statgraphics®
centurion

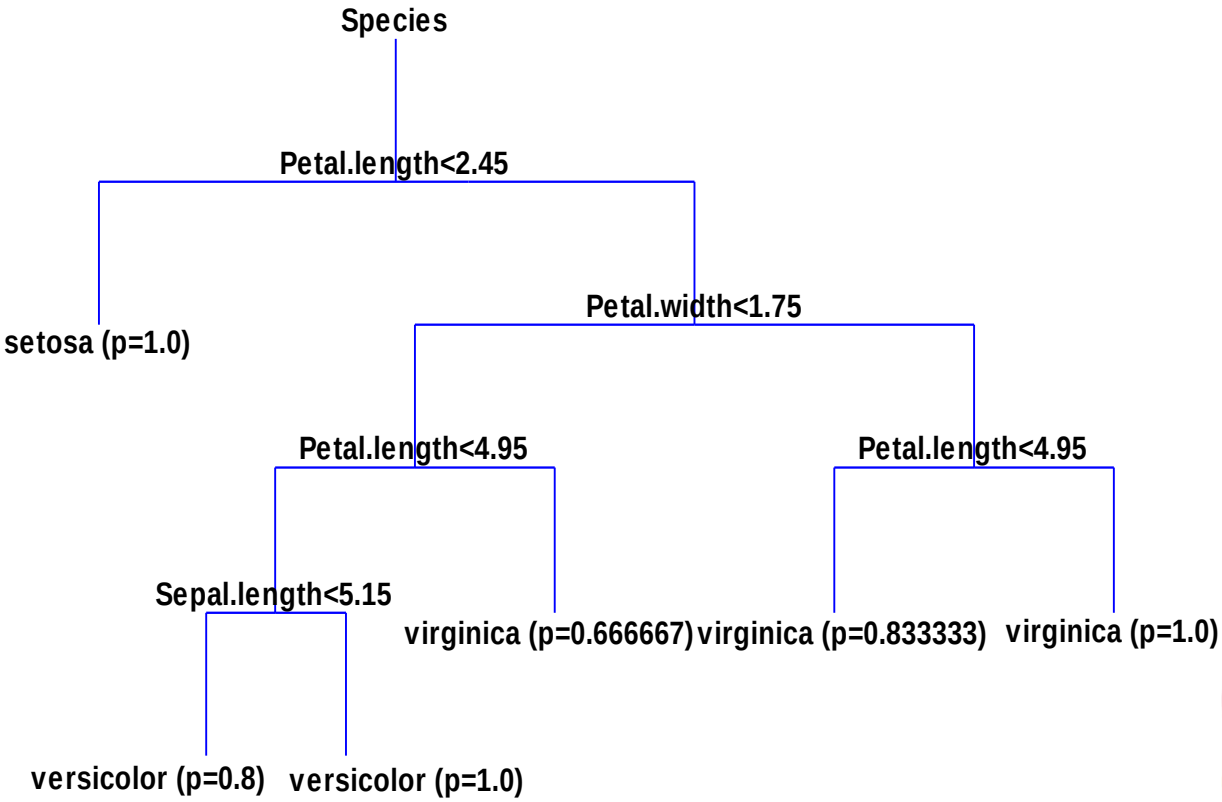
Fitting Classification and Regression Trees Using Statgraphics and R

Presented by
Dr. Neil W. Polhemus

Classification and Regression Trees

- Machine learning methods used to construct predictive models from data.
- Recursively partitions the data space using simple binary decisions.
- Commonly portrayed as a tree with a split at each decision node.

Example – Fisher's Iris Data



statgraphics®
statgraphics.com
centurion

Basic Model Structure

- Y: variable to be predicted
  - If categorical, we construct a classification tree.
  - If continuous, we construct a regression tree.

- X_1, X_2, … X_p: predictor variables
  - May be either categorical or continuous.

statgraphics®
statgraphics.com centurion

Partitioning Algorithm

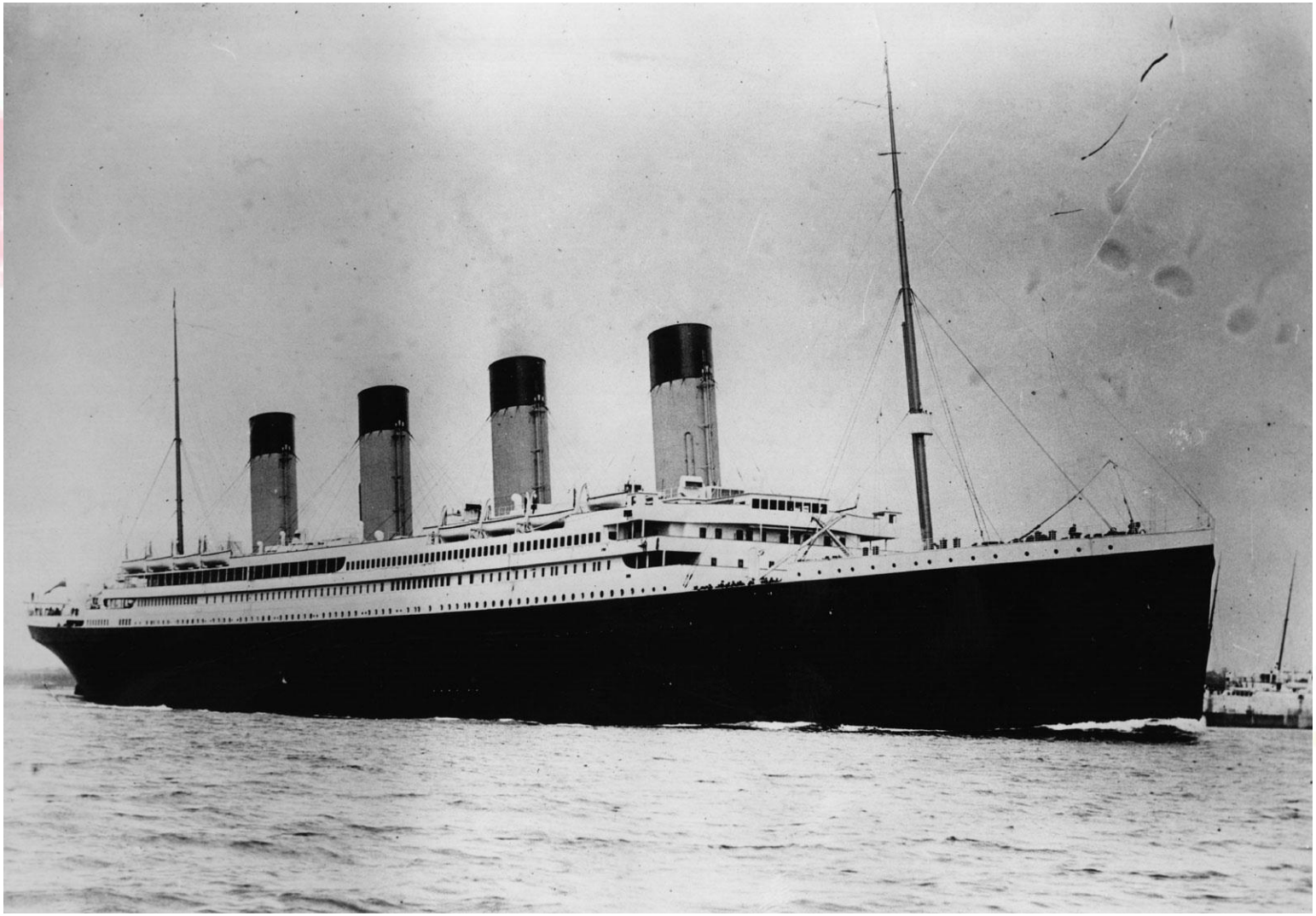
- Start at the root node.

- Amongst all variables X_j, find the split that minimizes the resulting average within-node deviance.
  - For a continuous variable, the split is of the form $X_j < c$.
  - For a discrete variable, the split divides the possible values into 2 distinct groups.

- If one or more stopping criteria are met after the split, stop. Otherwise consider splitting the child nodes.

statgraphics® centurion
statgraphics.com

RMS Titanic



statgraphics®
statgraphics.com centurion

Sample Data File

C:\Data\Webinars\titanic for trees.sgd

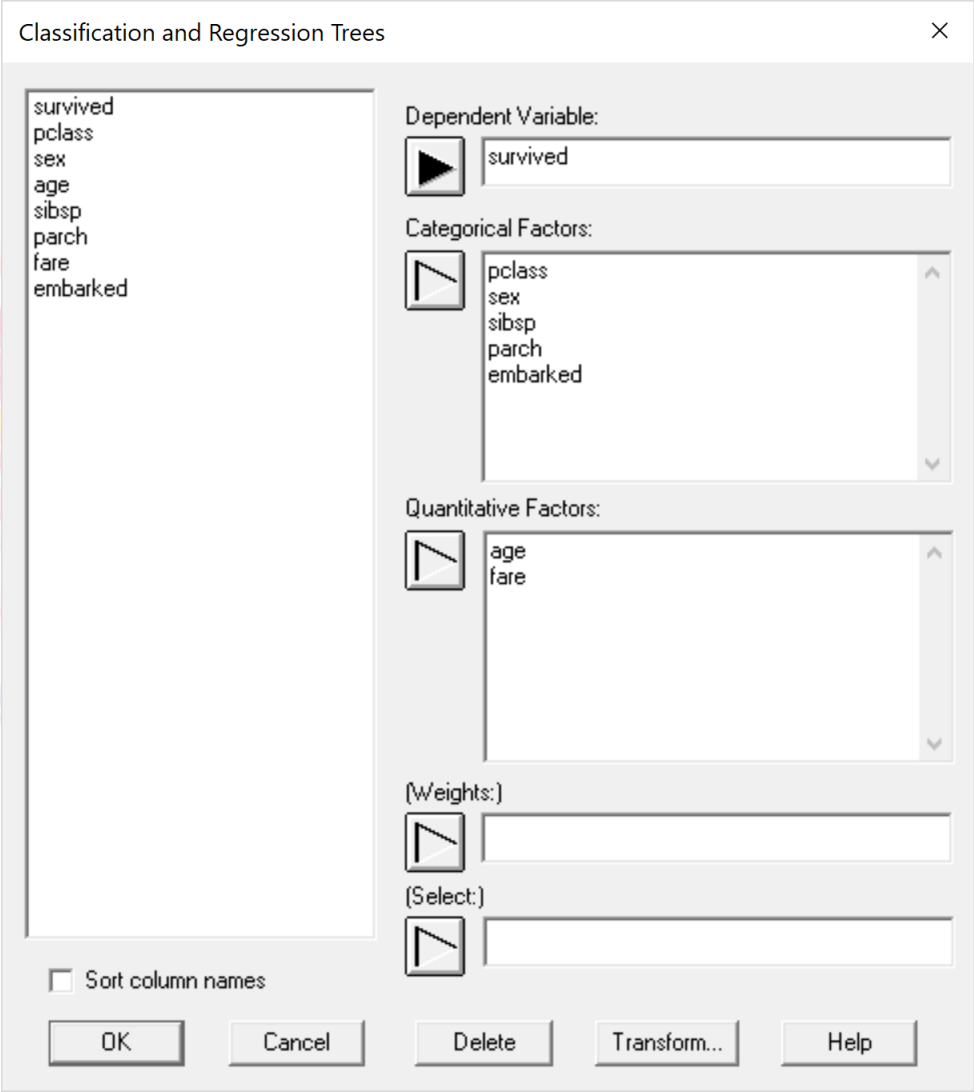
	survived	pclass	sex	age	sibsp	parch	fare	embarked
		1, 2 or 3			# siblings aboard	# parents or children aboard	British pounds	Location: C, Q or S
	Character	Numeric	Character	Numeric	Numeric	Numeric	Numeric	Character
1	Yes	1	female	29	0	0	211.3375	S
2	Yes	1	male	0.9167	1	2	151.5500	S
3	No	1	female	2	1	2	151.5500	S
4	No	1	male	30	1	2	151.5500	S
5	No	1	female	25	1	2	151.5500	S
6	Yes	1	male	48	0	0	26.5500	S
7	Yes	1	female	63	1	0	77.9583	S
8	No	1	male	39	0	0	0.0000	S
9	Yes	1	female	53	2	0	51.4792	S
10	No	1	male	71	0	0	49.5042	C
11	No	1	male	47	1	0	227.5250	C
12	Yes	1	female	18	1	0	227.5250	C
13	Yes	1	female	24	0	0	69.3000	C
14	Yes	1	female	26	0	0	78.8500	S
15	Yes	1	male	80	0	0	30.0000	S
16	No	1	male		0	0	25.9250	S

titanic for trees B C

Source: Frank Harrell and Thomas Cason, University of Virginia
http://biostat.mc.vanderbilt.edu/twiki/pub/Main/DataSets/titanic.html

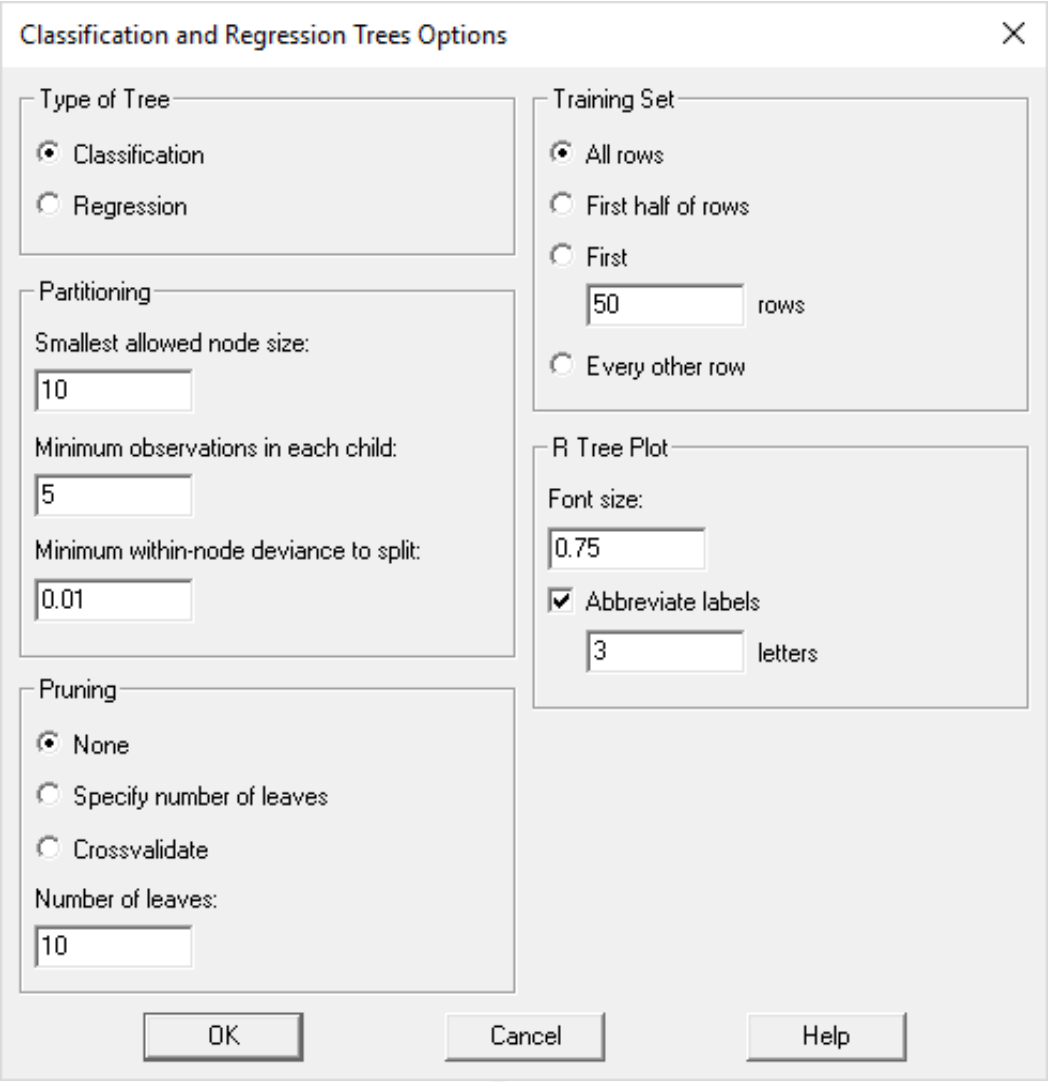
statgraphics® centurion
statgraphics.com

Data Input



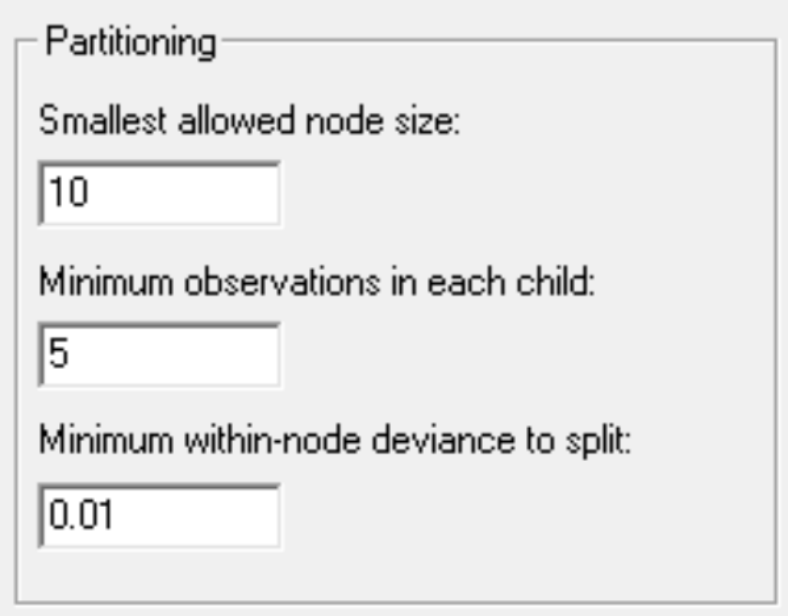
statgraphics® centurion
statgraphics.com

Analysis Options



statgraphics®
statgraphics.com
centurion

Partitioning Options



Node Impurity

- Measure the impurity in a tree using the residual mean deviance (RMD).

  n = number of observations in training set

  k = number of leaves

  $p_{i,j}$ = proportion of data of same type as i at its assigned leaf j

  $\hat{Y}_i$ = predicted value of observation i

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

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

statgraphics® centurion
statgraphics.com

Analysis Window

Classification and Regression Trees

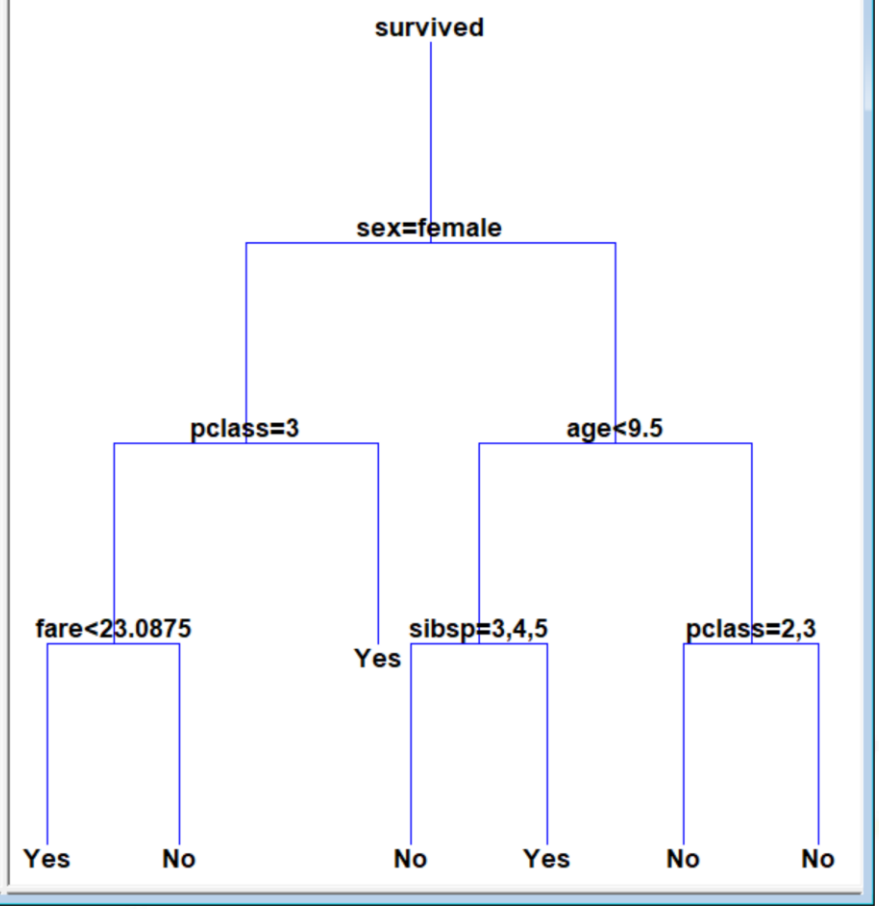
Classification and Regression Trees

```
d<-read.csv("C:\\\\Users\\\\NEIL~1.STA\\\\AppData\\\\Local\\\\Temp\\\
setwd("C:\\Users\\NEIL~1.STA\\AppData\\Local\\Temp\\")
library("tree")
treefit=tree(survived~pclass+sex+sibsp+parch+embarked+age+fare,contrc
)
summary(treefit)

##
## Classification tree:
## tree(formula = survived ~ pclass + sex + sibsp + parch + embarked
##     age + fare, data = d, control = tree.control(nobs = 1309,
##     mincut = 5, minsize = 10, mindev = 0.01))
## Variables actually used in tree construction:
## [1] "sex"    "pclass" "fare"   "age"    "sibsp"
## Number of terminal nodes:  7
## Residual mean deviance:  0.8524 = 884.7 / 1038
## Misclassification error rate: 0.1847 = 193 / 1045

plot(treefit)
text(treefit,pretty=0,cex=0.75)



p<-prune.tree(treefit)
newd<-read.csv("C:\\\\Users\\\\NEIL~1.STA\\\\AppData\\\\Local\\\\Temp
pd<-predict(treefit,newd,type="where")
write.table(treefit$frame,file="C:\\Users\\NEIL~1.STA\\AppData\\Local
write.table(treefit$where,file="C:\\Users\\NEIL~1.STA\\AppData\\Local
write.table(cbind(p$size,p$dev,p$k),file="C:\\Users\\NEIL~1.STA\\AppI
write.table(pd,file="C:\\Users\\NEIL~1.STA\\AppData\\Local\\Temp\\pre
```

The StatAdvisor

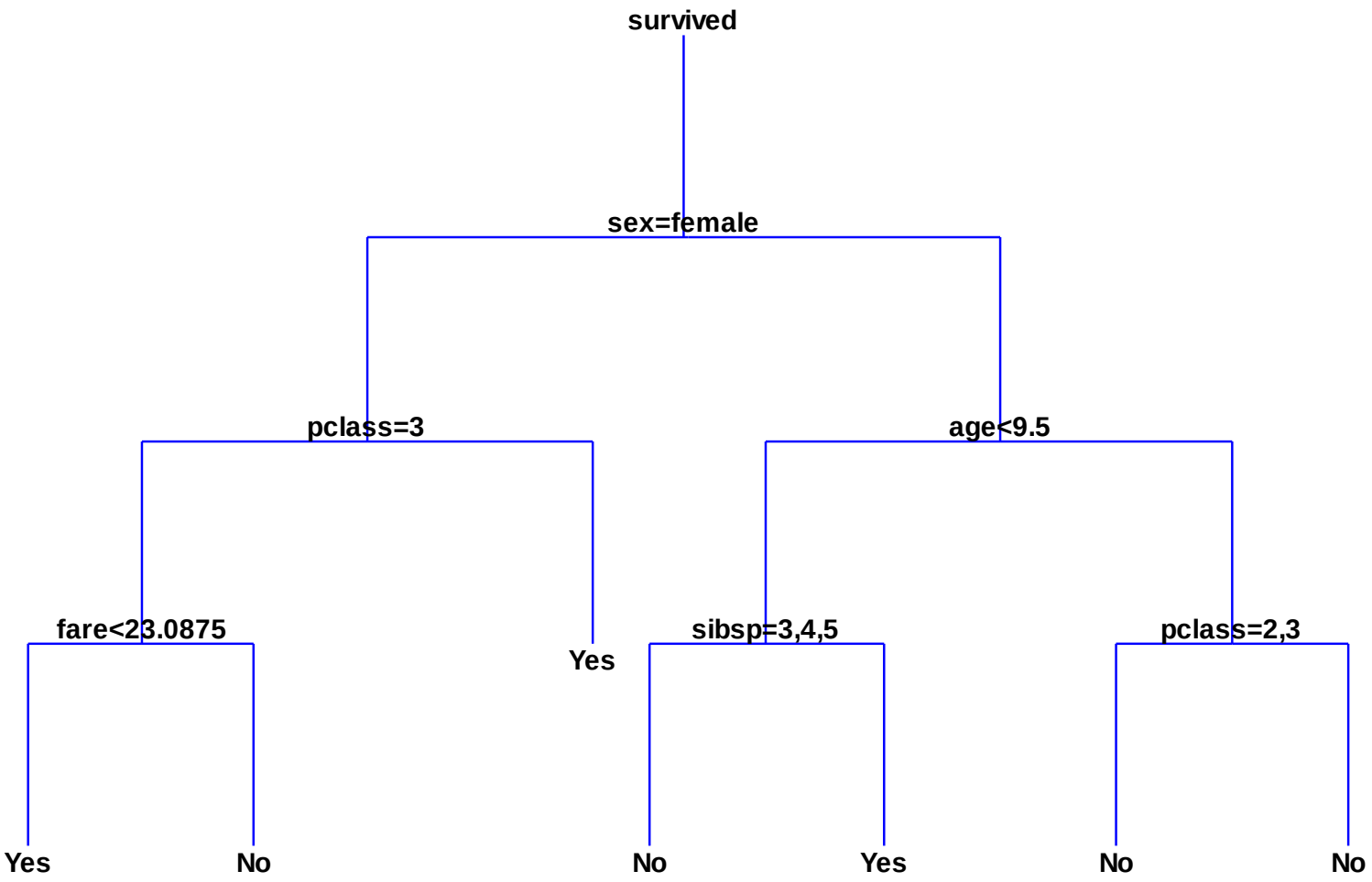
The output above shows the results of instructing the "tree" package in R to construct a classification tree to predic

survived. Use the Analysis Options dialog box to control how large a tree is created.



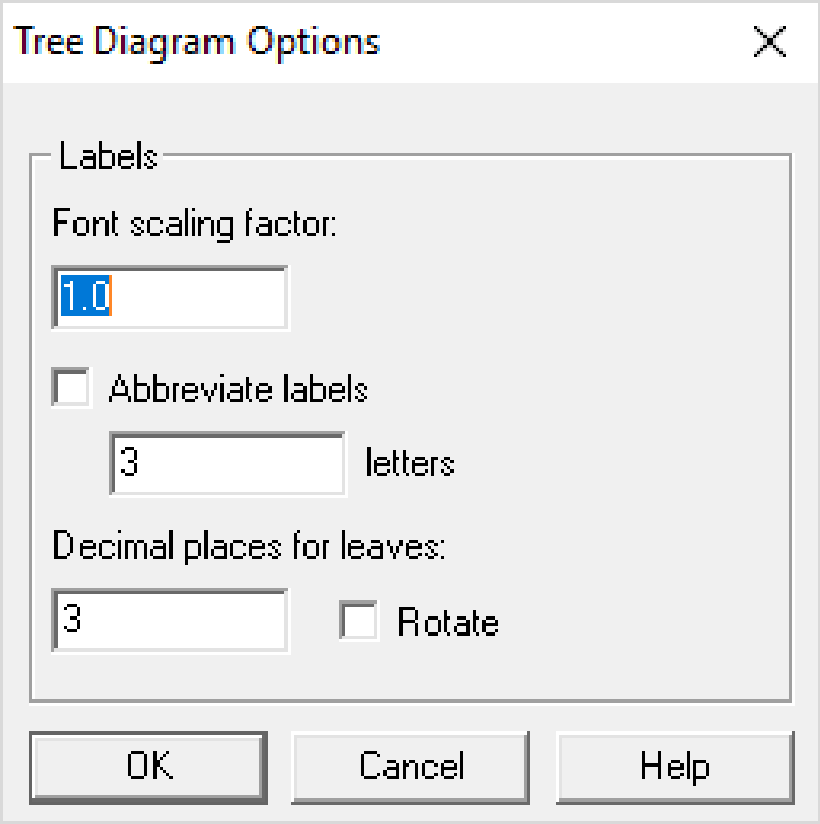
statgraphics® centurion
statgraphics.com

Decision Tree



statgraphics® centurion
statgraphics.com

Decision Tree Options



statgraphics®
statgraphics.com
centurion

Tree Structure

Node	Label	Size	Deviance	survived	Splits left	Splits right
1	sex	1045	1413.57	No	female	male
2	pclass	388	434.161	Yes	3	1,2
3	fare	152	210.295	No	<23.0875	>23.0875
4	<leaf>	131	180.986	Yes		
5	<leaf>	21	13.2087	No		
6	<leaf>	236	117.01	Yes		
7	age	657	667.388	No	<9.5	>9.5
8	sibsp	43	58.466	Yes	3,4,5	0,1,2
9	<leaf>	16	7.48133	No		
10	<leaf>	27	18.8369	Yes		
11	pclass	614	577.291	No	2,3	1
12	<leaf>	466	357.901	No		
13	<leaf>	148	189.319	No		

* Note: based on complete cases only.

statgraphics®
statgraphics.com centurion

Node Probabilities

Node	Label	Size	yprob.No	yprob.Yes
1	sex	1045	0.591388	0.408612
2	pclass	388	0.247423	0.752577
3	fare	152	0.526316	0.473684
4	<leaf>	131	0.465649	0.534351
5	<leaf>	21	0.904762	0.0952381
6	<leaf>	236	0.0677966	0.932203
7	age	657	0.794521	0.205479
8	sibsp	43	0.418605	0.581395
9	<leaf>	16	0.9375	0.0625
10	<leaf>	27	0.111111	0.888889
11	pclass	614	0.820847	0.179153
12	<leaf>	466	0.871245	0.128755
13	<leaf>	148	0.662162	0.337838

* Note: based on complete cases only.

statgraphics® centurion
statgraphics.com

Classification Table

Training set n=1309

Actual	*Group*	*Predicted*	
survived	*Size*	*No*	*Yes*
No	809	709	100
		(87.64%)	(12.36%)
Yes	500	142	358
		(28.40%)	(71.60%)

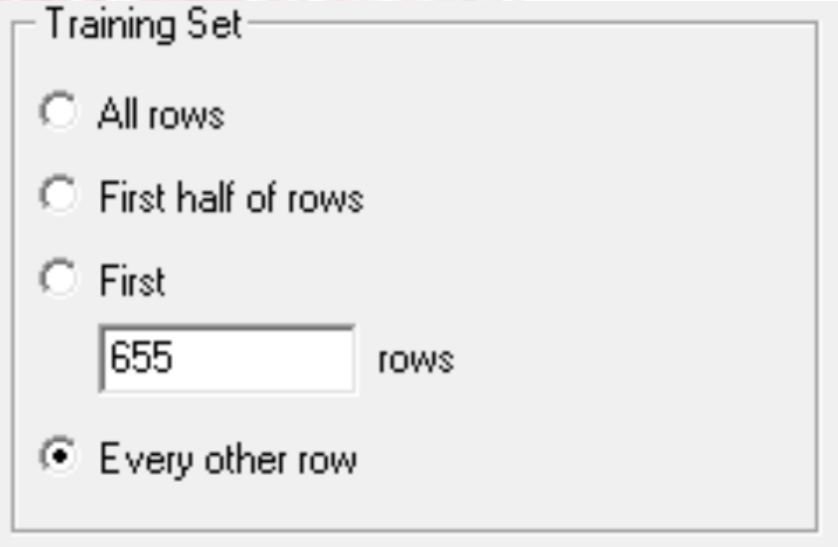
Percent of training cases correctly classified: 81.51%

* Note: based on both complete and partial cases.

statgraphics® centurion
statgraphics.com

Training and Validation Sets

- May separate the data into 2 sets:
  - Training set used to build the tree.
  - Test set used to estimate the tree misclassification percentages.



statgraphics®
statgraphics.com
centurion

Compare Results

Training set n=655

Actual	*Group*	*Predicted*	
survived	*Size*	*No*	*Yes*
No	408	400	8
		(98.04%)	(1.96%)
Yes	247	117	130
		(47.37%)	(52.63%)

Percent of training cases correctly classified: 80.92%

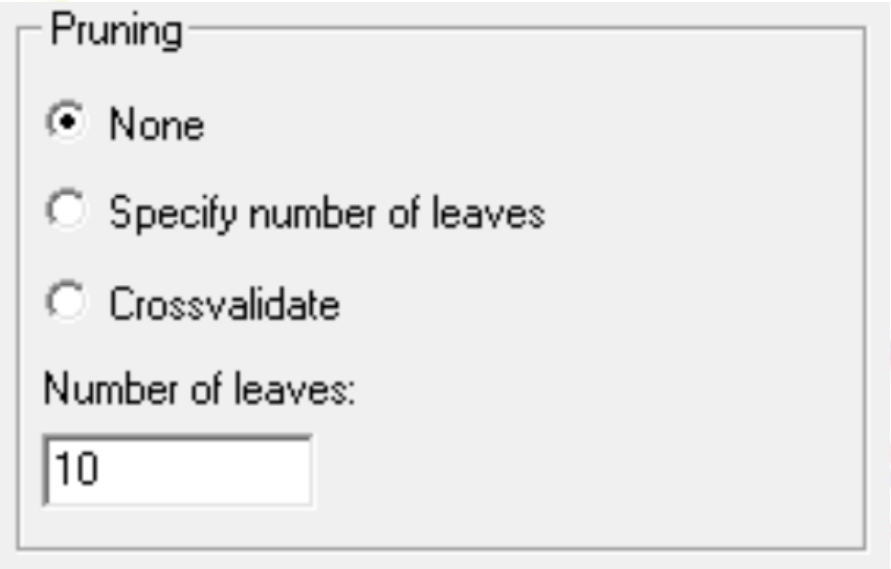
Validation set n=654

Actual	*Group*	*Predicted*	
survived	*Size*	*No*	*Yes*
No	401	387	14
		(96.51%)	(3.49%)
Yes	253	124	129
		(49.01%)	(50.99%)

Percent of validation cases correctly classified: 78.90%

Pruning Options

- Reduces the complexity of the tree by removing branches.



Pruning by Cross-validation

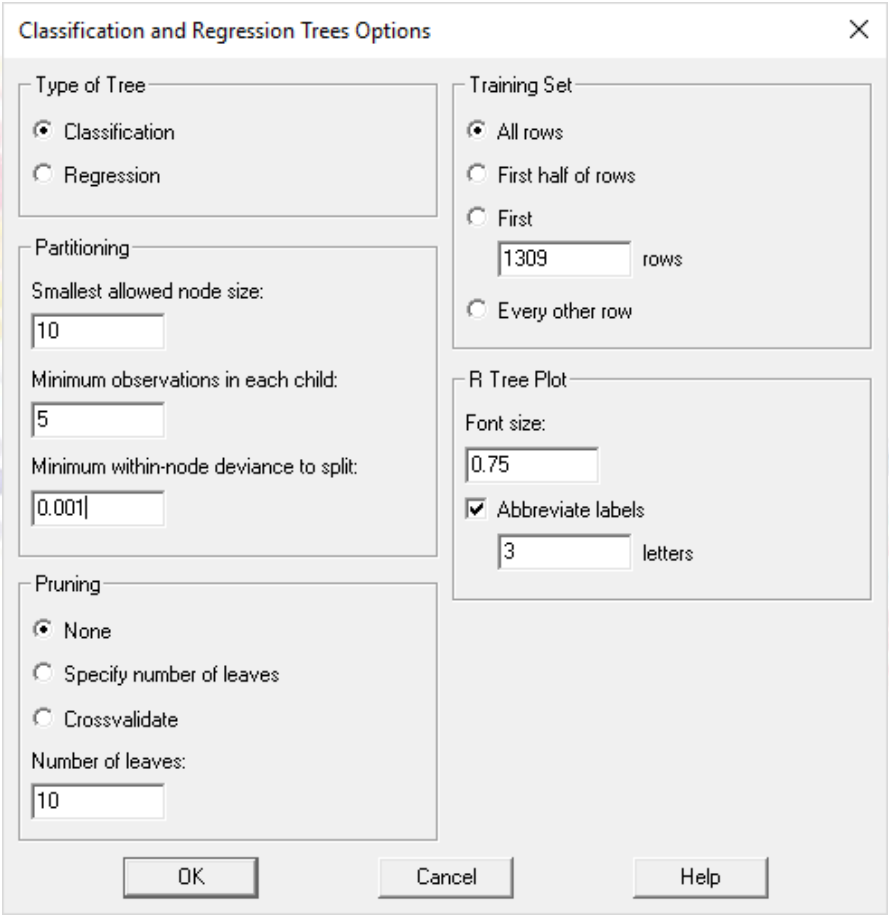
- Runs a 10-fold cross-validation experiment. Builds 10 trees, leaving out 10% of the data each time, and averages the results.
- Uses all of the observations for both training and validation.
- Can be used to determine the optimal size for the tree by increasing the number of leaves until you see warnings such as:

```
## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size
```

statgraphics® centurion
statgraphics.com

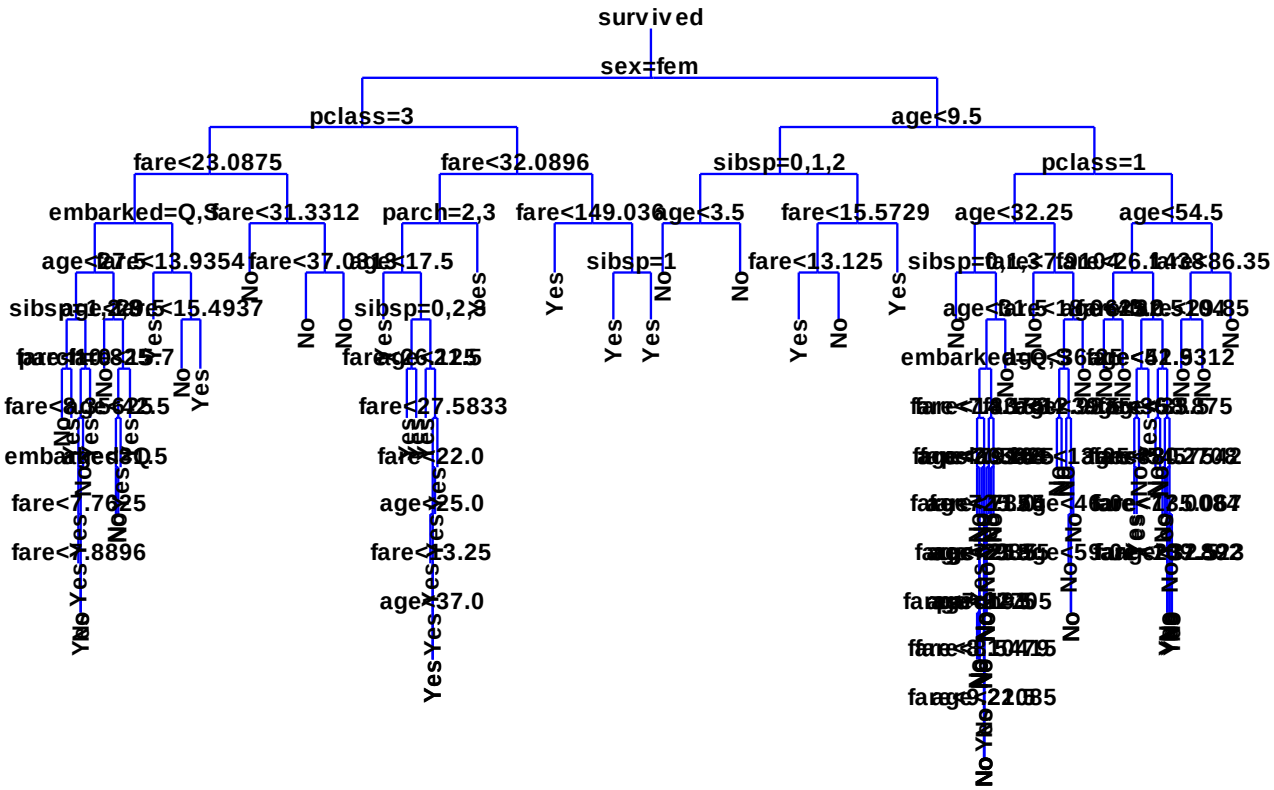
Pruning Example

- First reduce within-node deviance to fit a complex tree.



statgraphics® centurion
statgraphics.com

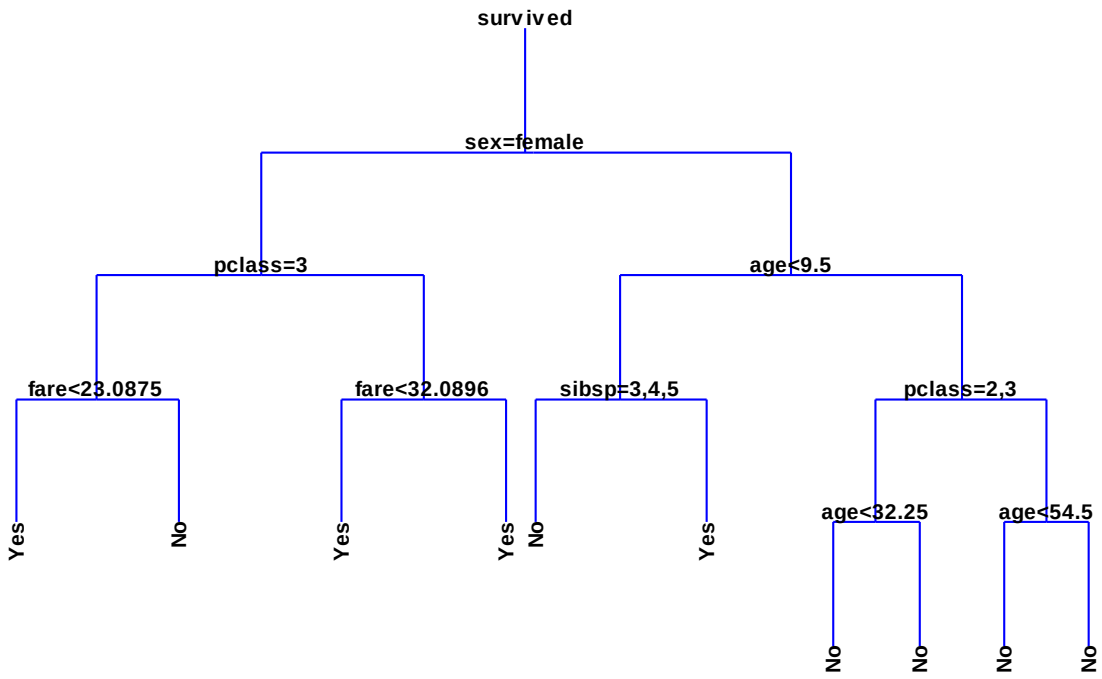
Pruning Example



statgraphics®
statgraphics.com
centurion

Pruning Example

- Now reduce the number of leaves to 10 and select cross-validation.



statgraphics®
statgraphics.com centurion

10 Leaves is Too Complex

Classification and Regression Trees - survived

Classification and Regression Trees

```
d<-read.csv("C:\\\\Users\\\\NEIL~1.STA\\\\AppData\\\\Local\\\\Temp\\\\data.csv",dec=".",sep=",",stri
setwd("C:\\Users\\NEIL~1.STA\\AppData\\Local\\Temp\\")
library("tree")
treefit=tree(survived~pclass+sex+sibsp+parch+embarked+age+fare,control=tree.control(nobs=1309,mincut
)
treefit=cv.tree(treefit,best=10)

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size

## Warning in prune.tree(structure(list(frame = structure(list(var =
## structure(c(3L, : best is bigger than tree size
```

statgraphics® centurion
statgraphics.com

Finding Optimal Number of Leaves

- Step 1: Copy script from Statgraphics to Word and modify last line.

```
d<-
read.csv("C:\\\\Users\\\\NEIL~1.STA\\\\AppData\\\\Local\\\\Temp\
\\\data.csv",dec=".",sep=",",stringsAsFactors=TRUE,colClasses=c(
"pclass"="factor","sex"="factor","sibsp"="factor","parch"="facto
r","embarked"="factor"))
setwd("C:\\Users\\NEIL~1.STA\\AppData\\Local\\Temp\\")
library("tree")
treefit=tree(survived~pclass+sex+sibsp+parch+embarked+age+fare,c
ontrol=tree.control(nobs=1309,mincut=5,minsize=10,mindev=0.001),
data=d
)
cv.tree(treefit)
```

statgraphics® centurion
statgraphics.com

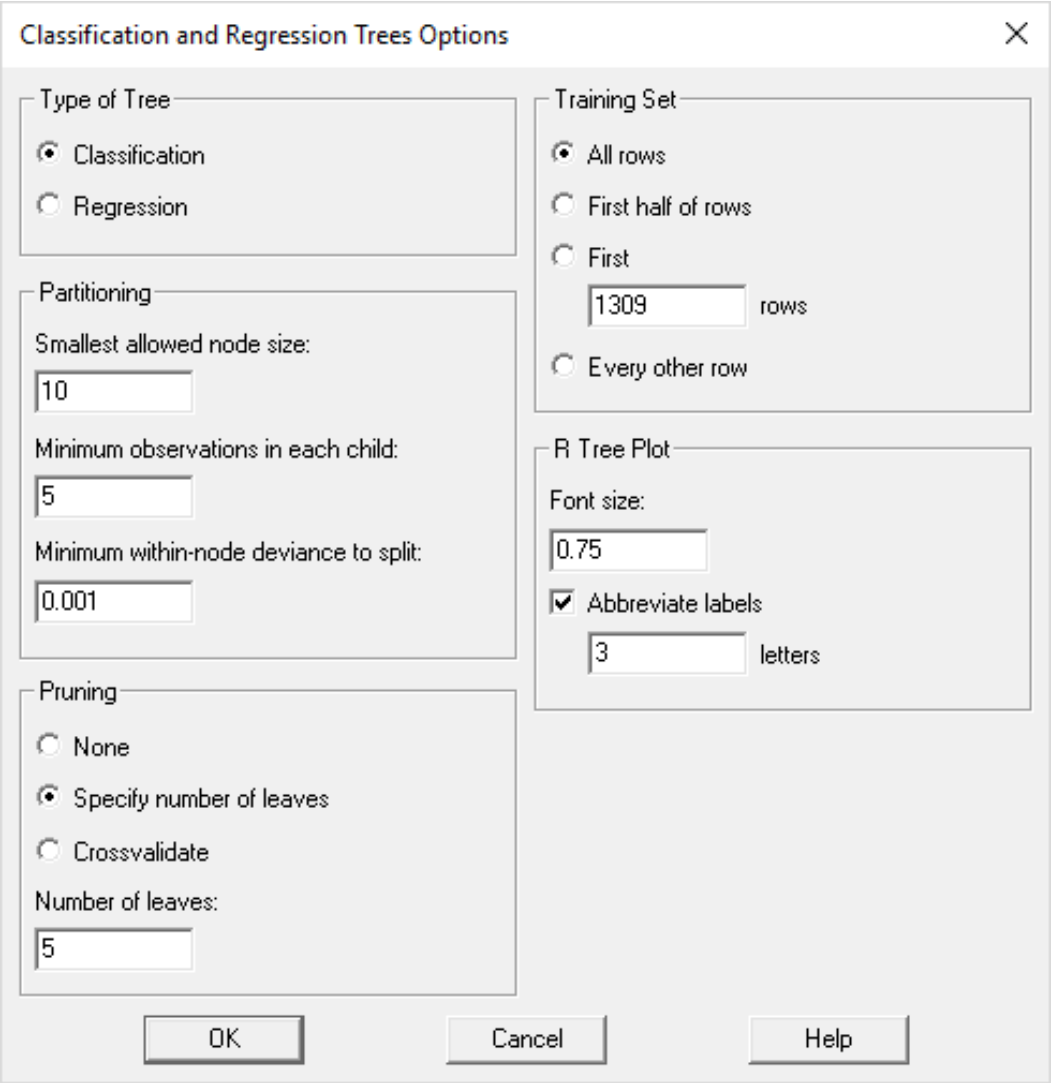
Finding Optimal Number of Leaves

- Step 2: Copy modified script to R and run it. Look for size with minimum deviance.

R Console

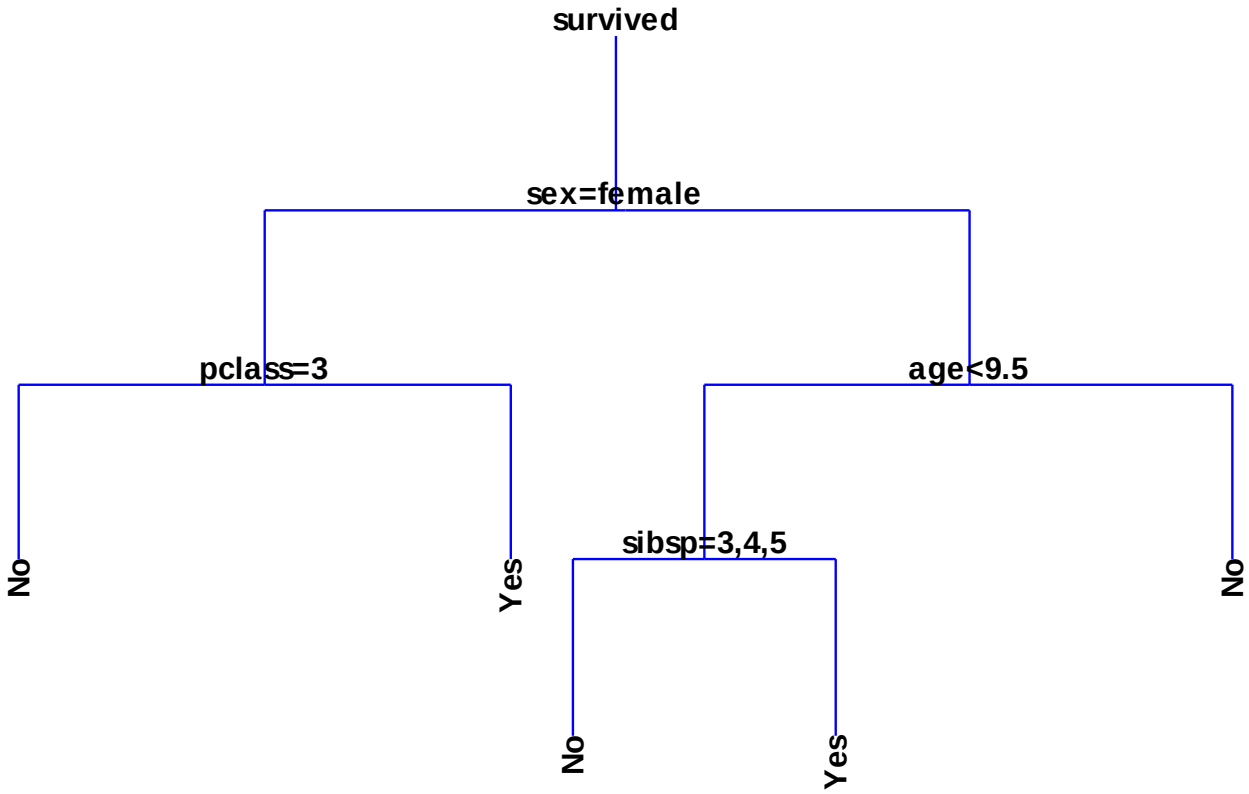
```
> d<-read.csv("C:\\\\Users\\\\NEIL~1.STA\\\\AppData\\\\Local\\\\Temp\\\\data.csv$
> setwd("C:\\Users\\NEIL~1.STA\\AppData\\Local\\Temp\\")
> library("tree")
> treefit=tree(survived~pclass+sex+sibsp+parch+embarked+age+fare,control=tree.co$
+ )
> cv.tree(treefit)
$`size`
 [1] 92 91 90 89 88 87 86 85 83 82 81 80 76 74 73 72 71 70 67 66 65 62 61 60 59
[26] 58 56 54 50 47 44 42 41 40 35 32 31 29 27 24 21 20 18 13 12 10  9  8  7  6
[51]  5  3  2  1

$dev
 [1]  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877
 [8]  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877
[15]  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877
[22]  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877
[29]  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877
[36]  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877
[43]  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877  938.0877
[50]  934.8098 1016.7447 1020.9768 1105.9453 1414.5408
```

Prune Tree



statgraphics®
statgraphics.com
centurion

Final Tree



statgraphics®
statgraphics.com
centurion

Predict Additional Cases

- To make predictions for additional cases, add them to the bottom of the original data, leaving the cell for *Y* blank.

C:\Data\Webinars\titanic for trees.sgd

	survived	pclass	sex	age	sibsp	parch	fare	embarked
		1, 2 or 3			# siblings aboard	# parents or children aboard	British pounds	Location: C, Q or S
	Character	Numeric	Character	Numeric	Numeric	Numeric	Numeric	Character
1304	No	3	male		0	0	14.4583	C
1305	No	3	female	14.5	1	0	14.4542	C
1306	No	3	female		1	0	14.4542	C
1307	No	3	male	26.5	0	0	7.2250	C
1308	No	3	male	27	0	0	7.2250	C
1309	No	3	male	29	0	0	7.8750	S
1310		1	female	35	0	2	15	S
1311		1	male	35	0	2	15	S
1312								
1313								

titanic for trees B C

statgraphics® centurion
statgraphics.com

Predictions and Residuals

Classification and Regression Trees

1302	12	No	No
1303	7	No	No
1304	7	No	No
1305	4	Yes	No
1306	4	Yes	No
1307	12	No	No
1308	12	No	No
1309	12	No	No

Residual mean deviance = 0.869329

Unknown set n=2

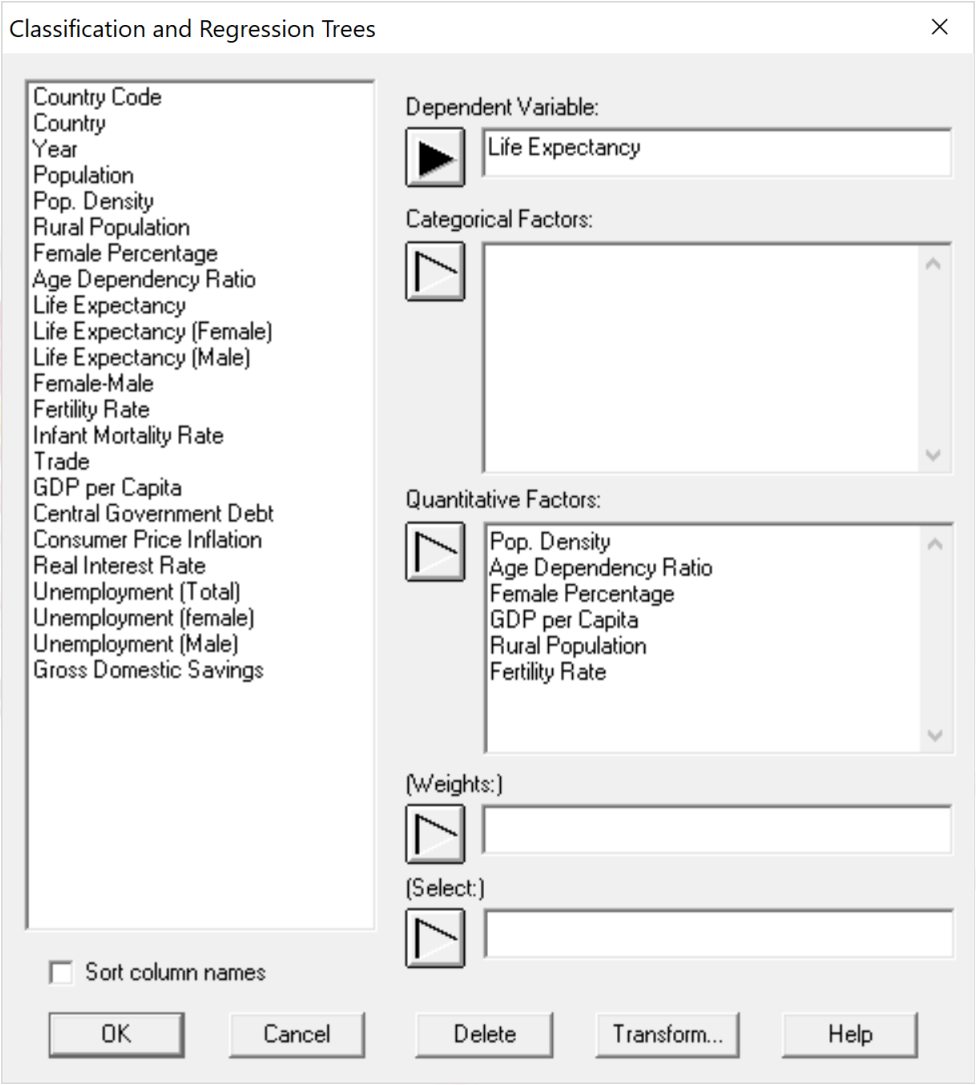
Row	*Leaf*	*Predicted*	*Observed*
1310	6	Yes	
1311	13	No	

The StatAdvisor

This table shows the observed and predicted values of survived for each observation. It also indicates the number of the terminating node which the observations reach after navigating the tree.

statgraphics® centurion
statgraphics.com

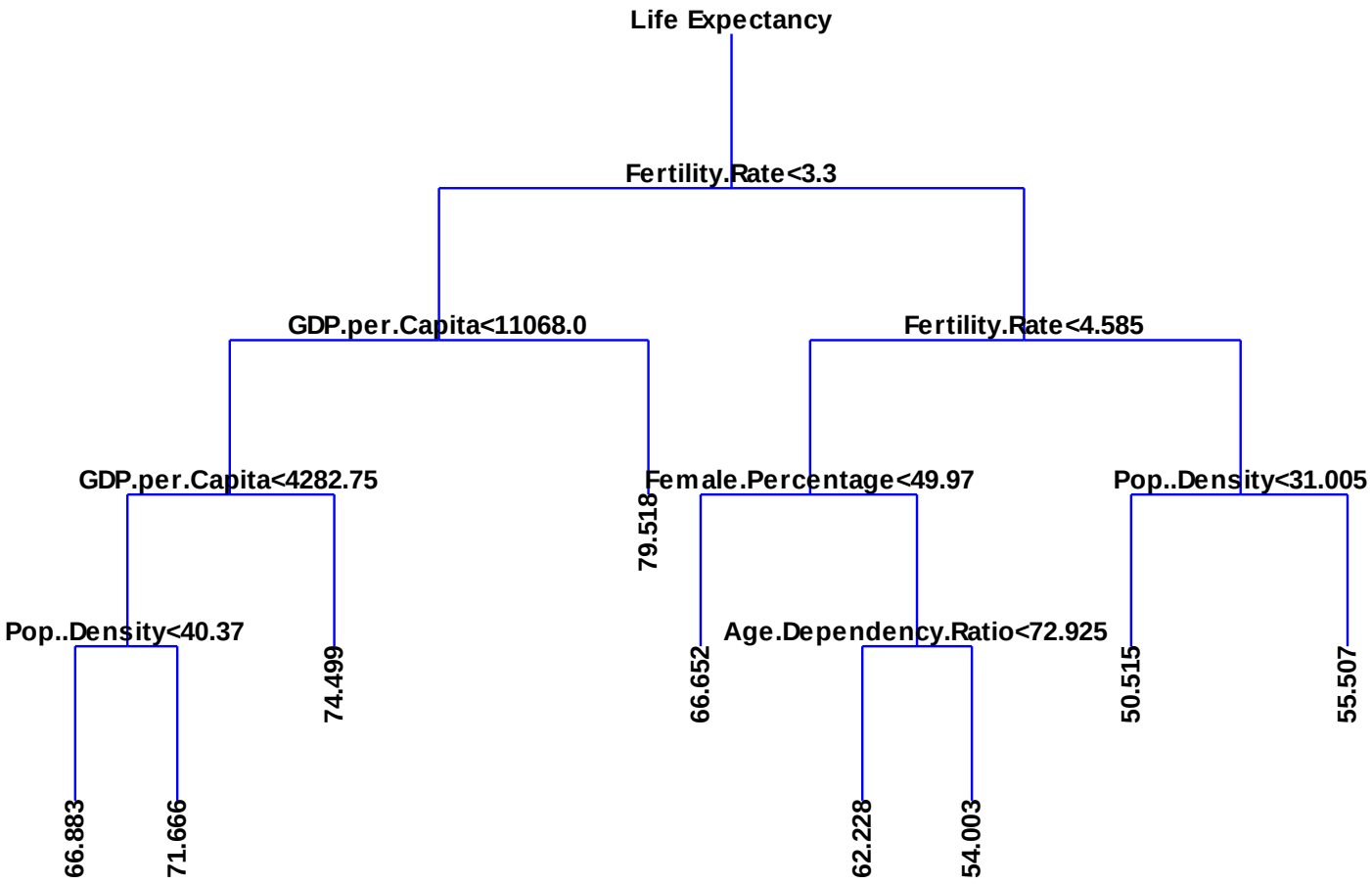
Example 2: World Bank Demographics



statgraphics® centurion

statgraphics.com

Decision Tree



statgraphics®
statgraphics.com
centurion

References

- StatFolios and data files are at: *www.statgraphics.com/webinars*

- R Package "tree" (2015) https://cran.r-project.org/web/packages/tree/tree.pdf

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