

## Classification and Regression Trees



Revised: 12/10/2018



Summary .....	1
Data Input.....	4
Analysis Options.....	6
Tables and Graphs.....	7
Analysis Summary .....	8
R Tree Diagram.....	9
Tree Diagram .....	10
R Tree Structure.....	11
Node Probabilities.....	12
Predictions and Residuals .....	12
2-D Scatterplot .....	14
3-D Scatterplot.....	16
Classification Table .....	19
Observed versus Predicted.....	20
Deviance Plot.....	21
Save Results .....	24
Example 2 .....	25
References.....	27

## Summary

The *Classification and Regression Trees* procedure implements a machine-learning process to predict observations from data. It creates models of 2 forms:

1. *Classification models* that divide observations into groups based on their observed characteristics.

2. *Regression models* that predict the value of a dependent variable.

The models are constructed by creating a tree, each node of which corresponds to a binary decision. Given a particular observation, one travels down the branches of the tree until a terminating leaf is found. Each leaf of the tree is associated with a predicted class or value.

Observations are typically divided into three sets:

1. A *training* set which is used to construct the tree.
2. A *validation* set for which the actual classification or value is known, which can be used to validate the model.
3. A *prediction* set for which the actual classification or value is not known but for which predictions are desired.

The dependent variable may be either categorical or quantitative, as may the predictor variables.

The calculations are performed by the “tree” package in R. To run the procedure, R must be installed on your computer together with the *tree* package. For information on downloading and installing R, refer to the document titled “R – Installation and Configuration”.

**Sample StatFolios:** *tree1.sgp* and *tree2.sgp*

## Sample Data

The first example uses the classic set of data from Fisher (1936), contained in file *iris.sgd*. The data consist of a total of  $n = 150$  irises, 50 from each of  $g = 3$  different species: *setosa*, *versicolor*, and *virginica*. Measurements were made on  $p = 4$  variables, describing the length and width of the sepal and petal. The table below shows a partial list of the data in that file:

<i>Sample</i>	<i>Sepal length</i>	<i>Sepal width</i>	<i>Petal length</i>	<i>Petal width</i>	<i>Species</i>
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa
...	...	...	...	...	...

A classification model is desired that uses the 4 quantitative variables to determine the probable species of each iris.

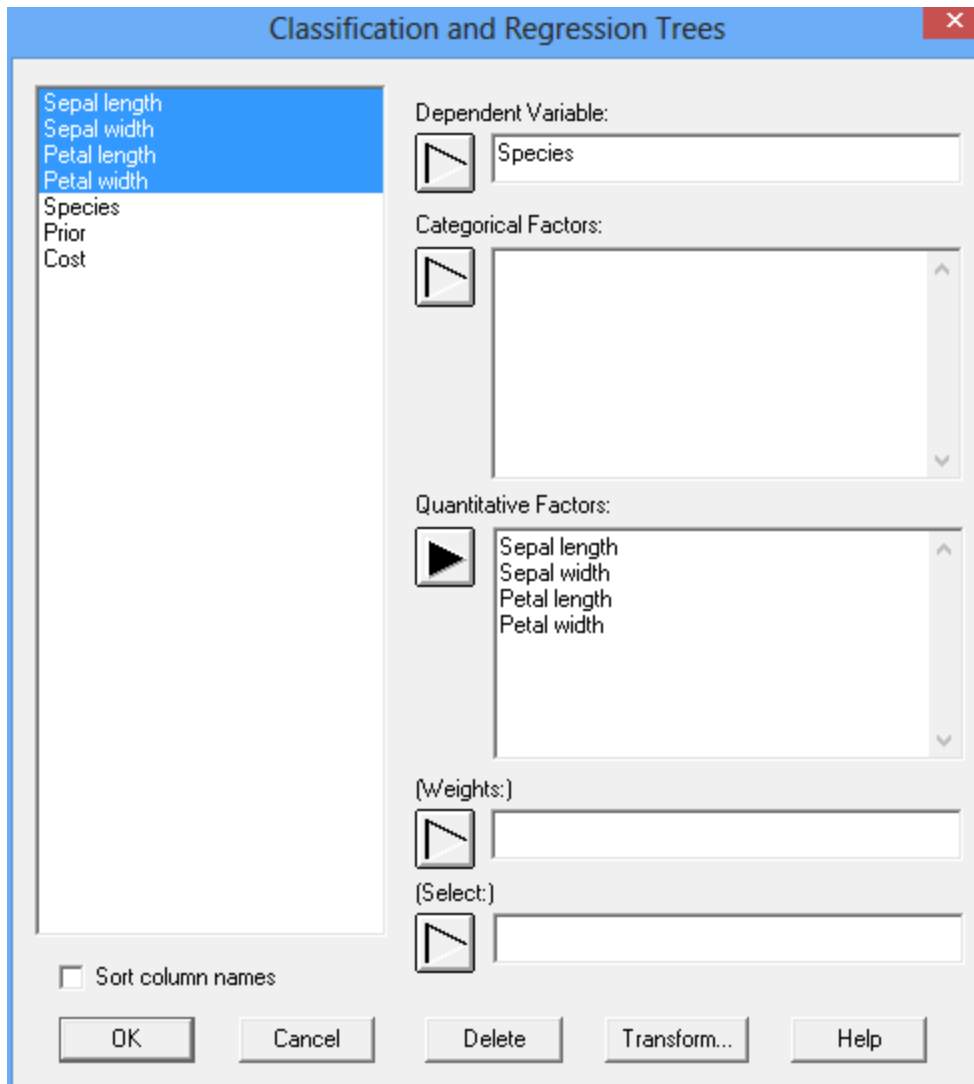
A second example uses the file *93cars.sgd* which contains information on 26 variables for  $n = 93$  models of automobiles, taken from Lock (1993). The table below shows a partial list of 8 columns from that file:

<i>Make</i>	<i>Model</i>	<i>Type</i>	<i>MPG Highway</i>	<i>Weight</i>	<i>Horsepower</i>	<i>Wheelbase</i>	<i>Drive Train</i>
Acura	Integra	Small	31	2705	140	102	front
Acura	Legend	Midsize	25	3560	200	115	front
Audi	90	Compact	26	3375	172	102	front
Audi	100	Midsize	26	3405	172	106	front
BMW	535i	Midsize	30	3640	208	109	rear
Buick	Century	Midsize	31	2880	110	105	front
Buick	LeSabre	Large	28	3470	170	111	front
...	...		...	...	...	...	...

## Data Input

### Example 1

When the *Classification and Regression Trees* procedure is selected from the Statgraphics menu, a data input dialog box is displayed. In the first example, 4 quantitative factors are used to construct a classification model for *species*:

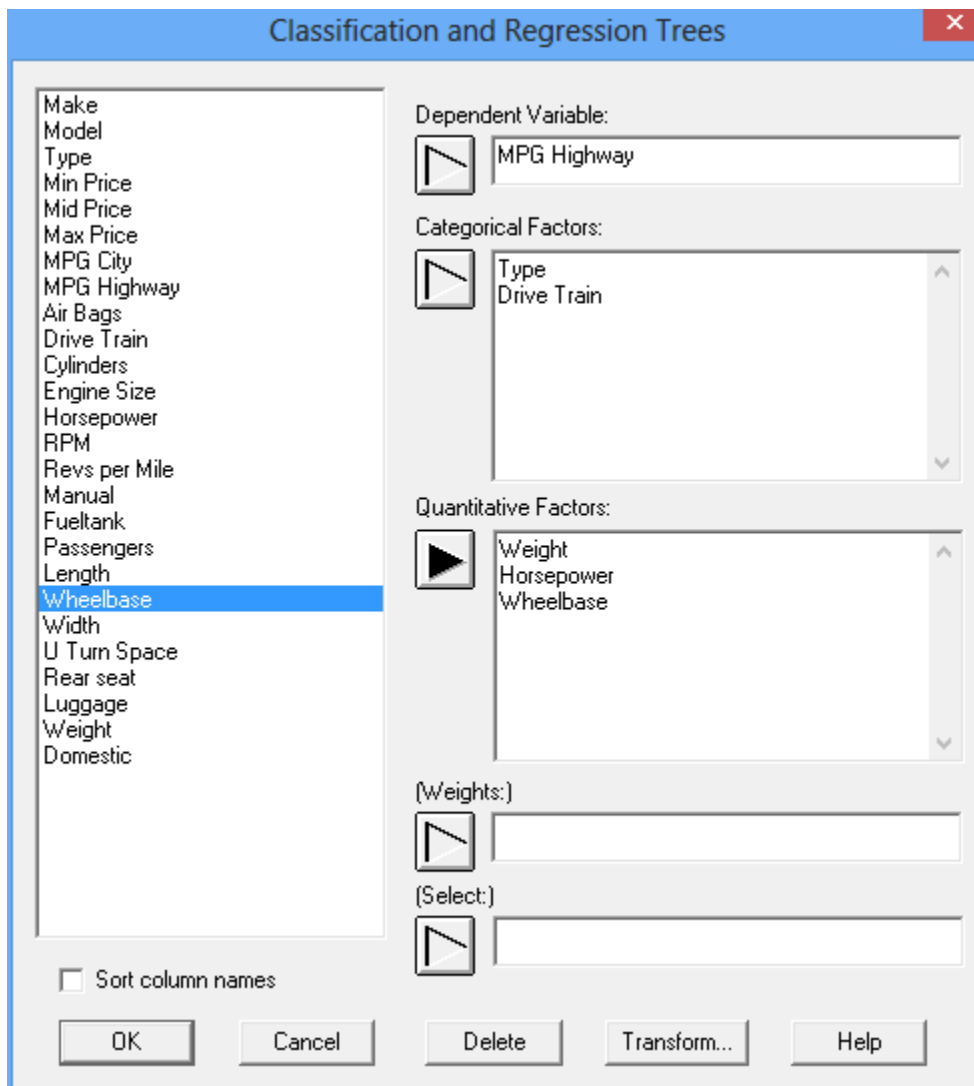


- **Dependent variable:** name of the column containing the class or value of the variable to be predicted. If fitting a classification model, this variable may be either categorical or quantitative. If fitting a regression model, this variable must be quantitative.
- **Categorical factors:** names of the columns containing the categorical variables (if any) that will be used to predict the dependent variable.
- **Quantitative factors:** names of the columns containing the continuous quantitative variables (if any) that will be used to predict the dependent variable.

- **Weights:** optional numeric column containing weights to be applied to each case when fitting the model. All weights must be positive.
- **Select:** optional Boolean column or expression identifying the cases (rows of the Databook) to be included in the analysis.

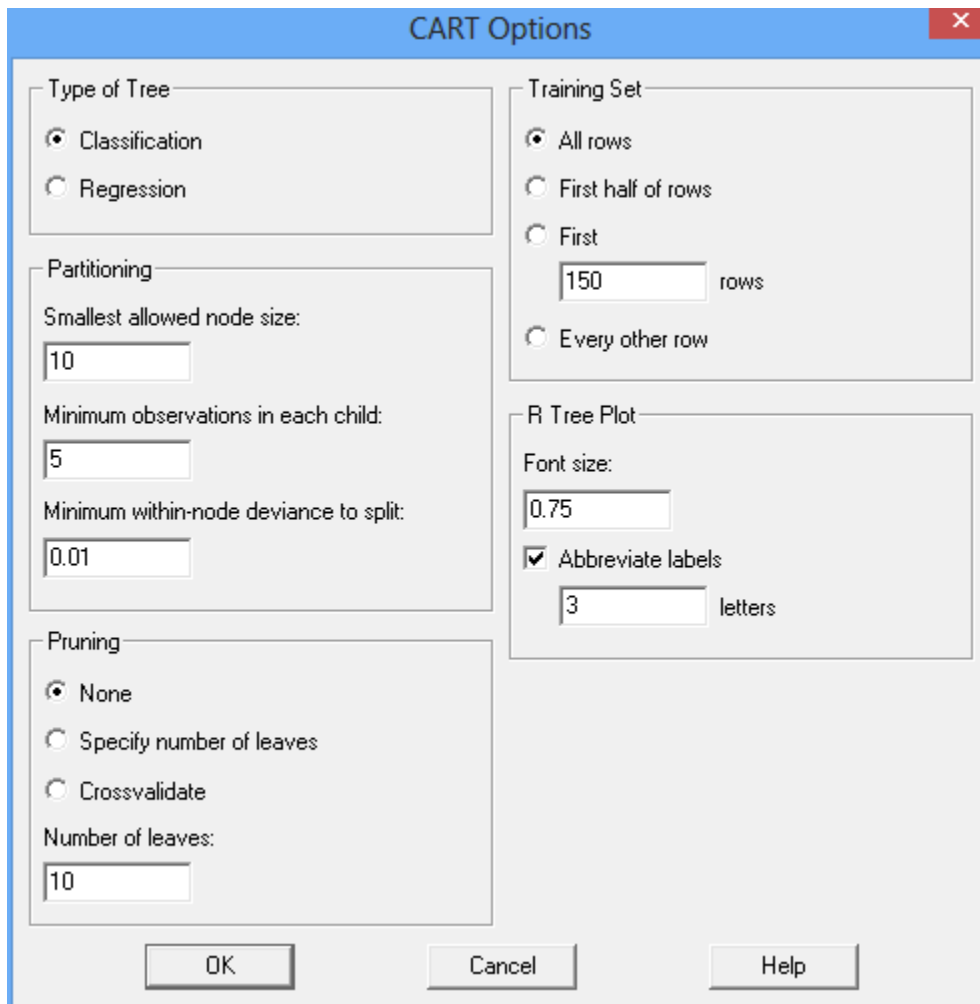
### Example 2

In the second example, 2 categorical factors and 3 quantitative factors are used to construct a regression model for *MPG Highway*:



## Analysis Options

The *Analysis Options* dialog box sets various options for fitting the model:

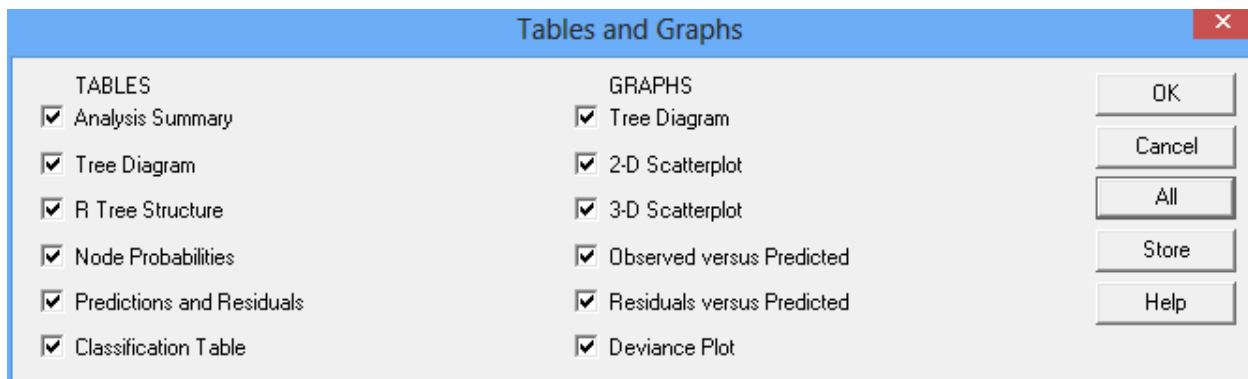


- **Type of Tree:** Classification trees are used to predict the value of categorical variables. Regression trees are used to predict the value of continuous quantitative variables.
- **Partitioning:** controls the initial partitioning of the tree into branches and leaves.
  - **Smallest allowed node size:** the minimum number of observations in a node for it to be split into 2 smaller nodes.
  - **Minimum number of observations in each child:** the minimum number of observations allowed in each child node after the split.
  - **Minimum within-node deviance to split:** the smallest deviance allowed for a node to be split. Smallest values cause more nodes to be created.
- **Pruning:** reduces the complexity of the tree by pruning branches.
  - **None:** do no pruning.

- **Specify number of leaves:** prune the tree so that it has the specified number of leaves. The tree returned is a pruning of the initial tree that has the smallest error for the size specified.
- **Crossvalidate:** uses crossvalidation to select the best pruning of the size specified.
- **Training Set:** observations to be included in the training set used to fit the tree.
- **R Tree Plot:** controls the labels on the tree generated by R.
  - **Font size:** scaling factor for the font size.
  - **Abbreviate levels:** if selected, displays levels of categorical factors using no more than the number of letters specified.

## Tables and Graphs

The following tables and graphs may be created:



## Analysis Summary

The *Analysis Summary* begins with a list of the R commands that were executed.

### Classification and Regression Trees

```
d<-
read.csv("C:\\\\Users\\\\Neil\\\\AppData\\\\Local\\\\Temp\\\\data.csv",dec=".",
",sep=",",stringsAsFactors=TRUE)
setwd("c:\\temp")
library("tree")

## Warning: package 'tree' was built under R version 3.2.5

treefit=tree(Species~Sepal.length+Sepal.width+Petal.length+Petal.width,control=tree.control(nobs=150,mincut=5,minsize=10,mindev=0.01),data=d)
summary(treefit)

##
## Classification tree:
## tree(formula = Species ~ Sepal.length + Sepal.width + Petal.length +
##       Petal.width, data = d, control = tree.control(nobs = 150,
##       mincut = 5, minsize = 10, mindev = 0.01))
## Variables actually used in tree construction:
## [1] "Petal.length" "Petal.width" "Sepal.length"
## Number of terminal nodes: 6
## Residual mean deviance: 0.1253 = 18.05 / 144
## Misclassification error rate: 0.02667 = 4 / 150

plot(treefit)
text(treefit,pretty=3,cex=0.75)

p<-prune.tree(treefit)
write.table(treefit$frame,file="C:\\Users\\Neil\\AppData\\Local\\Temp\\frame.csv",sep=",")
write.table(treefit$where,file="C:\\Users\\Neil\\AppData\\Local\\Temp\\where.csv",sep=",",row.names=FALSE)
write.table(cbind(p$size,p$dev,p$k),file="C:\\Users\\Neil\\AppData\\Local\\Temp\\prune.csv",sep=",",row.names=FALSE)
```

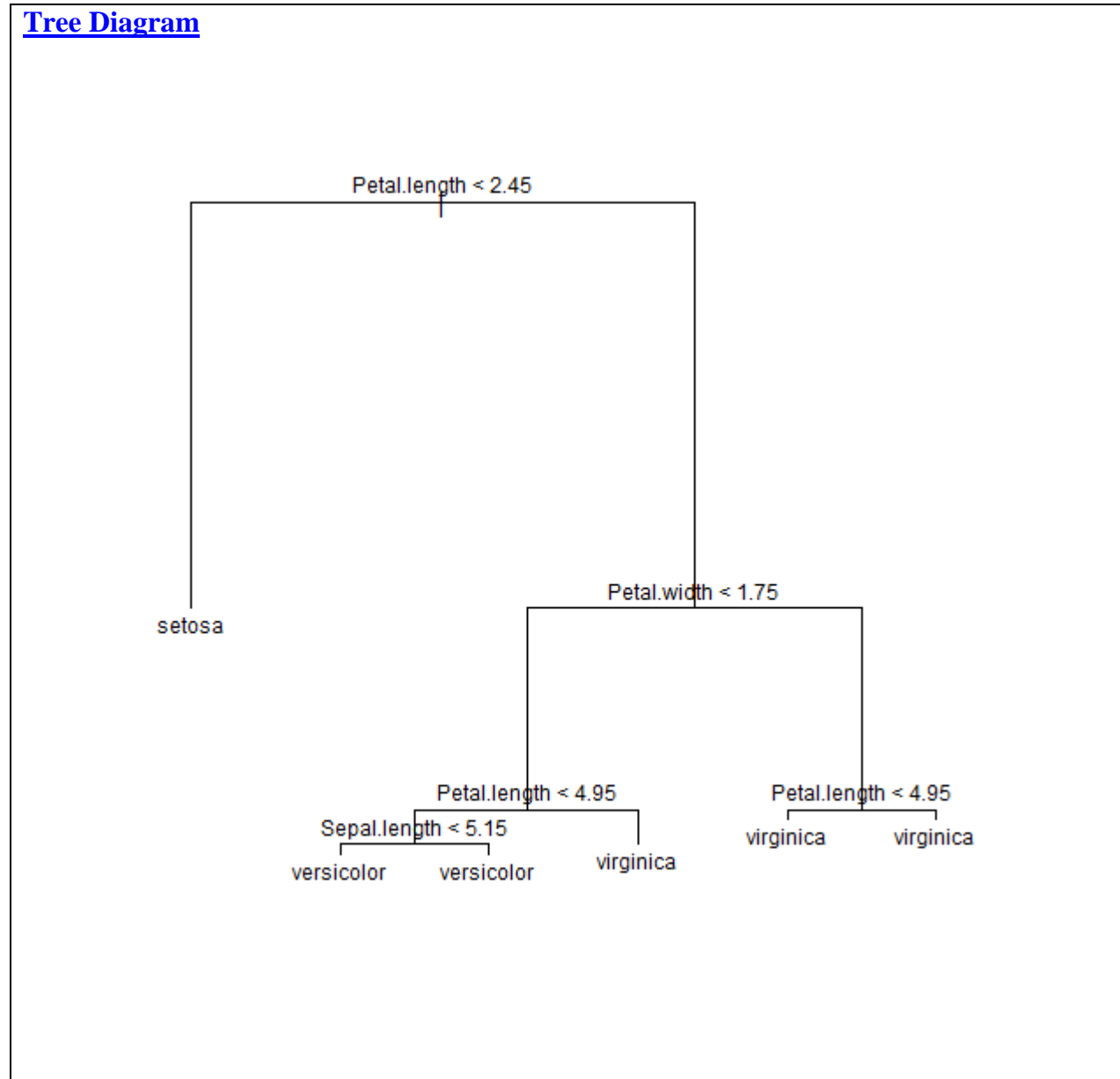
Of particular interest are:

1. **Variables actually used in tree construction:** shows which of the predictor variables were actually used to construct the tree. In this example, 3 of the 4 predictor variables were used.
2. **Number of terminal nodes:** the number of nodes after which no more splits are made (the leaves). This tree has 6 leaves.
3. **Residual mean deviance:** a measure of the error remaining in the tree after construction. For a regression tree, this is related to the mean squared error.
4. **Misclassification rate:** the proportion of observations in the training set that were predicted to fall in another class than they actually did. In this case, 4 of the 150 irises were not predicted correctly.



## R Tree Diagram

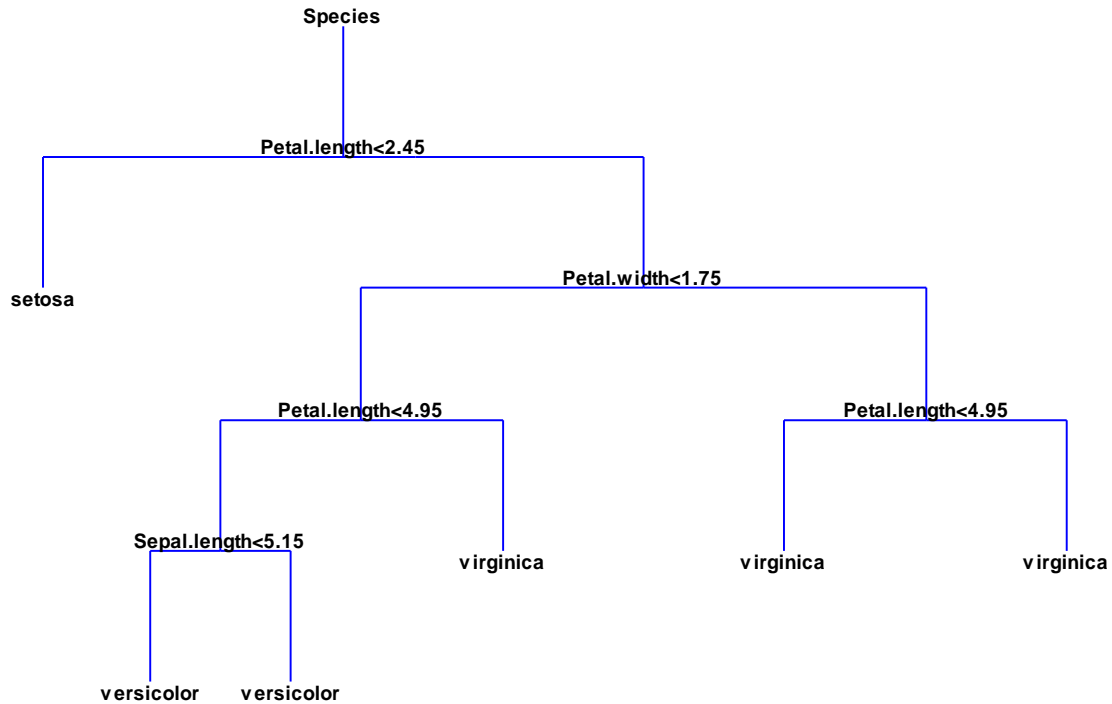
This pane shows the tree diagram generated by R:



To classify an observation, begin at the top of the tree. At each node, move left if the binary statement is true or move right if it is false. You will eventually reach a terminating node (leaf) at which the predicted value of the dependent variable is displayed. The size of the text is controlled by the *Analysis Options* dialog box.

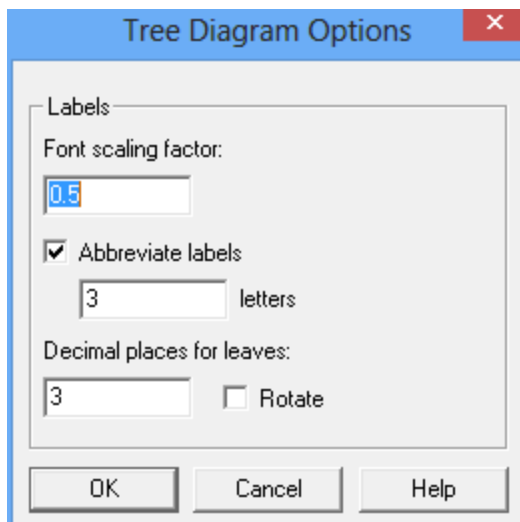
## Tree Diagram

This pane shows the tree diagram generated by Statgraphics:



If the tree is hard to read, use your mouse wheel or toolbar controls to zoom in on a section of it.

### Pane Options



- **Font scaling factor:** multiplies the default font size.
- **Abbreviate levels:** if selected, displays levels of categorical factors using no more than the number of letters specified.

- **Decimal places for leaves:** controls the format of the values displayed at the terminating nodes. Select “Rotate” to orient the labels vertically.

## R Tree Structure

This table summarizes the structure of the tree:

R Tree Structure						
Node	Label	Size	Deviance	Species	Splits left	Splits right
1	Petal.length	150	329.584	setosa	<2.45	>2.45
2	<leaf>	50	0.0	setosa		
3	Petal.width	100	138.629	versicolor	<1.75	>1.75
4	Petal.length	54	33.3175	versicolor	<4.95	>4.95
5	Sepal.length	48	9.72142	versicolor	<5.15	>5.15
6	<leaf>	5	5.00402	versicolor		
7	<leaf>	43	0.0	versicolor		
8	<leaf>	6	7.63817	virginica		
9	Petal.length	46	9.63538	virginica	<4.95	>4.95
10	<leaf>	6	5.40674	virginica		
11	<leaf>	40	0.0	virginica		

The table includes:

- **Node:** a number assigned to each node by R.
- **Label:** if not a terminating node, the variable involved in the decision to split the node.
- **Size:** the number of observations in the training set that reach the specified node. If the data are weights, the value shown is the sum of the weights.
- **Deviance:** a measure of the variability amongst all observations in the training set that reach the specified node.
- **Species:** for a classification tree, the class with the highest probability of arriving at that node (may be tied).
- **Splits left:** at a non-terminating node, the decision criterion that must be true to move along the branch to the left.
- **Splits right:** at a non-terminating node, the decision criterion that must be true to move along the branch to the right.

For example, at node #1, observations travel along the branch to the left if *Petal length* < 2.45 and along the branch to the right if *Petal length* > 2.45.

## Node Probabilities

This table shows the probability distribution of observations in the training set that reach each node:

Node Probabilities				
Node	Label	<i>yprob.setosa</i>	<i>yprob.versicolor</i>	<i>yprob.virginica</i>
1	Petal.length	0.333333	0.333333	0.333333
2	<leaf>	1.0	0.0	0.0
3	Petal.width	0.0	0.5	0.5
4	Petal.length	0.0	0.907407	0.0925926
5	Sepal.length	0.0	0.979167	0.0208333
6	<leaf>	0.0	0.8	0.2
7	<leaf>	0.0	1.0	0.0
8	<leaf>	0.0	0.333333	0.666667
9	Petal.length	0.0	0.0217391	0.978261
10	<leaf>	0.0	0.166667	0.833333
11	<leaf>	0.0	0.0	1.0

For example, at terminating node #8 the probability that the observation comes from the species *versicolor* equals 0.333 while the probability that it comes from the species *virginica* equals 0.667.

## Predictions and Residuals

This table shows the predicted values for each observation:

Predictions and Residuals			
Training set n=150			
Row	Leaf	Predicted	Observed
1	2	setosa	setosa
2	2	setosa	setosa
3	2	setosa	setosa
4	2	setosa	setosa
5	2	setosa	setosa
6	2	setosa	setosa
7	2	setosa	setosa
8	2	setosa	setosa
9	2	setosa	setosa
10	2	setosa	setosa
...	...	...	...

Residual mean deviance = 0.12534

- **Leaf:** the terminating node for the specified observation (displayed only for those observations contained in the training set used to build the tree).
- **Predicted:** the predicted value for the observation.
- **Observed:** the observed value of the observation.

The residual mean deviance is displayed at the bottom of the table. For a regression tree, the residual mean deviance is calculated by

$$RMD = \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{n-k} \quad (1)$$

where  $Y_i$  is the observed value of the dependent variable,  $\hat{Y}_i$  is the predicted value,  $n$  is the number of observations in the training set, and  $k$  equals the number of terminating nodes (leaves) in the fitted tree. In such a case, RMD is equivalent to the mean squared error. For a classification tree, the residual mean deviance is calculated by

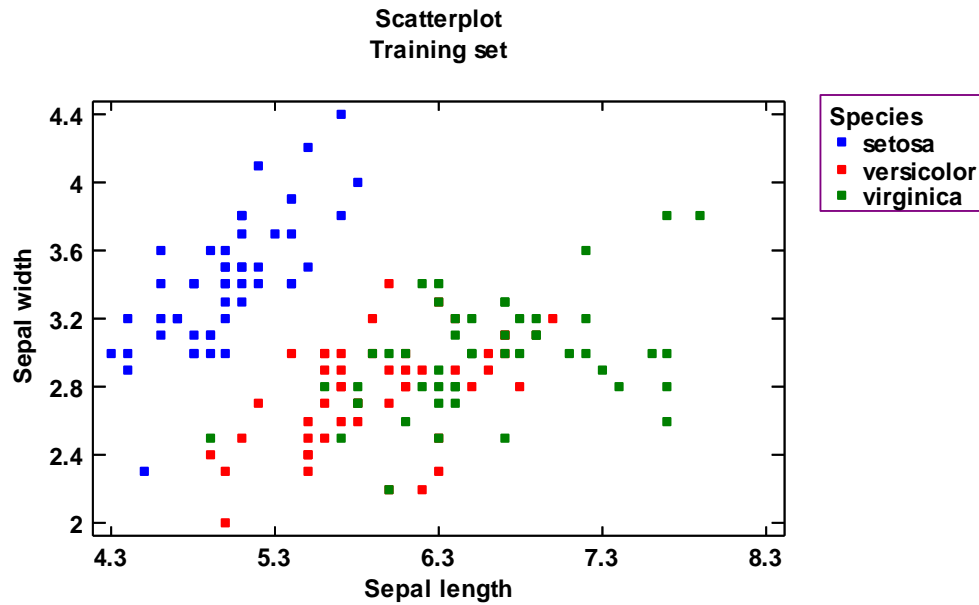
$$RMD = \frac{\sum_{i=1}^n -2\log(p_{i,j})}{n-k} \quad (2)$$

where  $p_{i,j}$  is the estimated probability that observation  $i$  would be assigned to class  $j$ , where  $j$  is the index of the class predicted by the model. In both cases, smaller values of RMD correspond to better predicting trees.

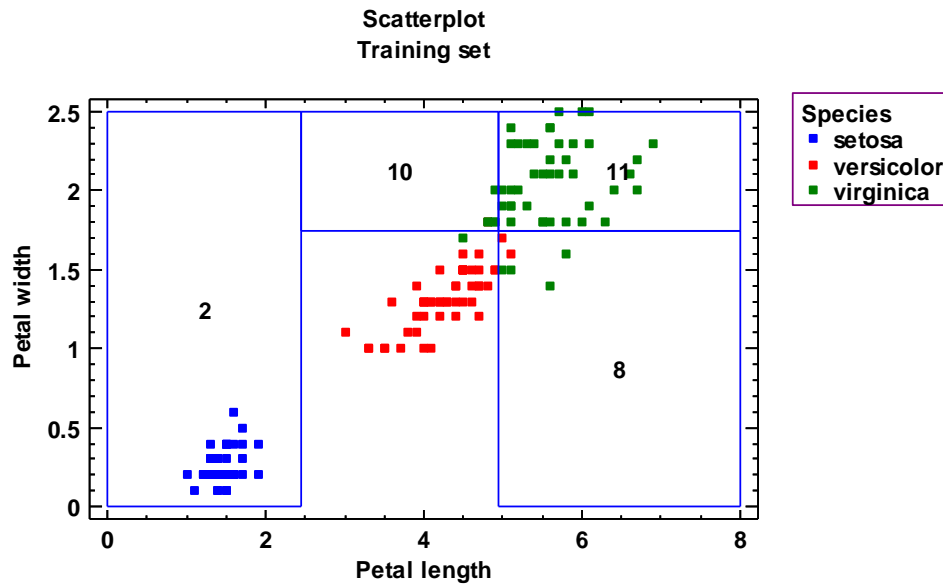
If some cases have been reserved for validation, a separate table of predictions is displayed. In such cases, the denominator of equations (1) and (2) is set equal to the number of observations in the validation set. Note that it is possible for the validation RMD to equal infinity if an observation in the validation set corresponds to a class given 0 probability by the fitted tree.

## 2-D Scatterplot

This graph plots the observations in the training set with respect to any 2 of the predictor variables:

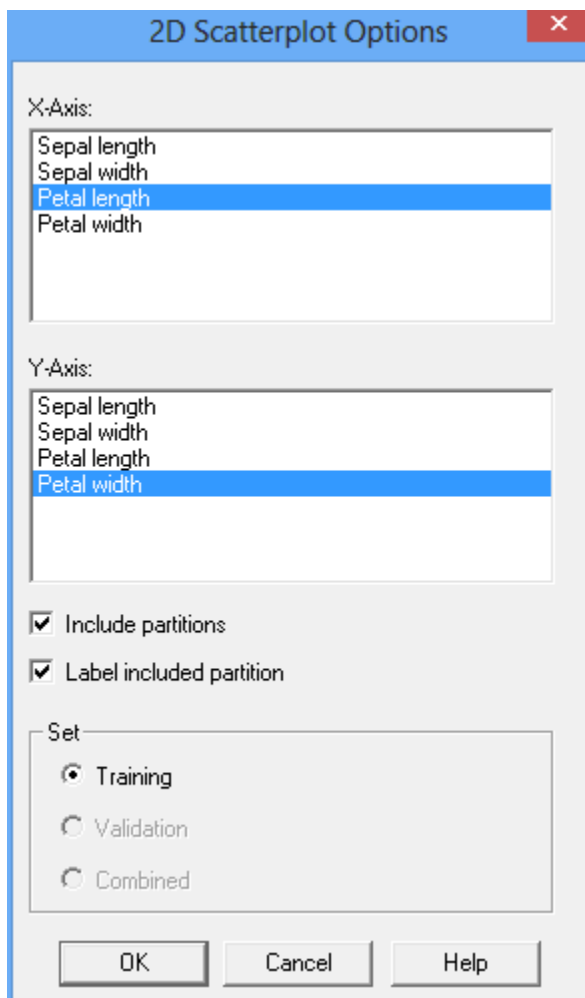


If the fitted tree partitions the observations into sections using only the 2 variables specified, the partitions are shown. For example, the plot below shows the points on axes defined by the 2 petal dimensions:



The partition on the left, which corresponds to terminating node #2, is based solely on *Petal length* and is uniquely *setosa*. The partition at the bottom right, which corresponds to terminating node #8, is based on both *Petal length* and *Petal width* and corresponds to both *versicolor* and *virginica*. The open section near the middle bottom of the plot is not partitioned since it involves variables other than just the 2 displayed on the X and Y axes.

### Pane Options

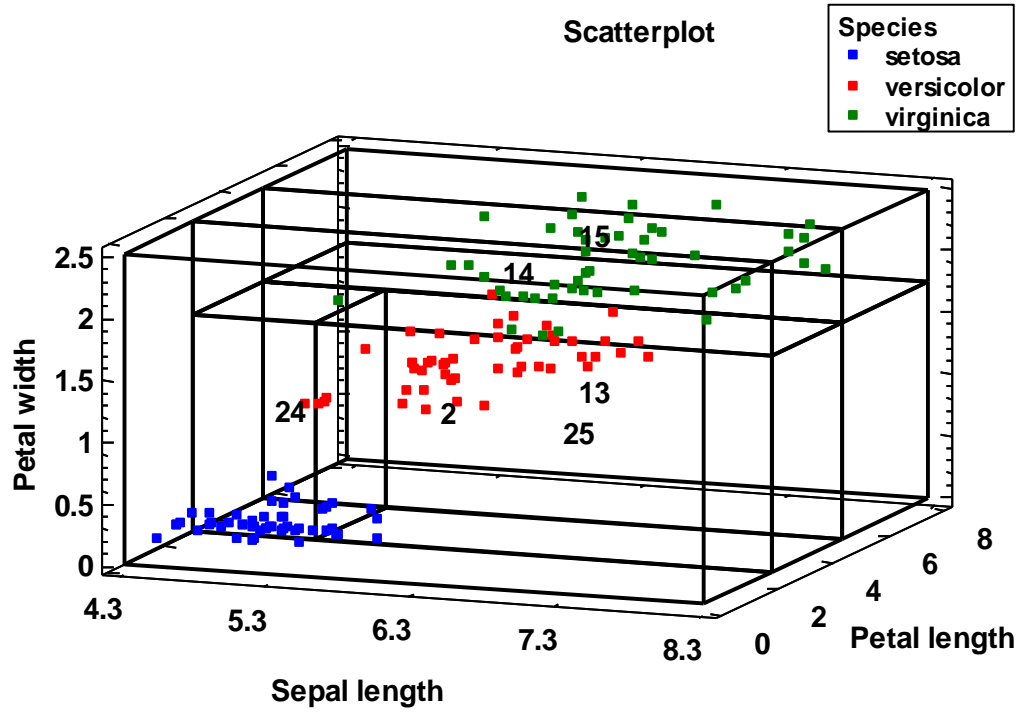


- **X-Axis:** variable to plot on the X axis.
- **Y-Axis:** variable to plot on the Y axis.
- **Include partitions:** if selected, the partitions are displayed.
- **Label included partitions:** if selected, the node numbers corresponding to included partitions are displayed.
- **Set:** set of points to be displayed on the plot.

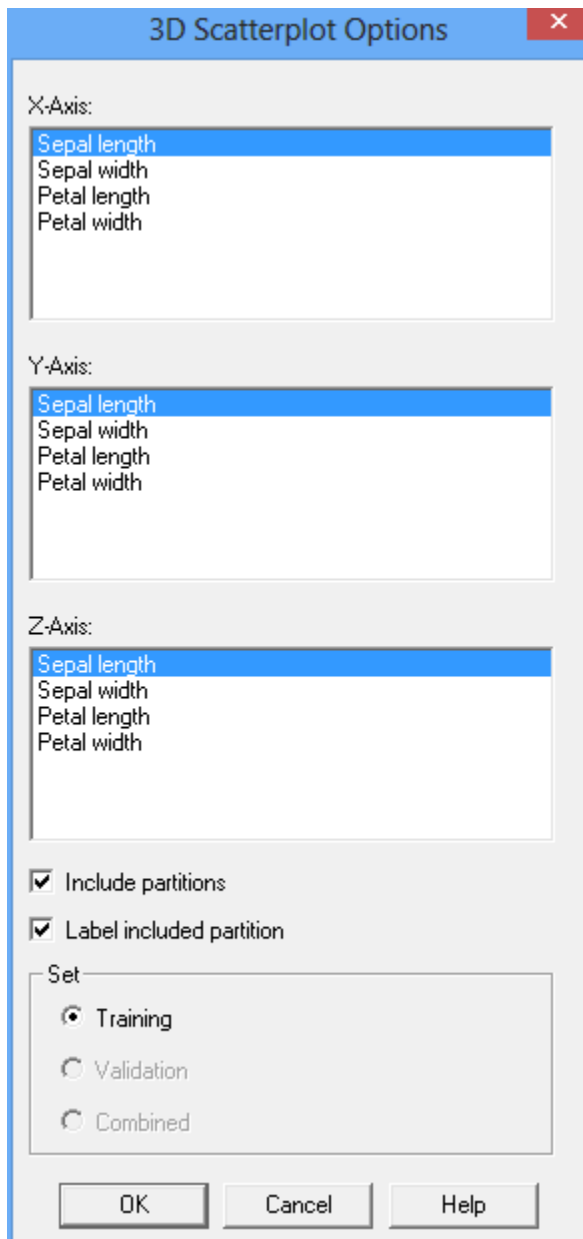
### 3-D Scatterplot

This graph plots the observations in the training set with respect to any 3 of the predictor variables:





Pane Options



- **X-Axis:** variable to plot on the X axis.
- **Y-Axis:** variable to plot on the Y axis.
- **Z-Axis:** variable to plot on the Z axis.
- **Include partitions:** if selected, the partitions are displayed.
- **Label included partitions:** if selected, the node numbers corresponding to included partitions are displayed.
- **Set:** set of points to be displayed on the plot.

## Classification Table

This table shows how well the fitted model performs in classifying the observations:

Classification Table				
Training set n=150				
Actual	Group	Predicted		
Species	Size	<i>setosa</i>	<i>versicolor</i>	<i>virginica</i>
<i>setosa</i>	50	50	0	0
		(100.00%)	( 0.00%)	( 0.00%)
<i>versicolor</i>	50	0	47	3
		( 0.00%)	( 94.00%)	( 6.00%)
<i>virginica</i>	50	0	1	49
		( 0.00%)	( 2.00%)	( 98.00%)

Percent of training cases correctly classified: **97.33%**

It shows:

- **Actual Species:** There is a row for each level of the dependent variable.
- **Group Size:** the number of observations in the training set that fall in the specified class.
- **Predicted:** the number of observations in the training set that were predicted to fall in the specified class.

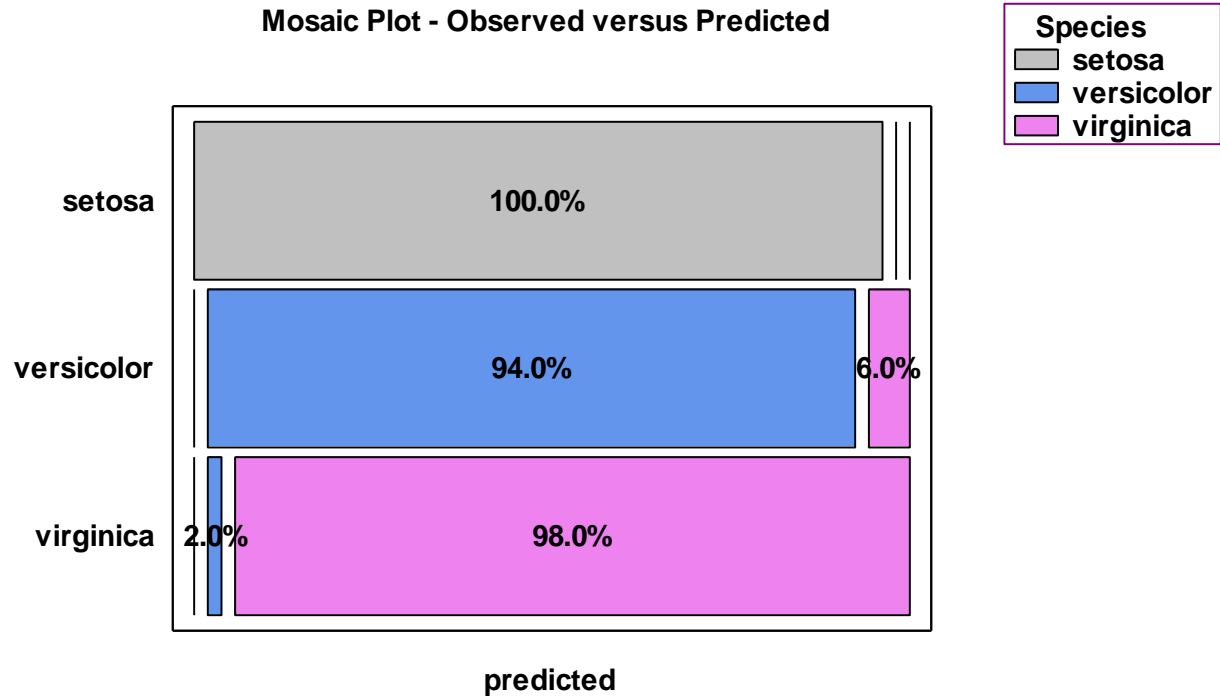
The percentage of correctly classified observations is displayed for each level of the dependent variable and for all of the observations combined.

For example, there were 50 irises of the species *virginica*. All but 1 were correctly predicted to be *virginica* except for 1 that was predicted to be *versicolor*. Of all 150 irises, 97.33% were correctly classified.

A separate table is displayed for any observations withheld for validation purposes.

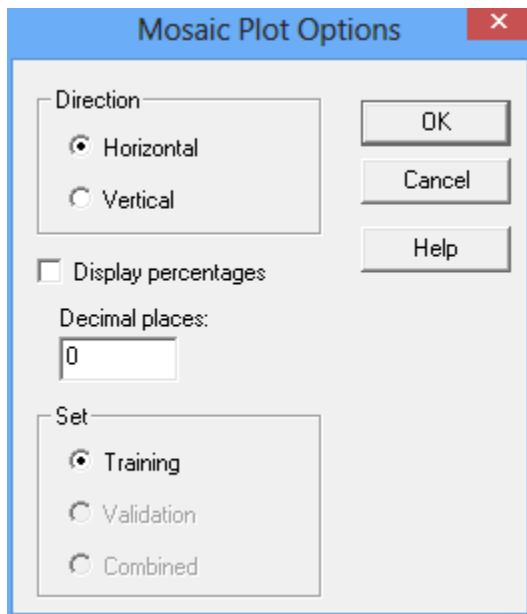
## Observed versus Predicted

When fitting a classification tree, this graph creates a mosaic plot.



By default, the mosaic plot contains a row for each level of the dependent variable. Bars are drawn in each row with length proportional to the number of times observations at that level were predicted to be of each class. The plot above shows that *setosa* was predicted correctly all the time. *Veriscolor* was occasionally predicted to be *virginica*, while *virginica* was occasionally predicted to be *versicolor*.

### Pane Options

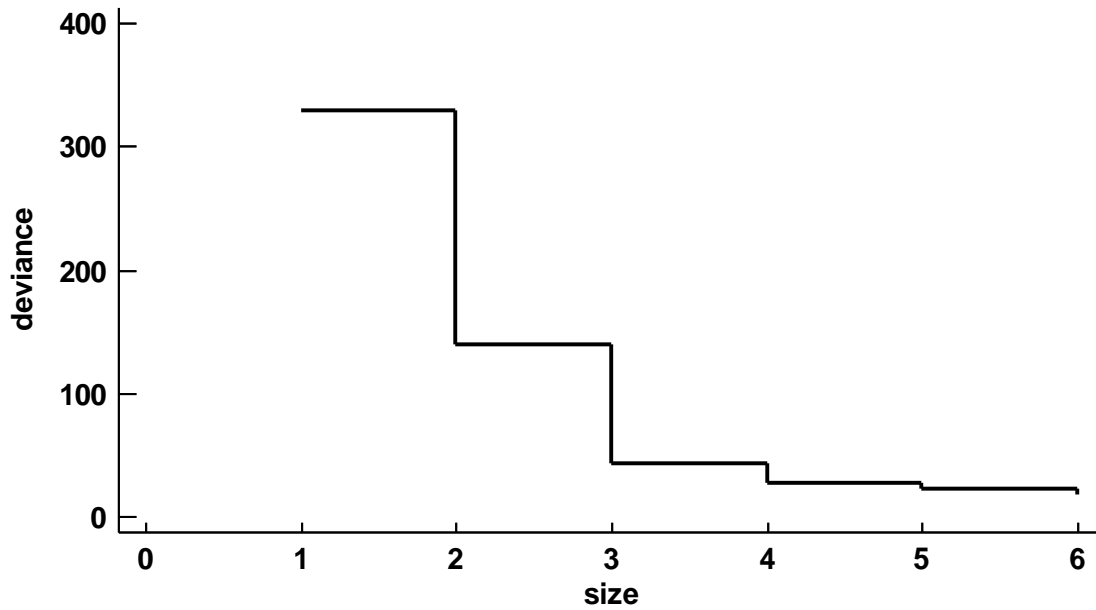


- **Direction:** orientation of the bars.
- **Display percentage:** whether the plot should display the percentage corresponding to each bar.
- **Set:** set of points to be displayed on the plot.

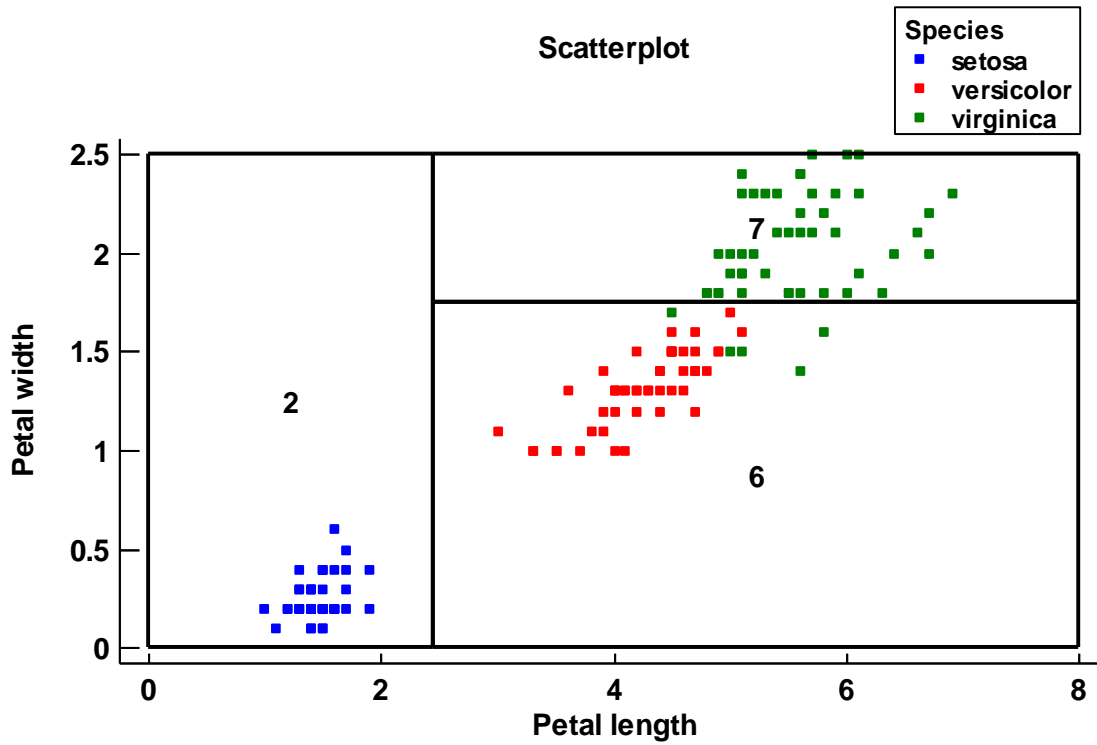
## Deviance Plot

The deviance plot shows the magnitude of the error associated with a tree containing various numbers of leaves (terminating nodes).

Deviance Plot



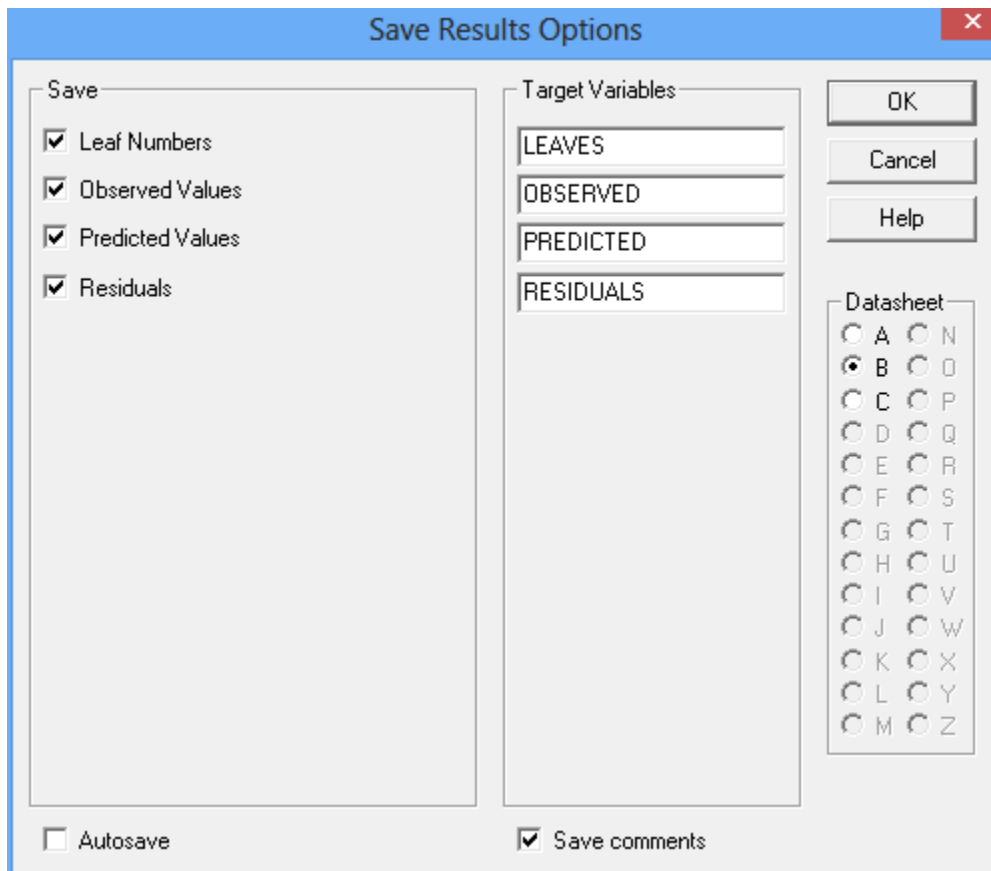
Large reductions due to the inclusion of another node show that the node had a significant reduction on the misclassification rate. The plot above shows that a tree with 3 leaves does almost as well as a tree with 6 leaves. Using *Analysis Options* to reduce the number of leaves to 3 results in the following partitions:



Such a tree uses only *Petal width* and *Petal length* to classify the observations. The pruned tree misclassifies 6 observations compared to the original tree that misclassified only 4 observations.

## Save Results

Selected output may be saved in a Statgraphics datasheet by pressing the *Save Results* button on the analysis toolbar. The following dialog box will be presented:



The dialog box is titled "Save Results Options" and contains the following sections:

- Save:** A list of items to be saved, each with a checked checkbox:
  - Leaf Numbers
  - Observed Values
  - Predicted Values
  - Residuals
- Target Variables:** A list of text boxes containing the names of the target variables:
  - LEAVES
  - OBSERVED
  - PREDICTED
  - RESIDUALS
- Datasheet:** A grid of radio buttons labeled with letters A through Z. Radio button 'B' is selected.
- Buttons:** OK, Cancel, and Help buttons are located on the right side.
- Checkboxes:**
  - Autosave
  - Save comments

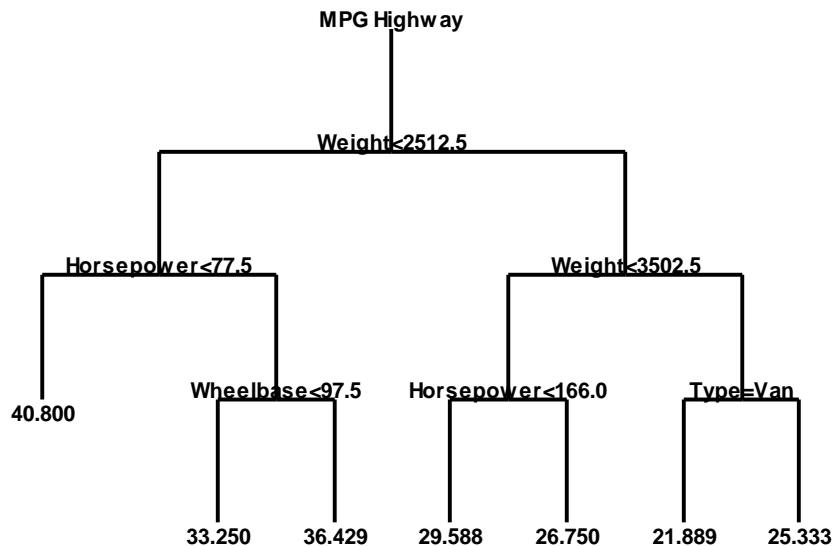
To save results, select:

- **Save:** select the items to be saved.
- **Target Variables:** enter names for the columns to be created.
- **Datasheet:** the datasheet into which the results will be saved.
- **Autosave:** if checked, the results will be saved automatically each time a saved StatFolio is loaded.
- **Save comments:** if checked, comments for each column will be saved in the second line of the datasheet header.



## Example 2

The second example fits a regression tree designed to predict *MPG Highway* from the *93cars.sgd* data file using 5 predictor variables. The default tree (shown below) uses 4 of the 5 predictors:



There are 7 terminating nodes.

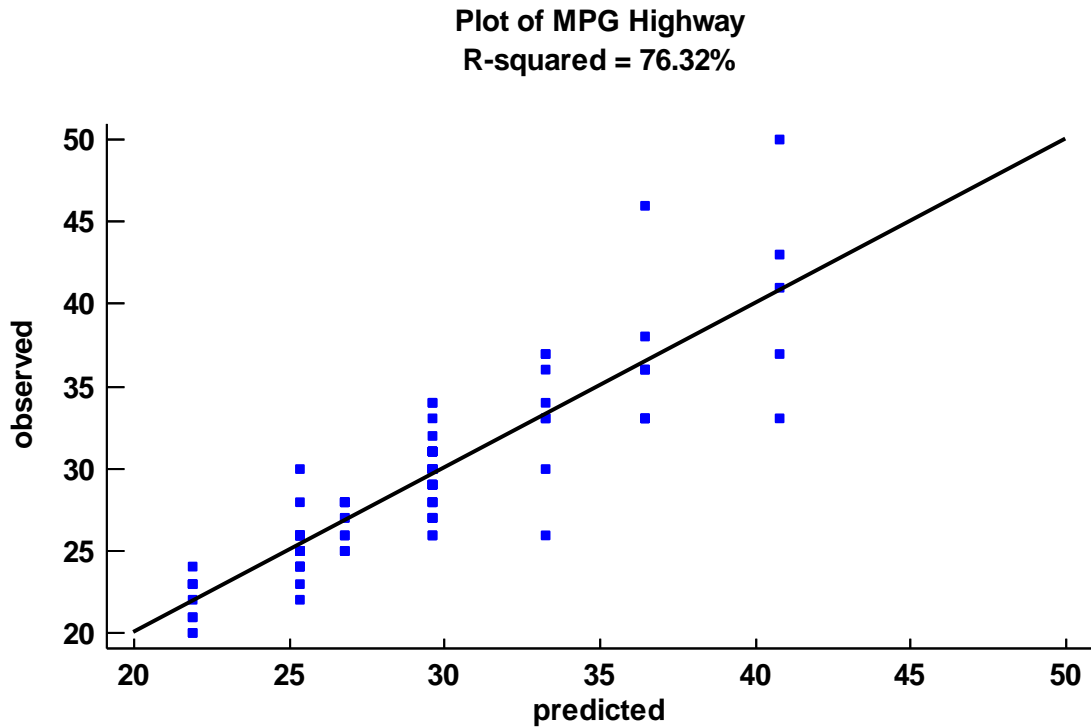
There are a few differences between a regression tree and a classification tree:

1. When displaying the report of observed and predicted values, residuals are also calculated and displayed.

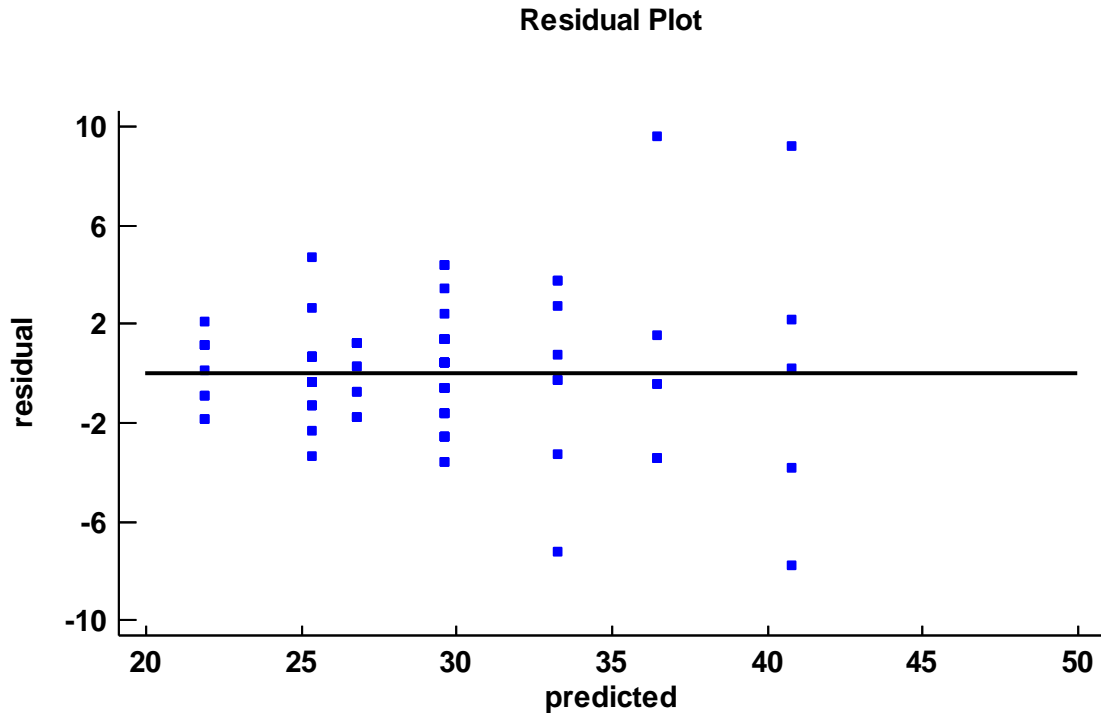
Predictions and Residuals				
Training set n=93				
Row	Leaf	Predicted	Observed	Residual
1	9	29.5882	31.0	1.41176
2	13	25.3333	25.0	-0.33333
3	10	26.75	26.0	-0.75
4	10	26.75	26.0	-0.75
5	13	25.3333	30.0	4.66667
...	...	...	...	...

The predicted values equal the average of all observations in the training set that arrive at a given leaf.

2. No table of node probabilities or classification table is created.
3. The graph of *Observed versus Predicted* values produces a scatterplot rather than a mosaic chart:



4. An additional graph is available plotting the *Residuals versus Predicted* values:



## References

Brieman, L., Friedman, J., Stone, C.J. and Olshen, R.A. (1998) Classification and Regression Trees. Wadsworth.

Fisher, R.A. (1936). “The use of multiple measurements in taxonomic problems.” Ann. Eugenics 7, Pt. II, 179-188.

Lock, R. H. (1993) “1993 New Car Data”. Journal of Statistics Education, vol. 1, no. 1.

R Package “tree” (2015) <https://cran.r-project.org/web/packages/tree/tree.pdf>

Ripley, B.D. (1996) Pattern Recognition and Neural Networks. Cambridge University Press.