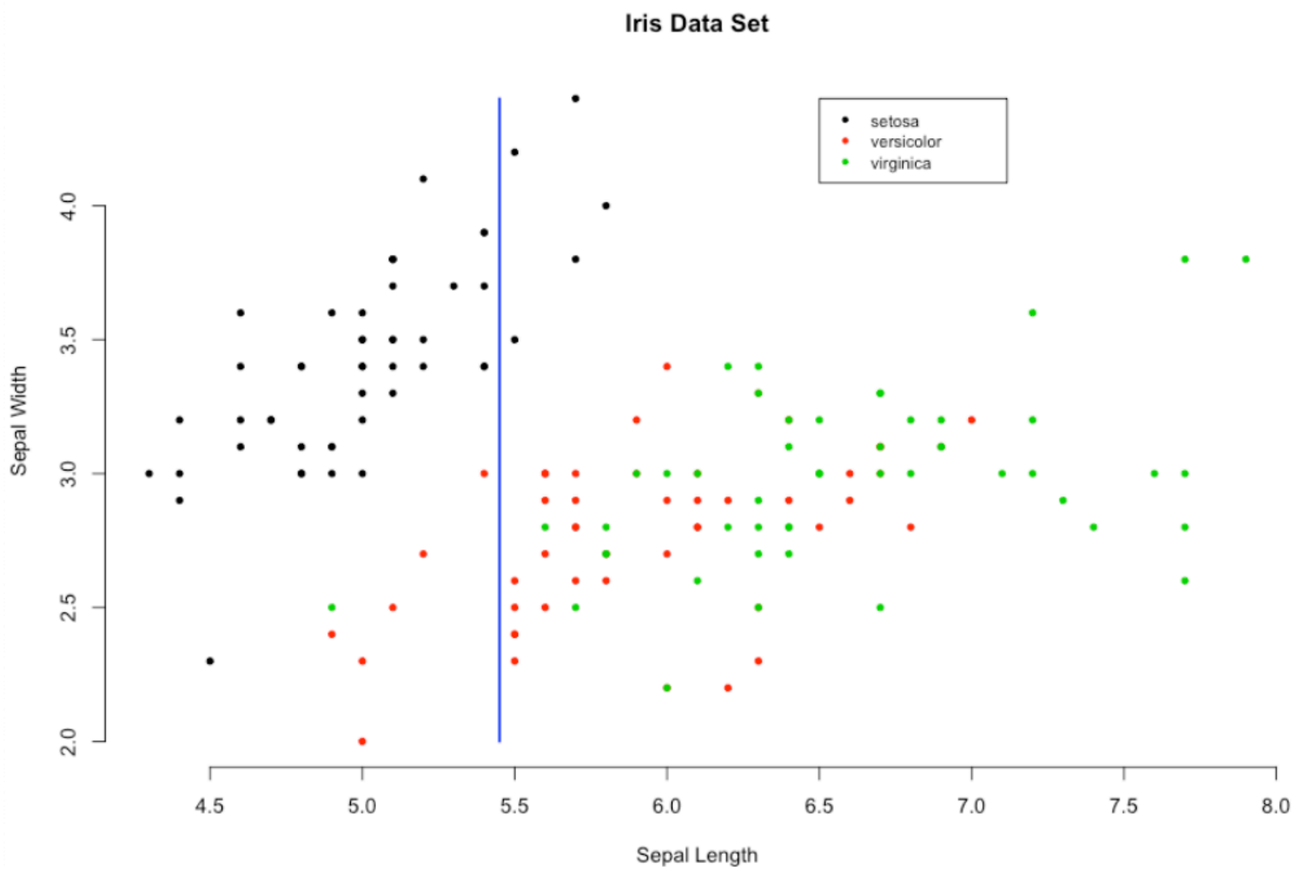



```
n= 150

node), split, n, loss, yval, (yprob)
      * denotes terminal node

1) root 150 100 setosa (0.33333333 0.33333333 0.33333333)
  2) Petal.Length< 2.45 50   0 setosa (1.00000000 0.00000000 0.00000000) *
  3) Petal.Length>=2.45 100  50 versicolor (0.00000000 0.50000000 0.50000000)
    6) Petal.Width< 1.75 54   5 versicolor (0.00000000 0.90740741 0.09259259) *
    7) Petal.Width>=1.75 46   1 virginica (0.00000000 0.02173913 0.97826087) *
```

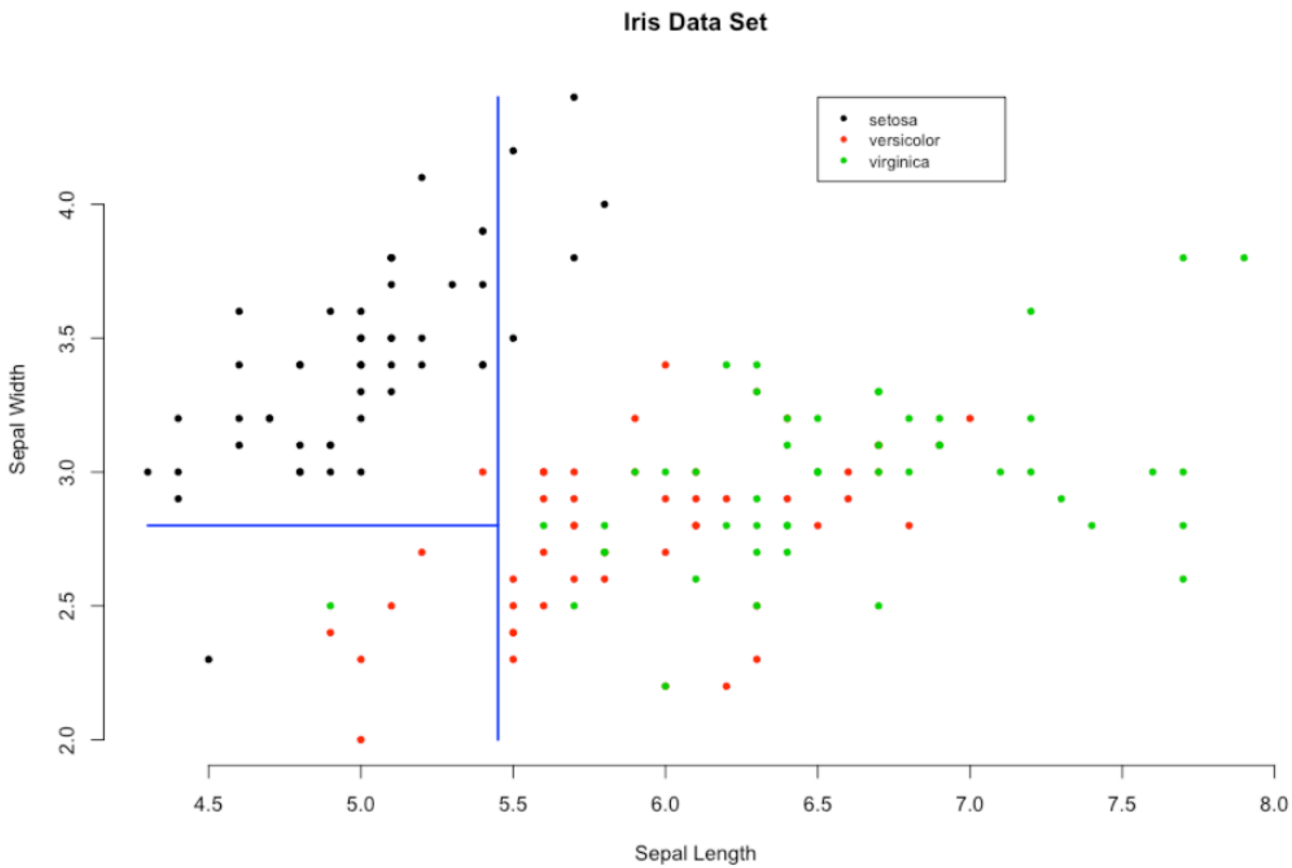
The Iris Data Set



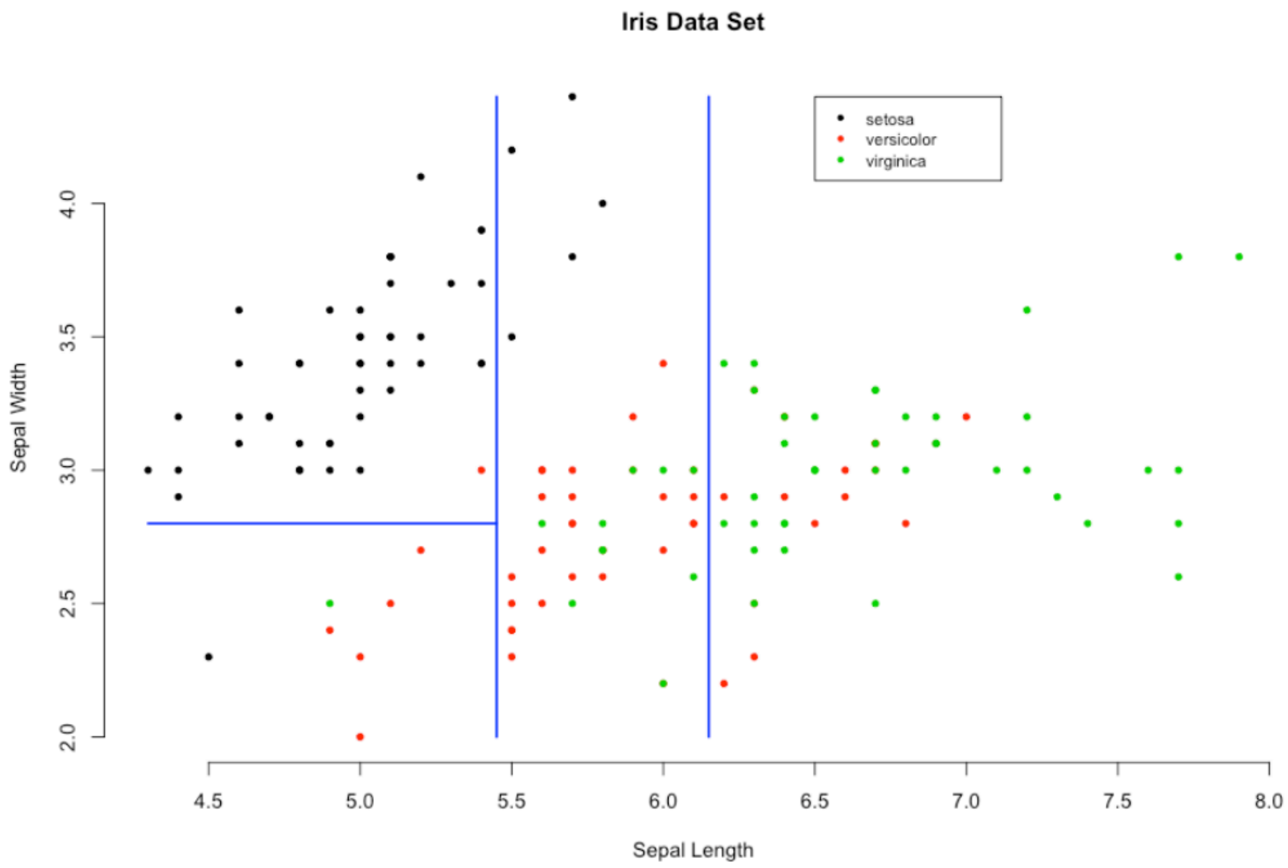
The Iris Data Set



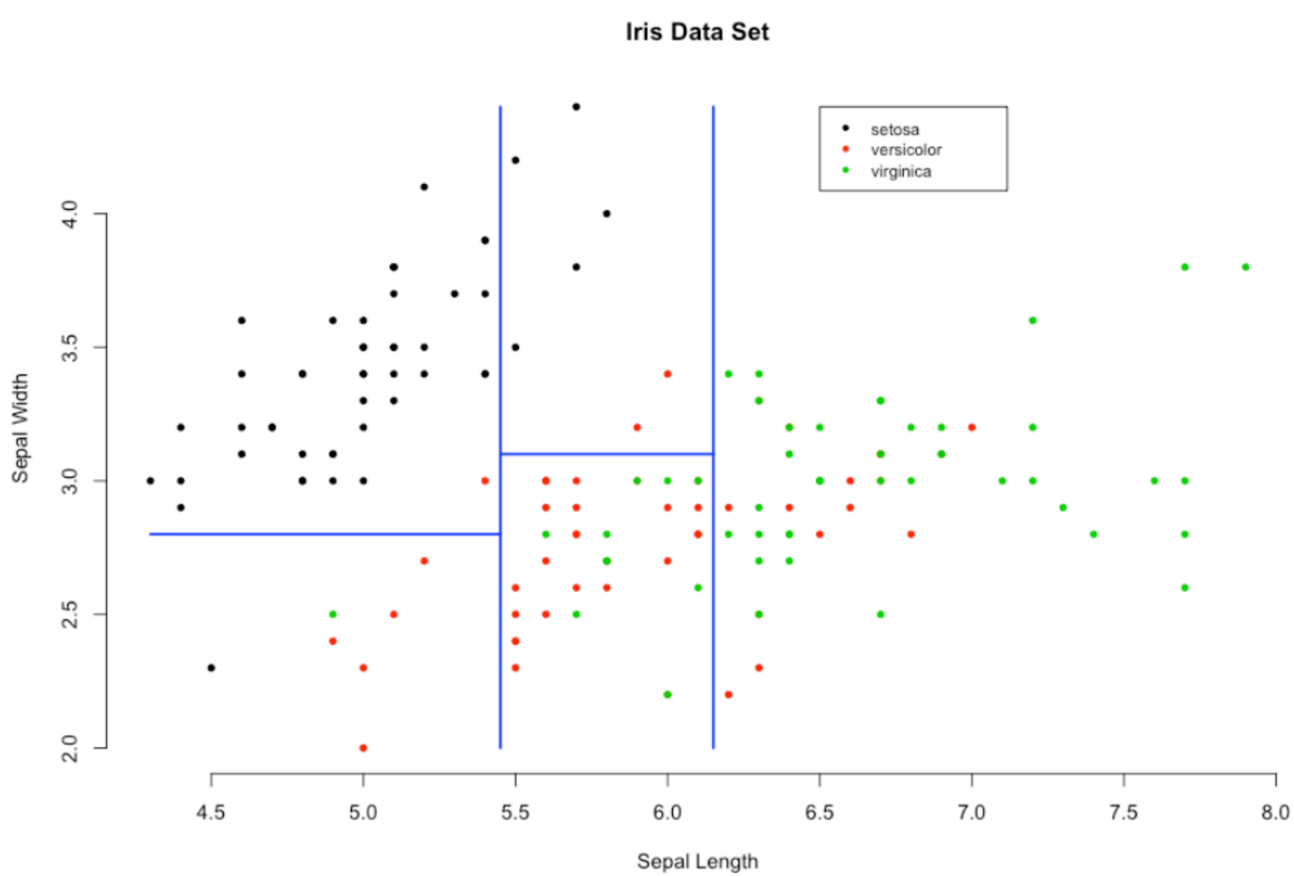
The Iris Data Set



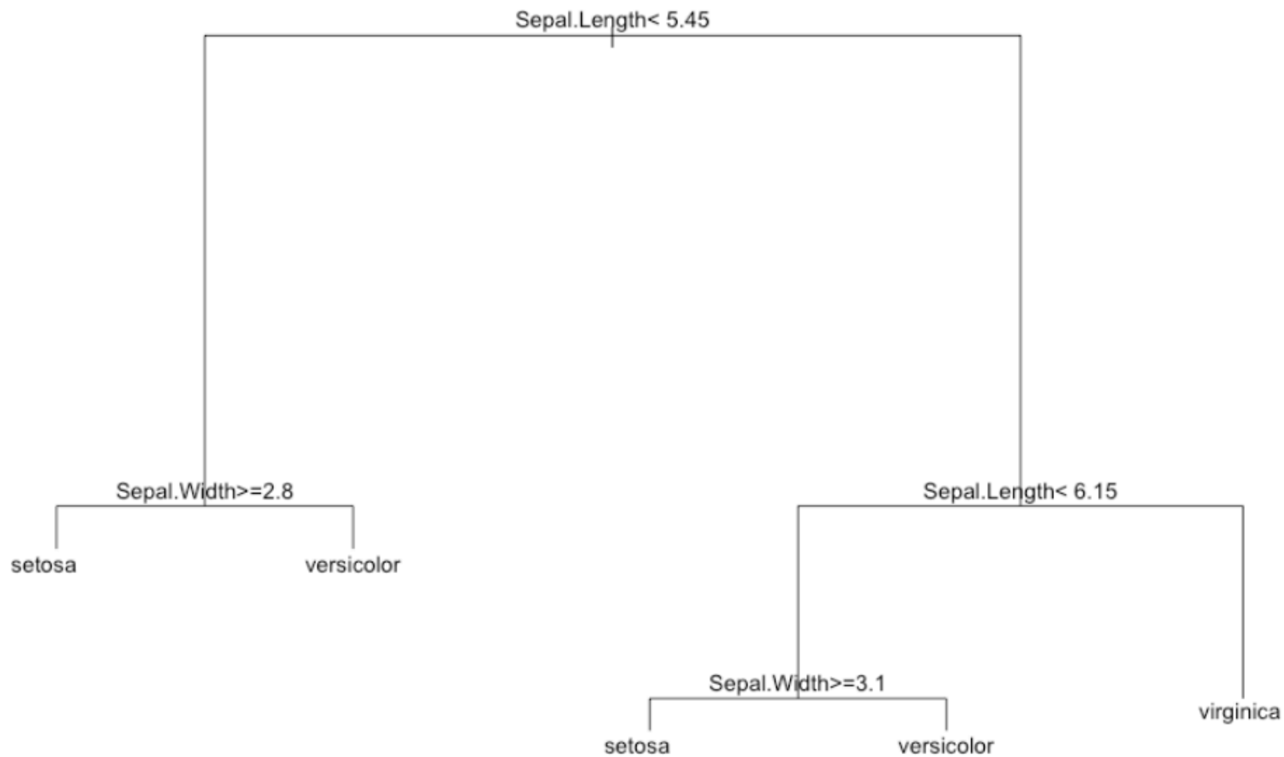
The Iris Data Set



The Iris Data Set



Decision Tree




```
n= 150

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 150 100 setosa (0.33333333 0.33333333 0.33333333)
   2) Sepal.Length< 5.45 52   7 setosa (0.86538462 0.11538462 0.01923077)
     4) Sepal.Width>=2.8 45   1 setosa (0.97777778 0.02222222 0.00000000) *
     5) Sepal.Width< 2.8 7   2 versicolor (0.14285714 0.71428571 0.14285714) *
   3) Sepal.Length>=5.45 98  49 virginica (0.05102041 0.44897959 0.50000000)
     6) Sepal.Length< 6.15 43  15 versicolor (0.11627907 0.65116279 0.23255814)
      12) Sepal.Width>=3.1 7   2 setosa (0.71428571 0.28571429 0.00000000) *
      13) Sepal.Width< 3.1 36  10 versicolor (0.00000000 0.72222222 0.27777778) *
     7) Sepal.Length>=6.15 55  16 virginica (0.00000000 0.29090909 0.70909091) *
```

Decision Tree Learning

"Method for approximating discrete-valued target functions, in which the learned function is represented by a decision tree."

Mitchell T.(1997)

Appropriate Problems for Decision Tree Learning

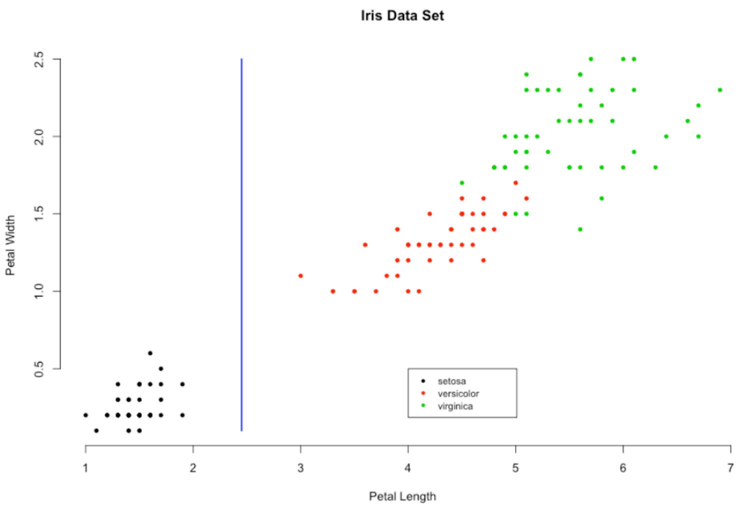
- Instances are represented by attribute-value pairs
- The target function has discrete output values
- Disjunctive descriptions may be required
- The training data may contain errors
- The training data may contain missing attribute values

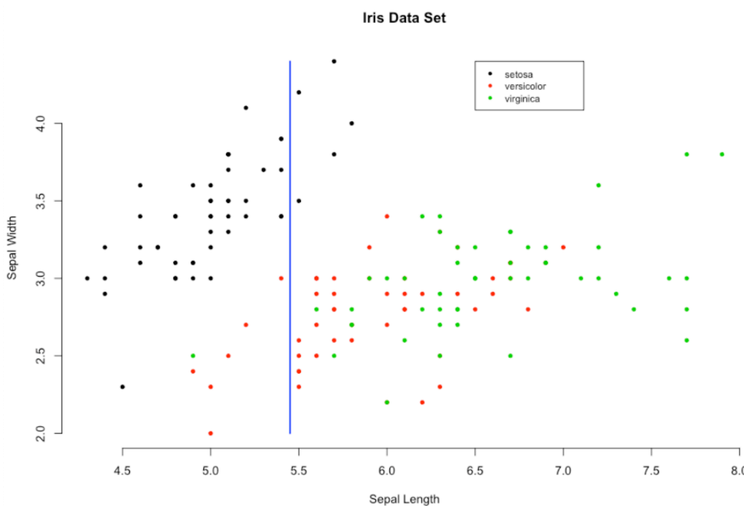
ID3 Algorithm (Quinlan 1983)

- Top-down
- At each level, answers the question: “which attribute should be tested to better discriminate the class?”
- Greedy search for an acceptable decision tree
- Never backtracks to reconsider earlier choices

Selecting the best attribute

- So, which attribute is the best classifier?
- Measures usually based on posterior distribution of classes after split.





Selecting the best attribute

Entropy measures the impurity of a collection of examples for all *n* classes

$$H(X) = -\sum_{i=1}^{n} p(x_i) \log p(x_i)$$

Information gain measures the expected reduction in entropy

$$IG(T,a) = \mathrm{H}(T) - \mathrm{H}(T|a)$$

Inductive bias of ID3

- Because of subtle interaction between the heuristic attribute selection and particular found examples, it is difficult to characterize precisely the inductive bias exhibited by ID3.

- However, we can approximately define it as **a preference for short decision trees over complex ones.**
- Trees that place high information gain attributes close to the root are preferred over those that do not.

Selecting the best attribute

- What if we use *Date* as predictor?
- There are so many possible values that we are bound to split into many small subsets, yielding high information gains
- However, this does not translate into better predictors.
- Can we do better than information gain?

C4.5 (Quinlan 1993)

Information gain ratio penalizes attributes by incorporating a term, called split information or intrinsic value, that is sensitive to how broadly and uniformly the attribute splits the data.

$$IG(T,a) = \mathrm{H}(T) - \mathrm{H}(T|a)$$

$$IGR(Ex,a) = IG/IV$$

where *IV* is the entropy of the attribute variable.

Improvements from ID3

C4.5 (and later C5.0) improved ID3 by:

- Handling heterogeneous attributes
- Handling missing values
- Handling costs
- Pruning trees after creation
- Boosting

CART (Breiman et al. 1984)

Gini impurity is a measure of how often a randomly chosen element from the set would be incorrectly labelled if it was randomly labelled according to the distribution of labels in the subset.

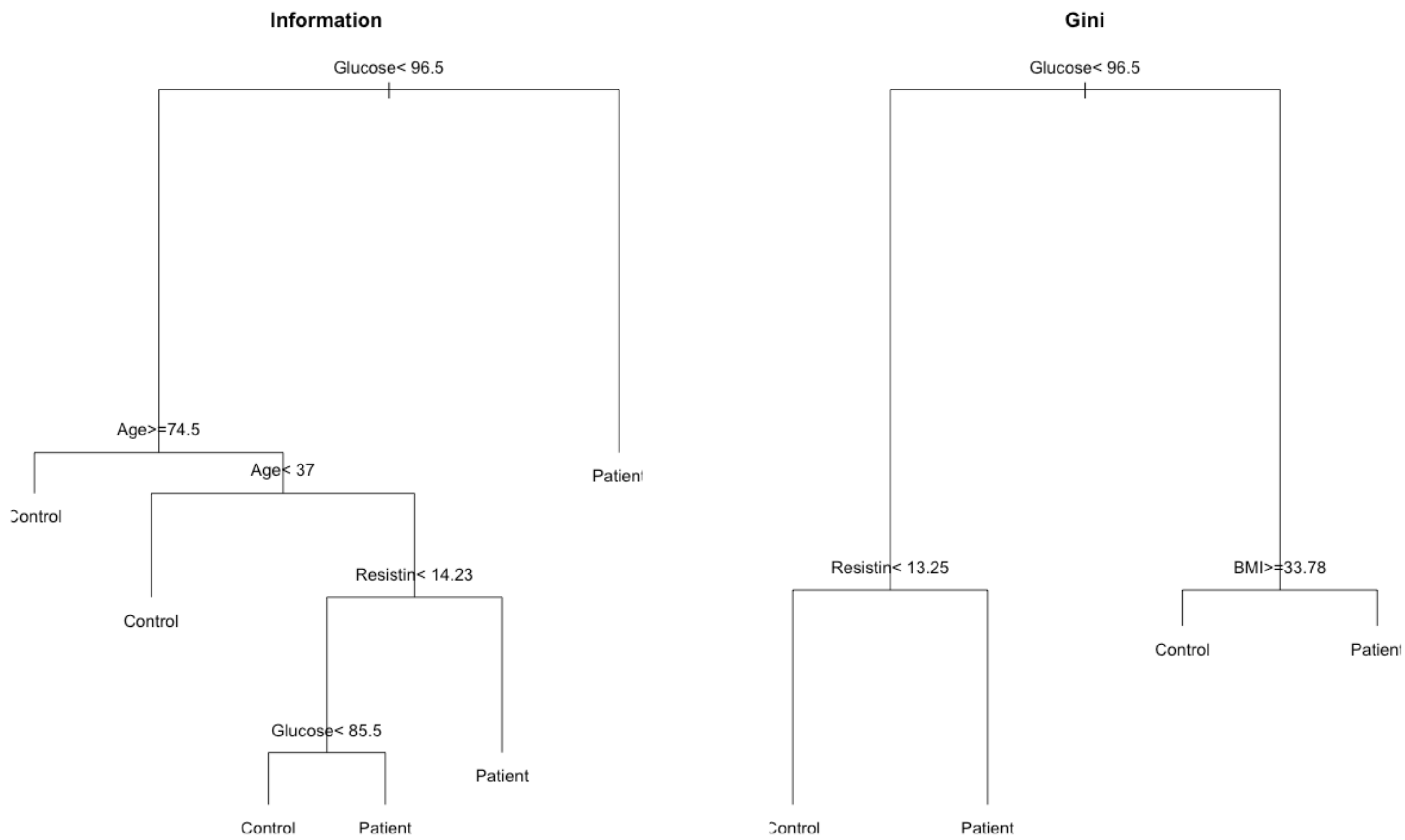
$$\mathrm{I}_G(p) = 1 - \sum_{i=1}^{J} {p_i}^2$$

Variance reduction is a broader estimate, also introduced in CART, for continuous target variables.

Using ‘rpart’ and the Breast Cancer Coimbra data set

```r
# Load package 'rpart'
library(rpart)
# Learn tree with Gini impurity
tree.gini <- rpart(Classification ~ ., data=dataset)
# Learn tree with information gain
tree.information <- rpart(Classification ~ ., data=dataset,
                          parms = list(split = "information"))

# Summary
summary(tree.information)
summary(tree.gini)

# Plot
par(mfrow=c(1,2))
plot(tree.information, main="Information")
text(tree.information)
plot(tree.gini, main="Gini")
text(tree.gini)
```

Using 'rpart' and the Breast Cancer Coimbra data set



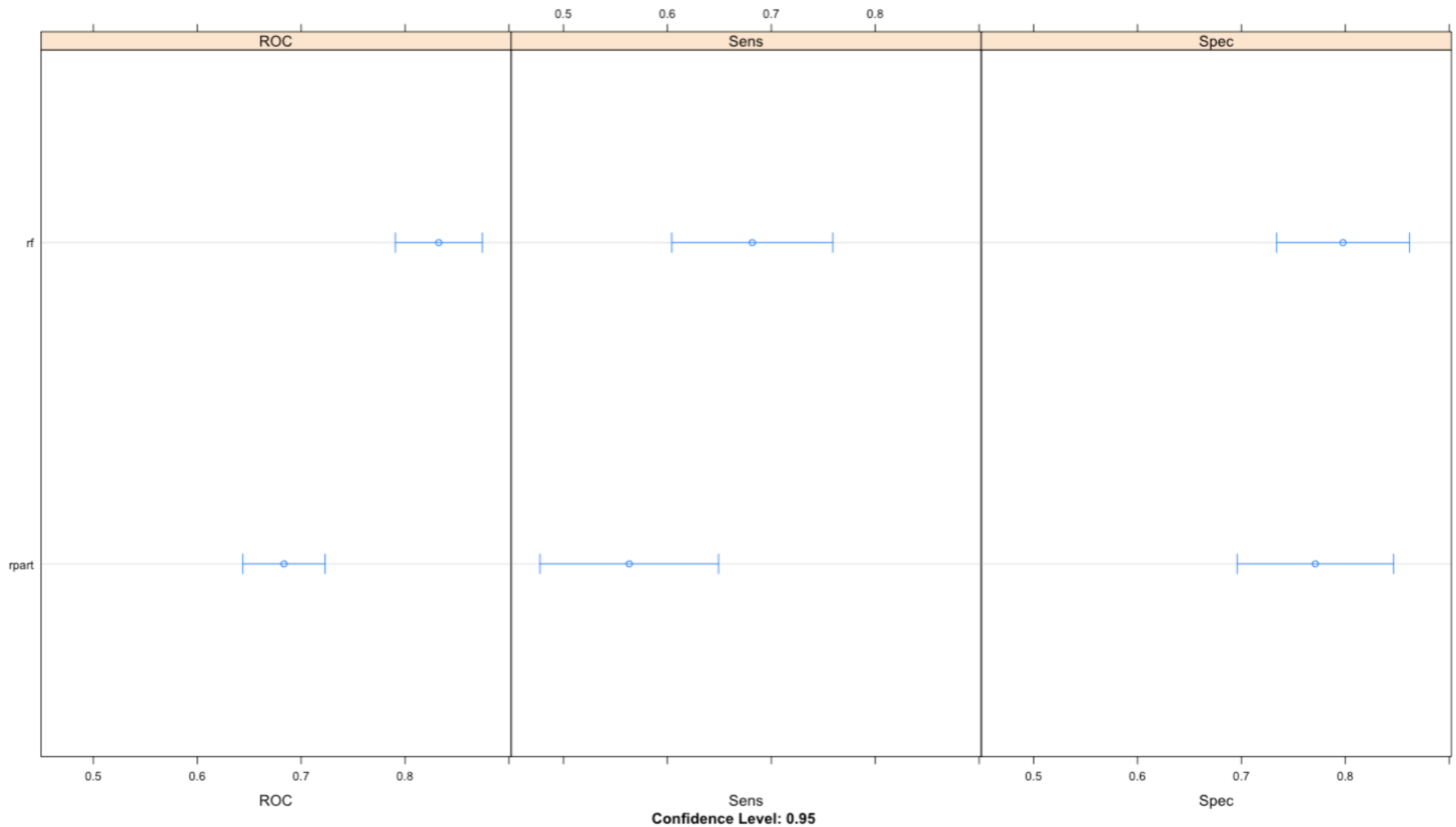
Random Forests (Breiman 2001)

- To avoid the heuristic decisions and reduce inductive bias of decision trees, **random forests** have been proposed.
- The idea is to combine the **bagging** approach (proposed by Breiman and better discussed in next classes) and **random selection of features:**
  - Multiple bootstrapped samples are taken from the original data set.
  - A decision tree with randomly selected features is learned from each sample.
  - Final classification of the random forest is usually done by **majority vote** of the ensemble.

Using ‘caret’ and the Breast Cancer Coimbra data set

```
# Run algorithms using 3 times 10-fold cross validation
metric <- "ROC"
control <- trainControl(method="repeatedcv", number=10,
                        summaryFunction=twoClassSummary,
                        classProbs=T,
                        savePredictions = T, repeats = 3)

set.seed(7)
fit.cart.rcv <- train(Classification ~ ., data=dataset, method="rpart", metric=metric, trControl=control)

set.seed(7)
fit.rf.rcv <- train(Classification ~ ., data=dataset, method="rf", metric=metric, trControl=control)

# Summarize accuracy of models
fit.models <- list(rpart=fit.cart.rcv, rf=fit.rf.rcv)
results <- resamples(fit.models)
summary(results)

# ROC curves for models
par(mfrow=c(1,2))
rocs <- lapply(fit.models, function(fit){plot.roc(fit$pred$obs,fit$pred$Patient,
                                                  main=paste("3 x 10-fold CV -",fit$method), debug=F, print.auc=T)})

# Compare accuracy of models
dotplot(results)
```

Using ‘caret’ and the Breast Cancer Coimbra data set



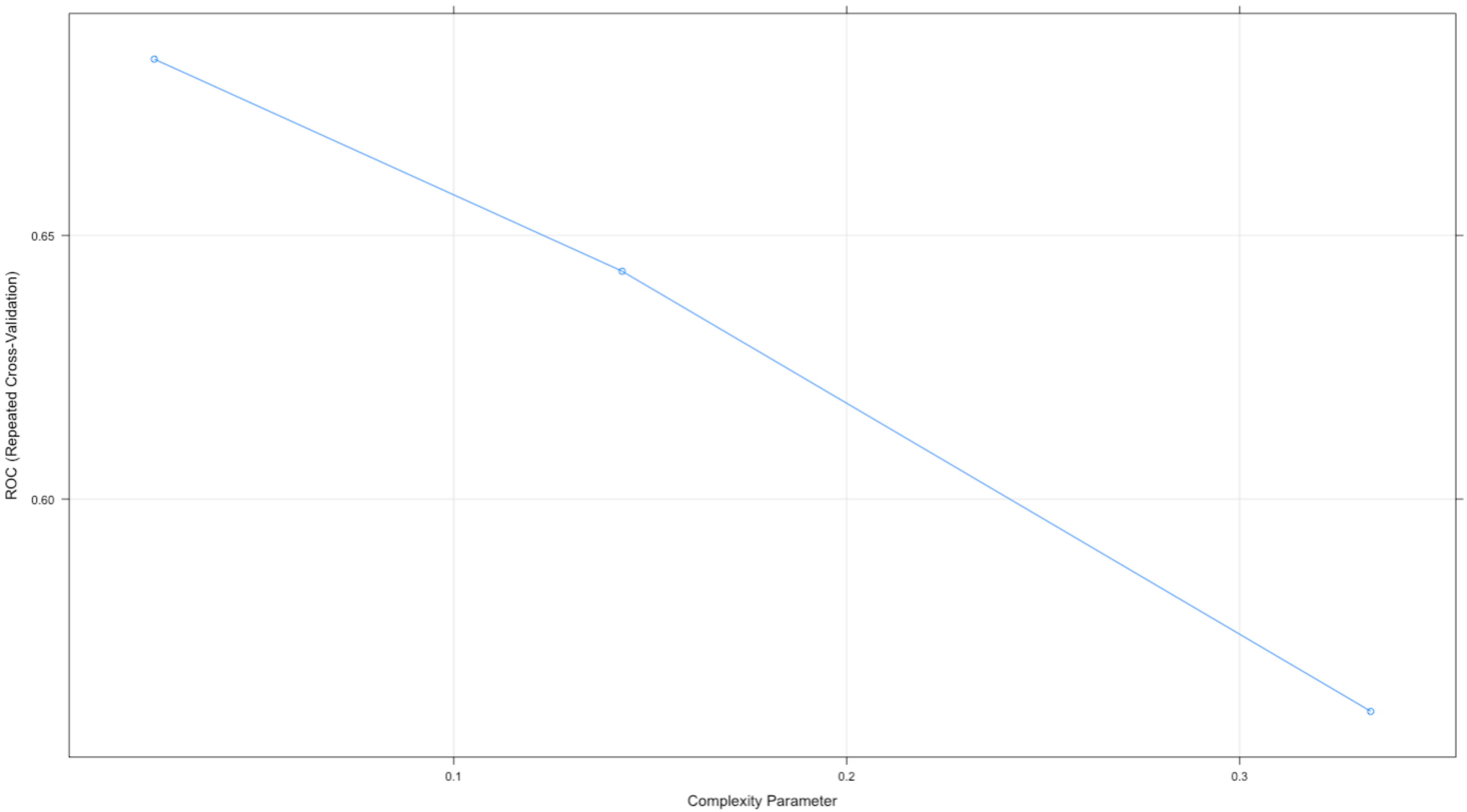
Using ‘caret’ and the Breast Cancer Coimbra data set

```r
# Inspect models
print(fit.cart.rcv)
getModelInfo(fit.cart.rcv)
getModelInfo(fit.cart.rcv)$rpart
getModelInfo(fit.cart.rcv)$rpart$parameters

# Inspect models
print(fit.rf.rcv)
getModelInfo(fit.rf.rcv)
getModelInfo(fit.rf.rcv)$rf
getModelInfo(fit.rf.rcv)$rf$parameters

# ROC complexity for models
plot(fit.cart.rcv)
plot(fit.rf.rcv)
```

Analysis for Random Forest

```
CART

94 samples
 9 predictor
 2 classes: 'Control', 'Patient'

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 85, 85, 83, 84, 84, 85, ...
Resampling results across tuning parameters:

  cp          ROC        Sens       Spec
  0.02380952  0.6834167  0.5633333  0.7711111
  0.14285714  0.6431944  0.5583333  0.6888889
  0.33333333  0.5597222  0.4516667  0.6677778

ROC was used to select the optimal model using the largest value.
The final value used for the model was cp = 0.02380952.
```

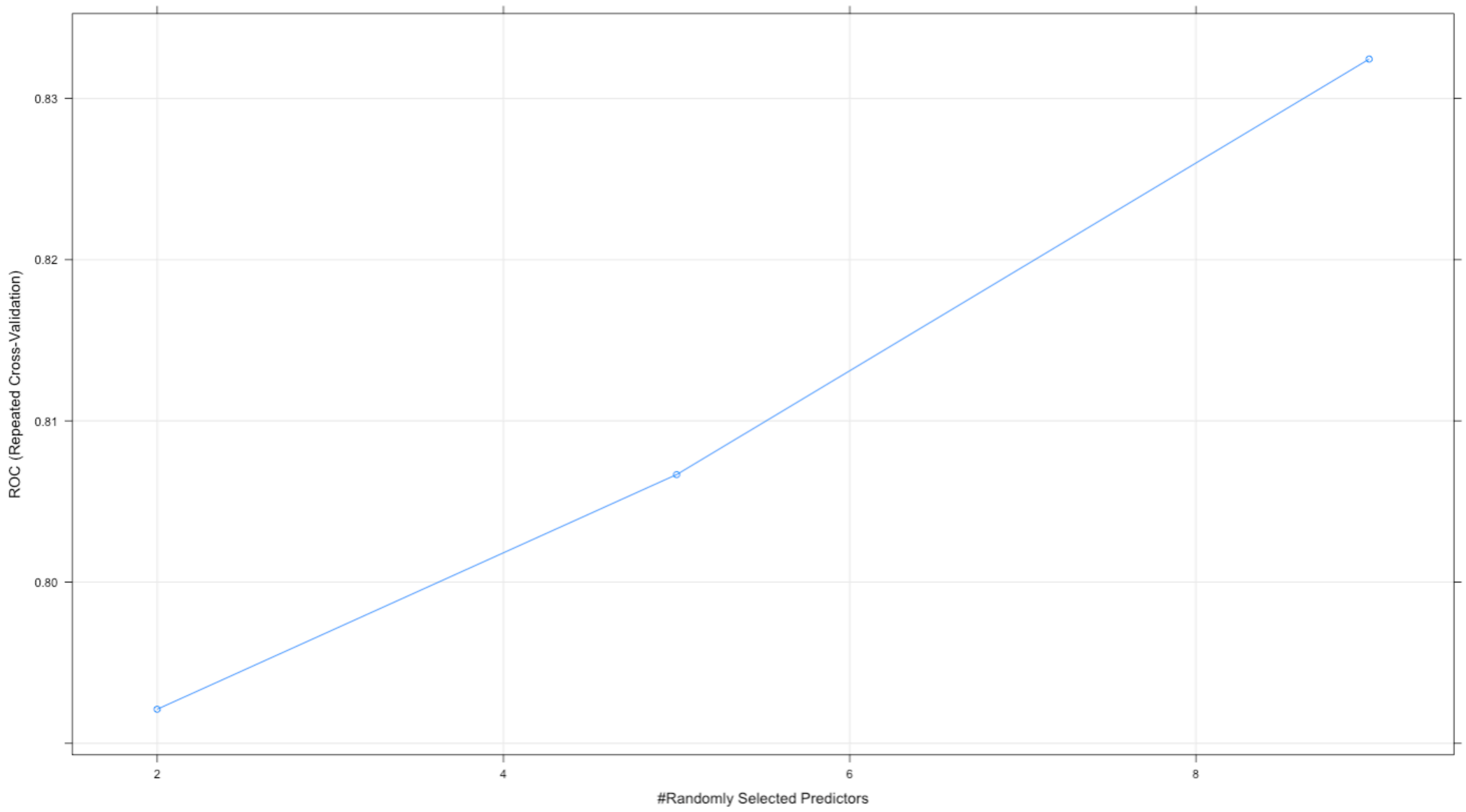
Analysis for Decision Tree



Analysis for Random Forest

```
Random Forest

94 samples
 9 predictor
 2 classes: 'Control', 'Patient'

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 85, 85, 83, 84, 84, 85, ...
Resampling results across tuning parameters:

  mtry  ROC        Sens       Spec
  2     0.7921111  0.6283333  0.7577778
  5     0.8066667  0.6566667  0.7900000
  9     0.8324444  0.6816667  0.7977778

ROC was used to select the optimal model using the largest value.
The final value used for the model was mtry = 9.
```

Analysis for Random Forest



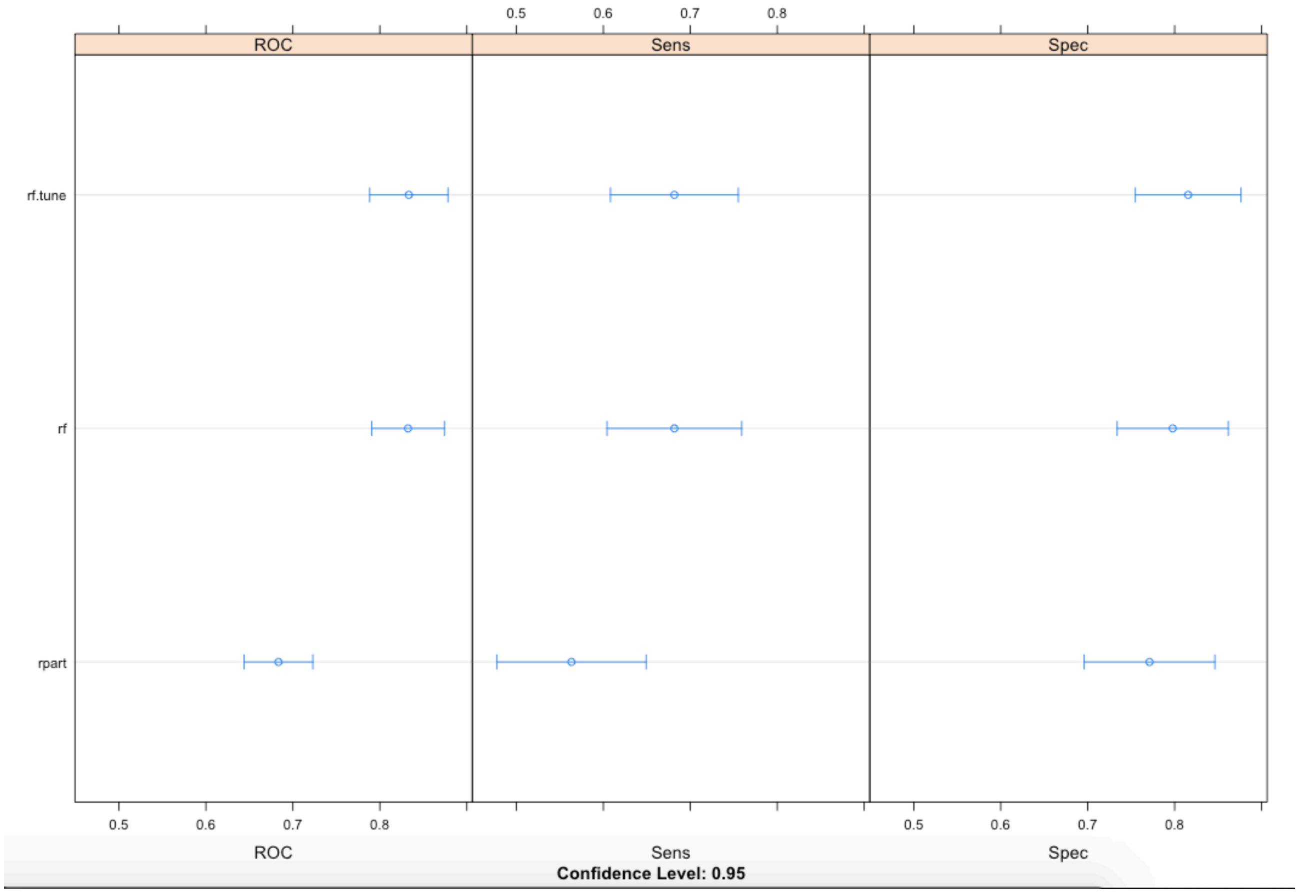
Using 'caret' and the Breast Cancer Coimbra data set

```
# Improve Random Forest
myGrid <- expand.grid(mtry = 1:9)

set.seed(7)
fit.rf.rcv.tune <- train(Classification ~ ., data=dataset, method="rf", metric=metric, trControl=control,
                    tuneGrid = myGrid)

# Summarize accuracy of models
fit.models <- list(rpart=fit.cart.rcv, rf=fit.rf.rcv, rf.tune=fit.rf.rcv.tune)
results <- resamples(fit.models)
summary(results)

# ROC curves for models
par(mfrow=c(1,3))
rocs <- lapply(fit.models, function(fit){plot.roc(fit$pred$obs,fit$pred$Patient,
                                           main=paste("3 x 10-fold CV -",fit$method), debug=F, print.auc=T)})
# Compare accuracy of models
dotplot(results)
```

Using 'caret' and the Breast Cancer Coimbra data set

```
Call:
summary.resamples(object = results)

Models: rpart, rf, rf.tune
Number of resamples: 30

ROC
         Min.  1st Qu.  Median      Mean   3rd Qu. Max. NA's
rpart    0.45 0.646875 0.68125 0.6834167 0.7437500  0.9    0
rf       0.60 0.750000 0.85000 0.8324444 0.9191667  1.0    0
rf.tune  0.60 0.752500 0.85000 0.8333333 0.9300000  1.0    0

Sens
         Min. 1st Qu. Median      Mean 3rd Qu. Max. NA's
rpart    0.00     0.5   0.50 0.5633333  0.7500    1    0
rf       0.25     0.5   0.75 0.6816667  0.7875    1    0
rf.tune  0.25     0.5   0.75 0.6816667  0.7875    1    0

Spec
         Min.   1st Qu. Median      Mean   3rd Qu. Max. NA's
rpart     0.2 0.6166667    0.8 0.7711111 0.9583333    1    0
rf        0.4 0.6666667    0.8 0.7977778 1.0000000    1    0
rf.tune   0.4 0.7000000    0.8 0.8155556 1.0000000    1    0
```

Using ‘caret’ and the Breast Cancer Coimbra data set



Using 'caret' and the Breast Cancer Coimbra data set

```
Random Forest

94 samples
 9 predictor
 2 classes: 'Control', 'Patient'

No pre-processing
Resampling: Cross-Validated (10 fold, repeated 3 times)
Summary of sample sizes: 85, 85, 83, 84, 84, 85, ...
Resampling results across tuning parameters:

  mtry  ROC        Sens       Spec
  1     0.7866111  0.6266667  0.7655556
  2     0.7939444  0.6416667  0.7655556
  3     0.8070556  0.6483333  0.7833333
  4     0.8120556  0.6816667  0.8100000
  5     0.8108889  0.6583333  0.7977778
  6     0.8253889  0.6566667  0.7966667
  7     0.8298333  0.6733333  0.8088889
  8     0.8333333  0.6816667  0.8155556
  9     0.8325556  0.6816667  0.7966667

ROC was used to select the optimal model using the largest value.
The final value used for the model was mtry = 8.
```

Exercise

Compare a decision tree with a random forest in the Cervical Cancer (Risk Factors) data set (available from UCI repository), trying to accurately classify Dx.Cancer