Partitioning Data Using JMP® 10 Pro

Kathy Welch
CSCAR, The University of Michigan
Michigan SAS Users Group May 20, 2014

Introduction to Partitioning *

Goal

- Predict an outcome Y based on a set of predictors, Xs
- Good for exploring relationships when you don't have a prior model
- Good for large problems

*Intro Slides adapted from Dave Childers CSCAR workshop on Data Mining

Partitioning: Types of Variables

Y may be continuous or categorical

- Continuous: Regression trees
- Categorical: Classification trees

X may be continuous or categorical

- Continuous, can split between any two values
- Categorical, can make groups based on categories
- Ordinal, better to tell JMP® they are continuous

Partition Methods in JMP®

- **Decision Tree**
  - Classification and Regression trees
- **Bootstrap Forest** (available with JMP® Pro)
- **Boosted Tree** (available with JMP® Pro)

Decision Tree

- Makes a single pass through the data
- Produces a single tree
- Tree can be grown interactively or automatically, if validation is used

Two Types of Decision Trees

Regression Trees

- Outcome is continuous
  - Median house value
  - Wages

Classification Trees

- Outcome is categorical
  - Email vs. Spam
  - Diabetes
  - Type of car chosen

Regression

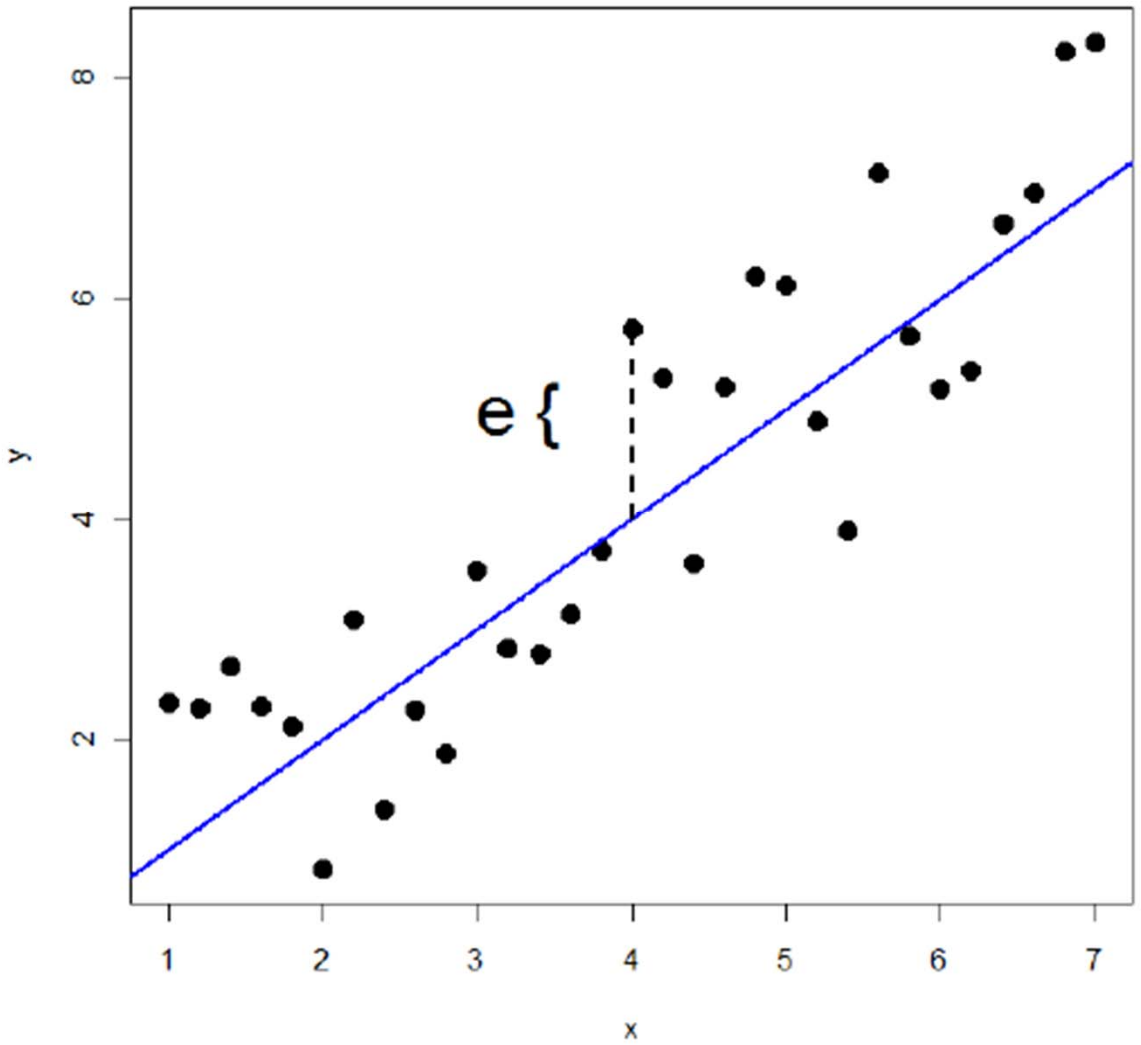
Model

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p + \varepsilon$$

Goals

- Estimate parameters
- Make inferences (confidence intervals and hypothesis tests)
- Use fitted model for prediction

Residuals



Residuals and Least Squares

Residuals

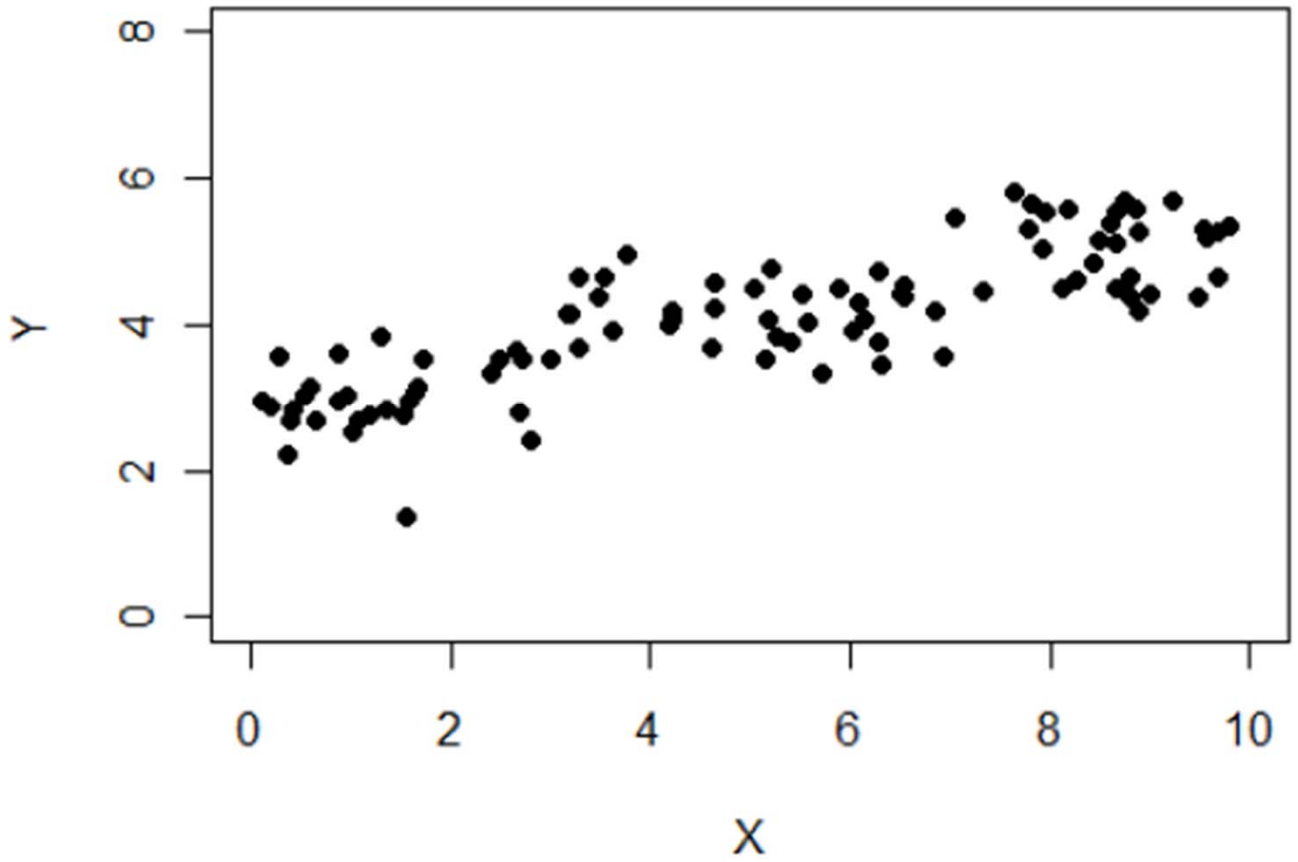
- Definition: the vertical distance between a data point and the line
- Each point has a residual
- The residual for the i^{th} person is denoted e_i

Method of Least Squares

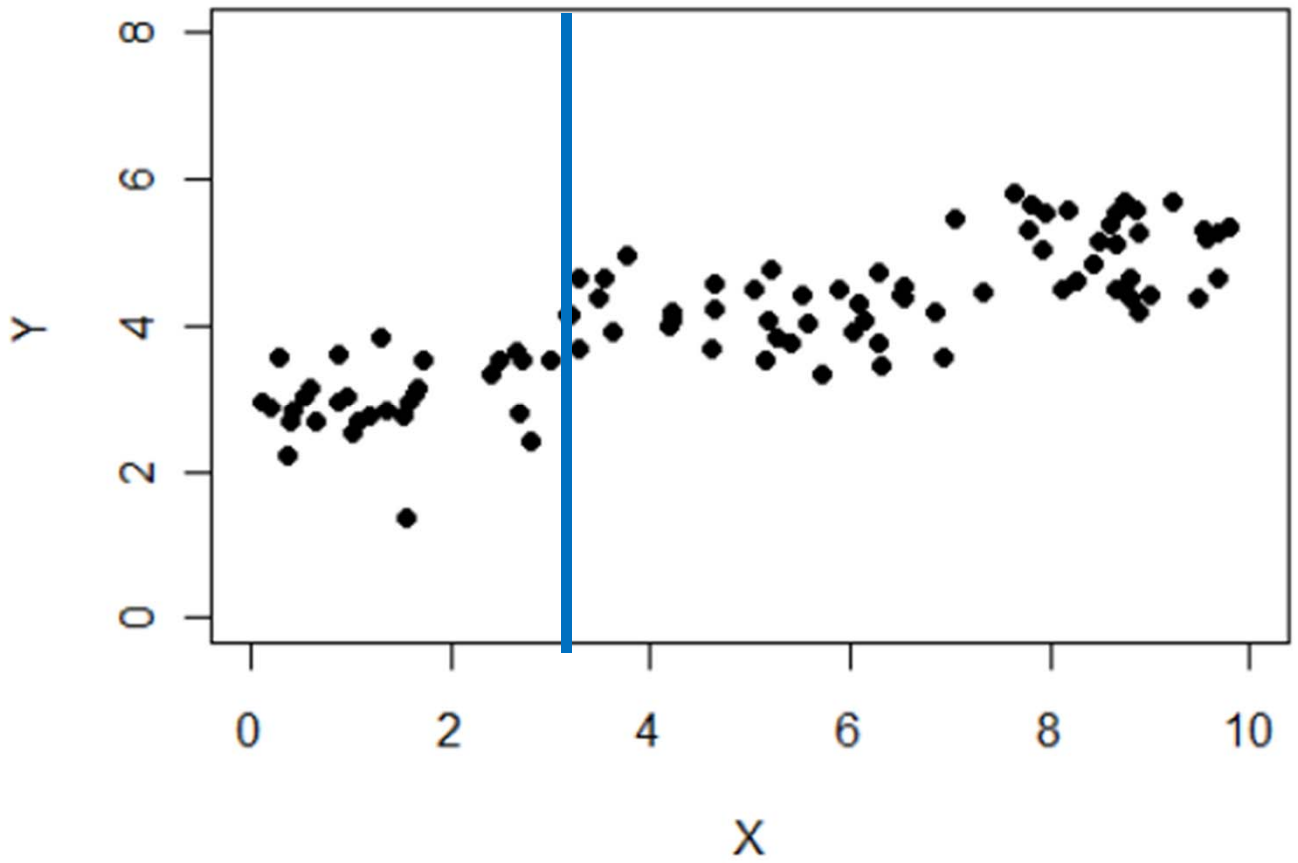
- The least squares regression line is the line that minimizes the sum of the squared residuals (RSS)

$$\sum e_i^2$$

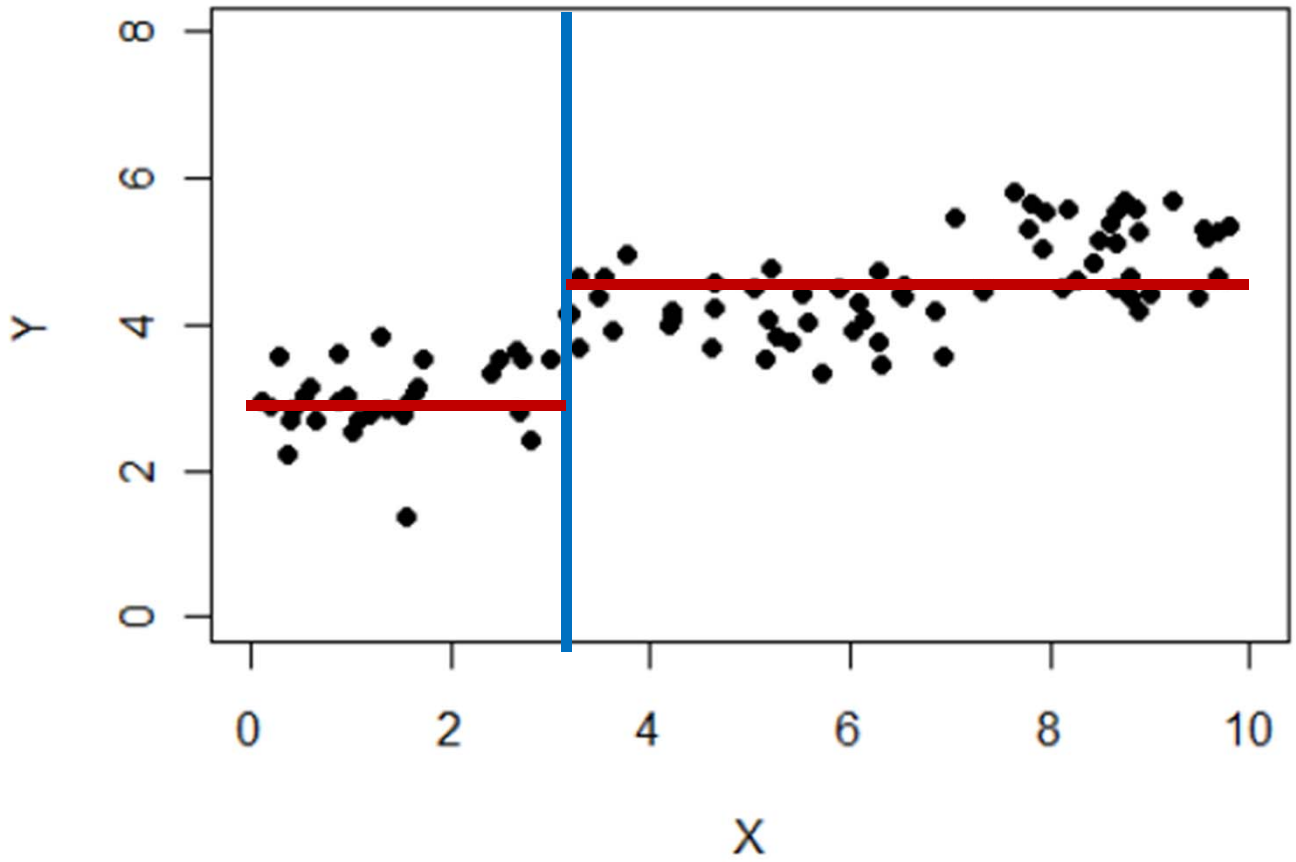
Regression Trees



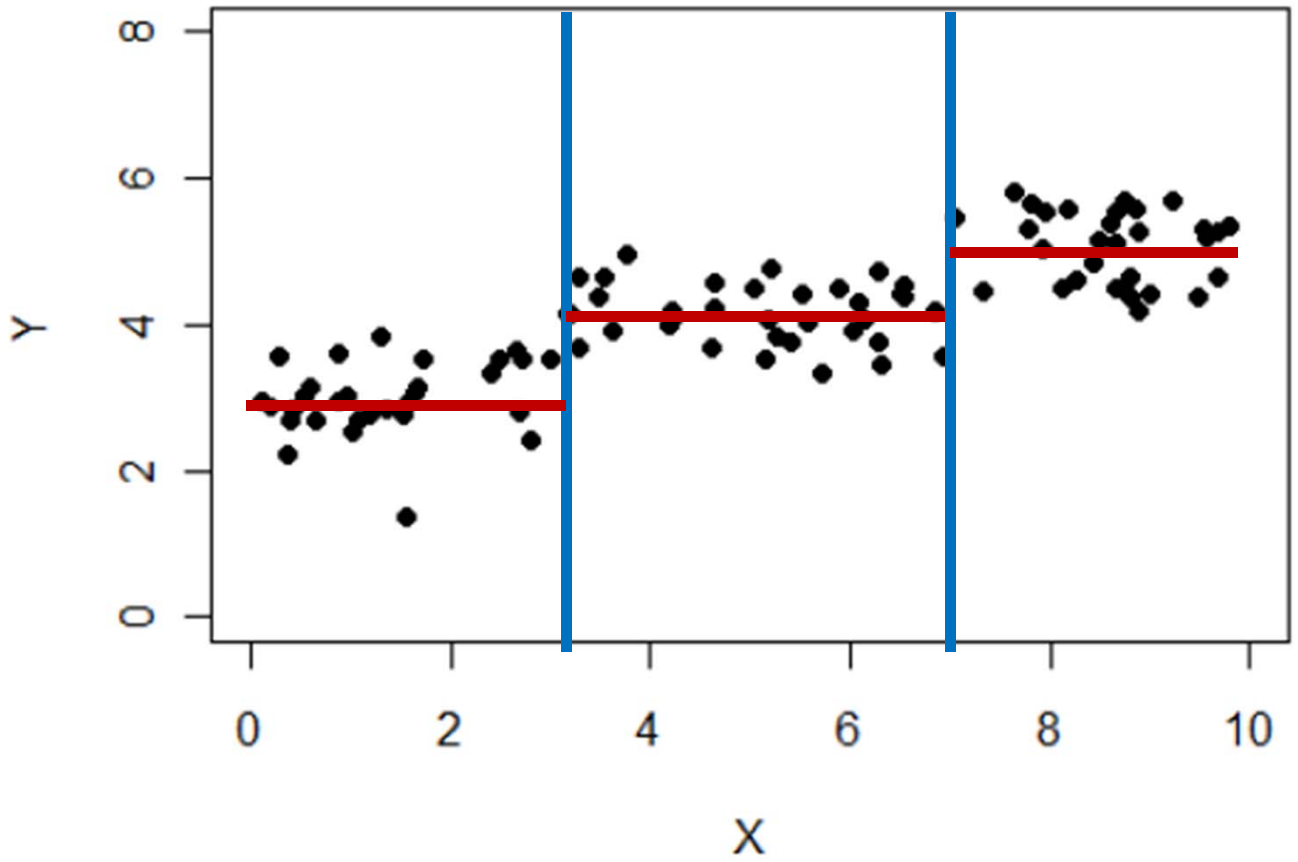
One Split



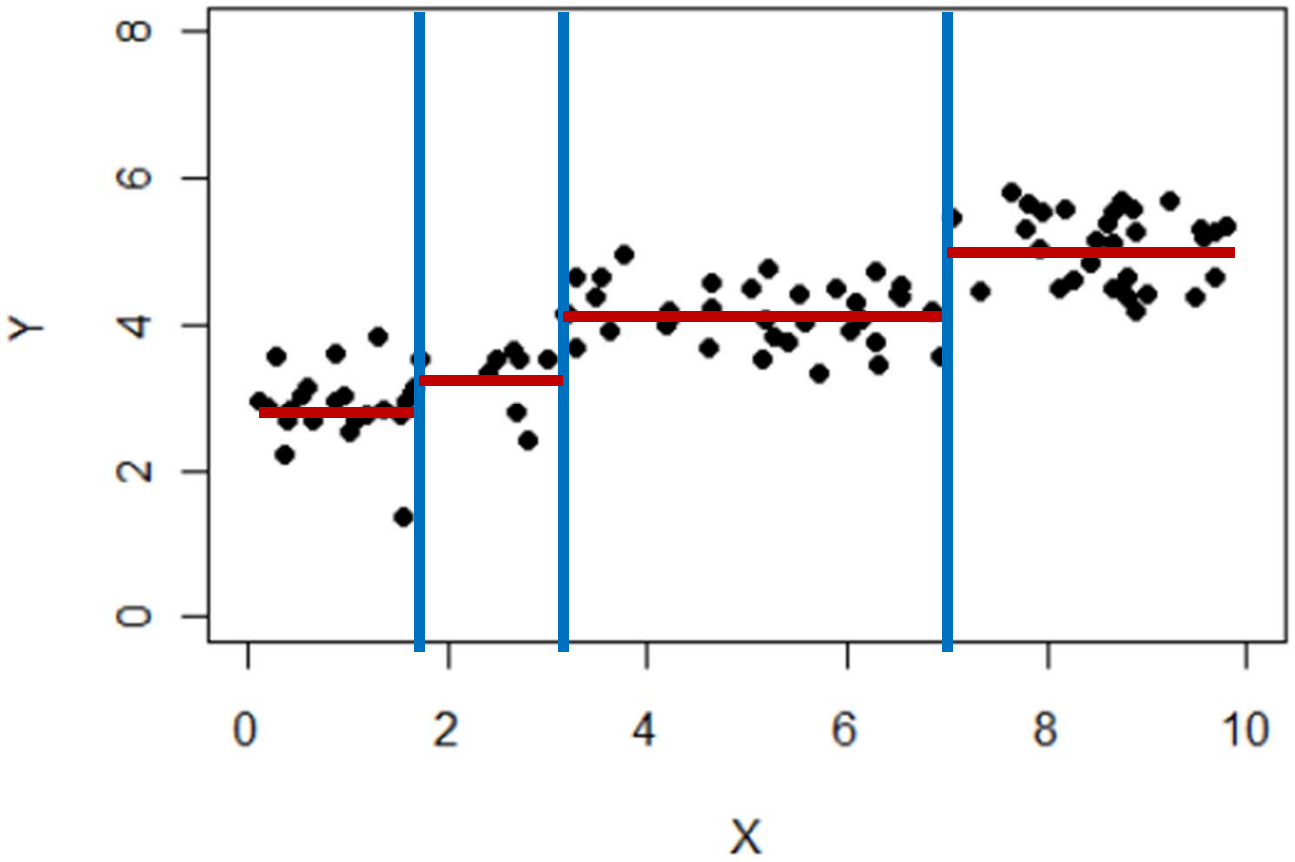
One Split



Two Splits



Three Splits



Decision Tree Algorithm

Steps

- Start with all cases in one group
- Choose "best" cut-point and partition cases into two groups
- Find next best cut-point and split again

Notes

- Splits allowed only within nodes, not across nodes
- Method is called **Recursive Partitioning**

Questions

- How is "best" split chosen?
- What if there are multiple predictors?
- How many splits should be used?
- How useful is the method for prediction?

How is "Best" Split Chosen?

One Method: Use RSS

- Recall for ordinary regression, the "best" fitting line minimizes the residual sum of squares: $\sum e_i^2$

Regression Trees and RSS:

- At each stage of tree growing, choose split that achieves highest reduction in RSS.

Proportion of Variance Explained

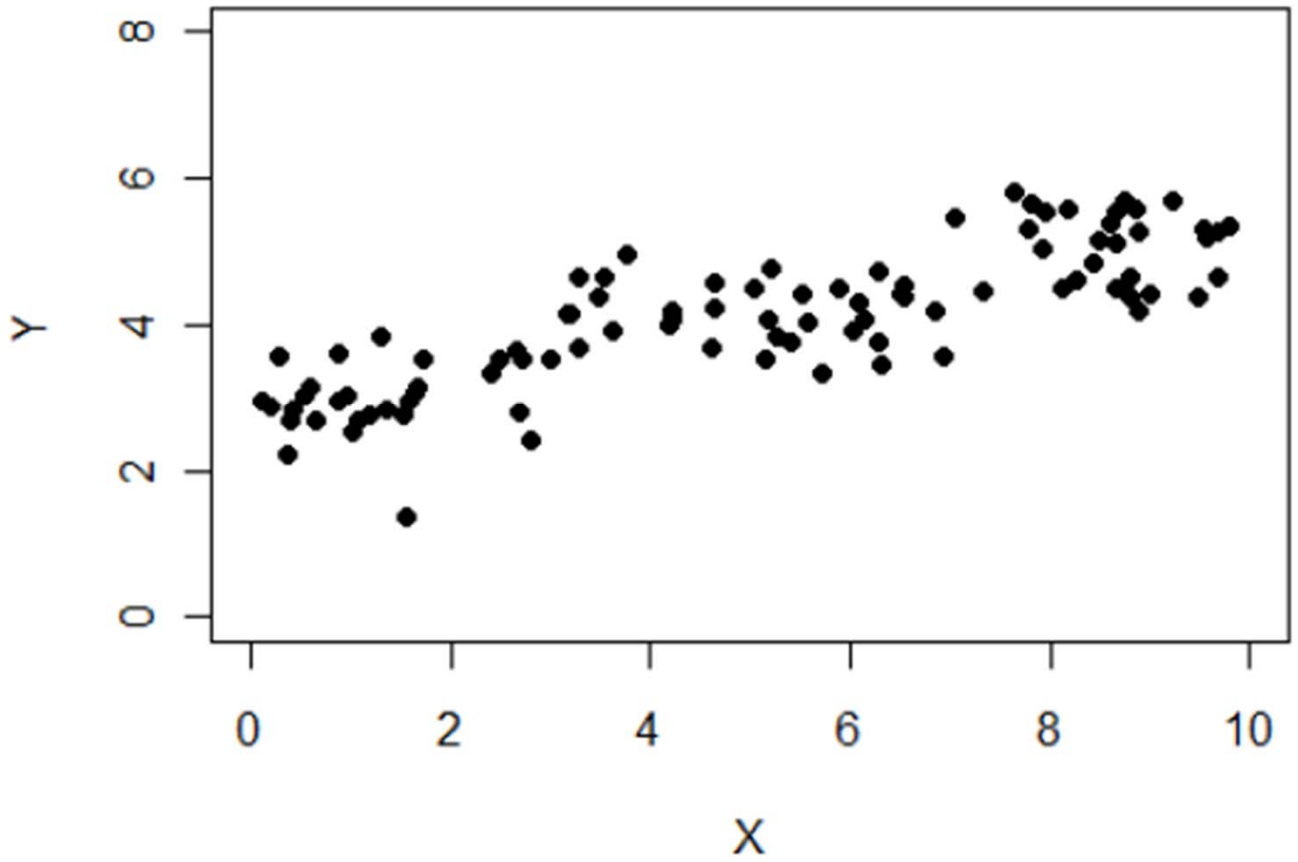
R^2

- R^2 is the proportion of variation in Y explained by the model
- R^2 is between 0 and 1
- Minimizing RSS equivalent to maximizing R^2

$$R^2 = 1 - \frac{\sum (Y_i - \hat{Y}_i)^2}{\sum (Y_i - \bar{Y})^2} = 1 - \frac{\text{RSS}}{\text{Total SS}}$$

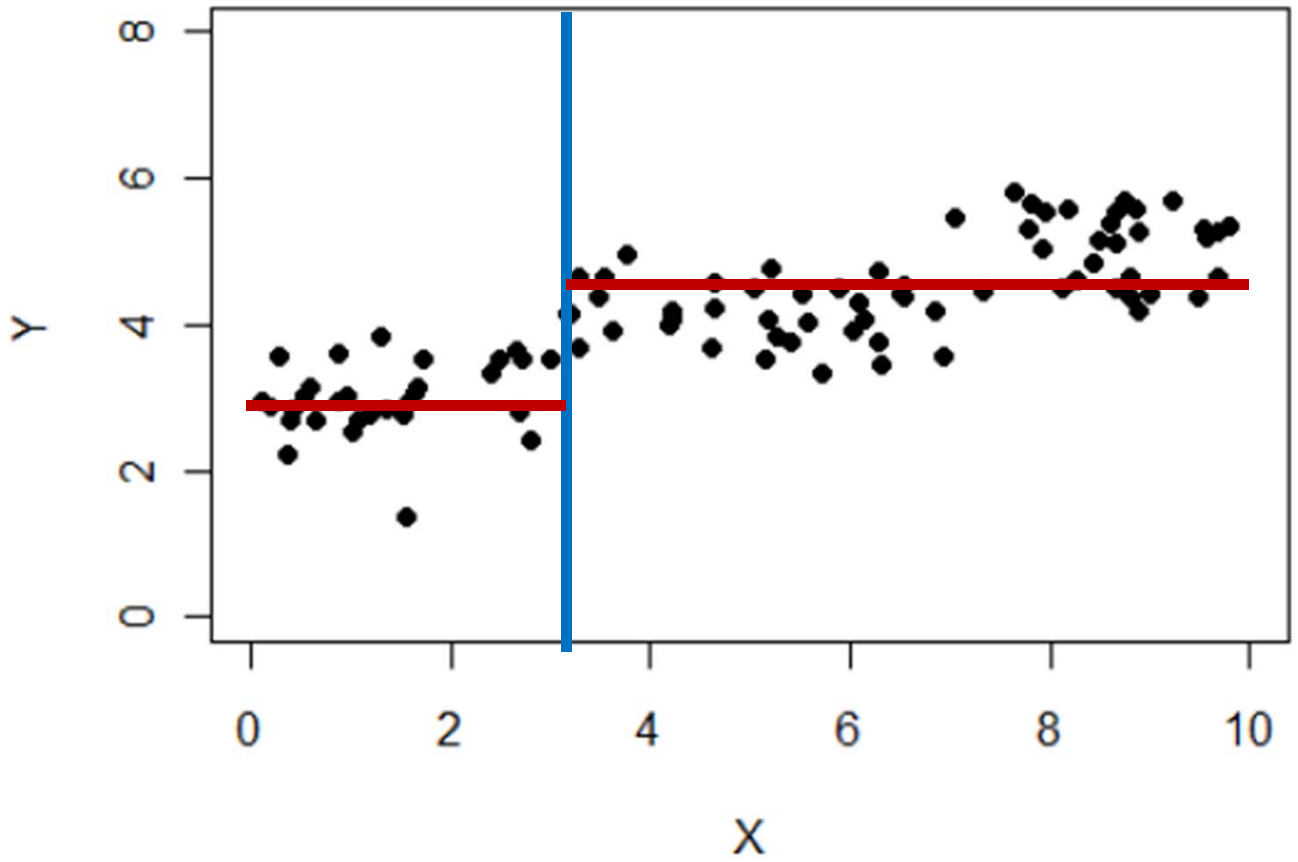
No Splits

RSS = 86.3



One Split

RSS = 34.2



What if There are Multiple Predictors?

Continuous X

- If there are n values there are n-1 possible splits

Categorical X

- If there are k categories there are $2^{k-1} - 1$ possible splits

LogWorth

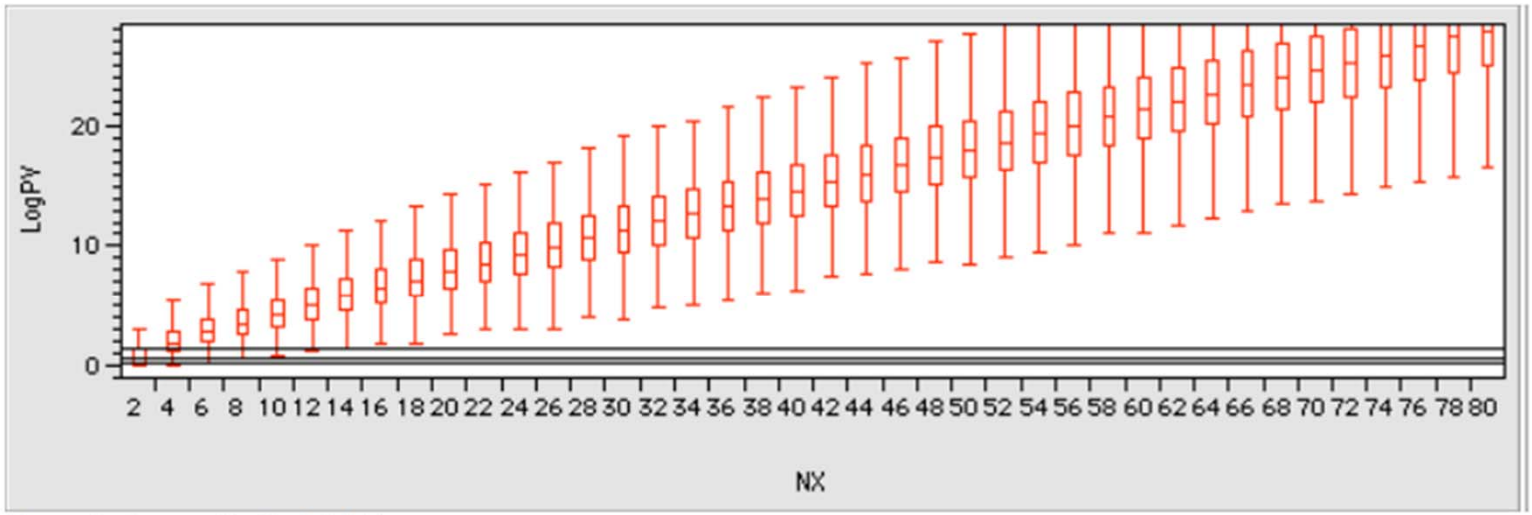
- Tends to weight X variables with different numbers of potential splits equally

LogWorth Criterion

- LogWorth developed by John Sall at JMP®
- $-\log_{10}$(adjusted p-value)
- Higher LogWorth corresponds to smaller adjusted p-value
- Split that produces the highest LogWorth is chosen
- This criterion is used by JMP®

LogWorth (Cont)

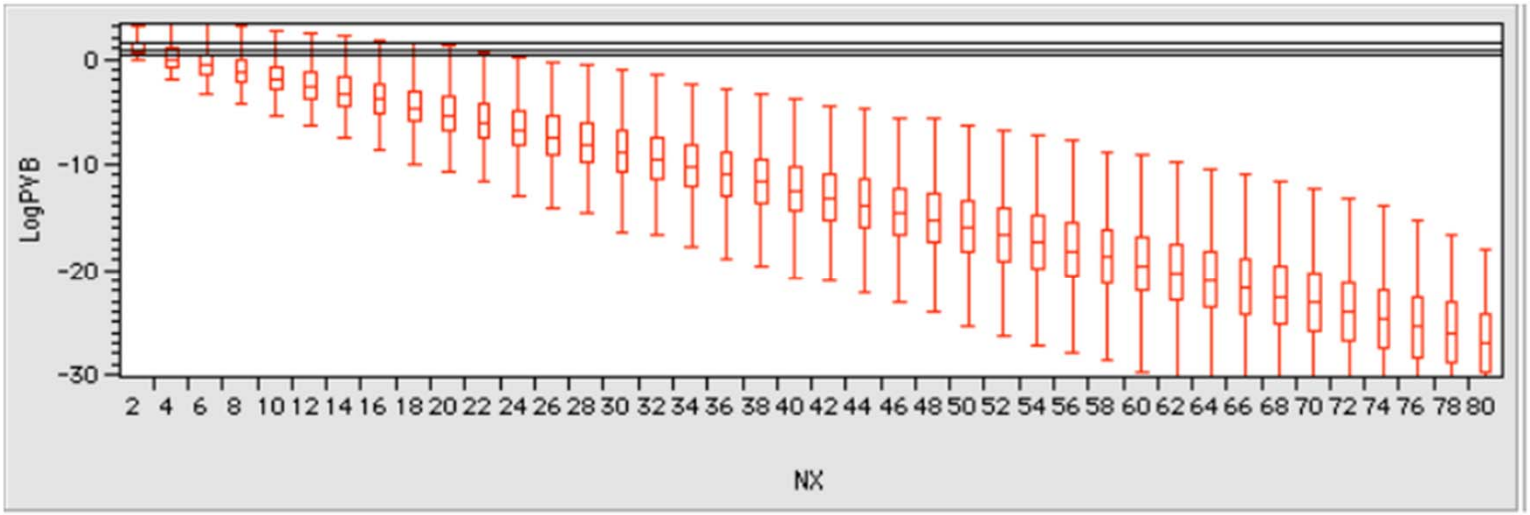
- If we assume the null case, with no relationship between Y and any X, by doing multiple tests we would often choose the X variable with more possible splits



Graph from Sall, 2002.

LogWorth (Cont)

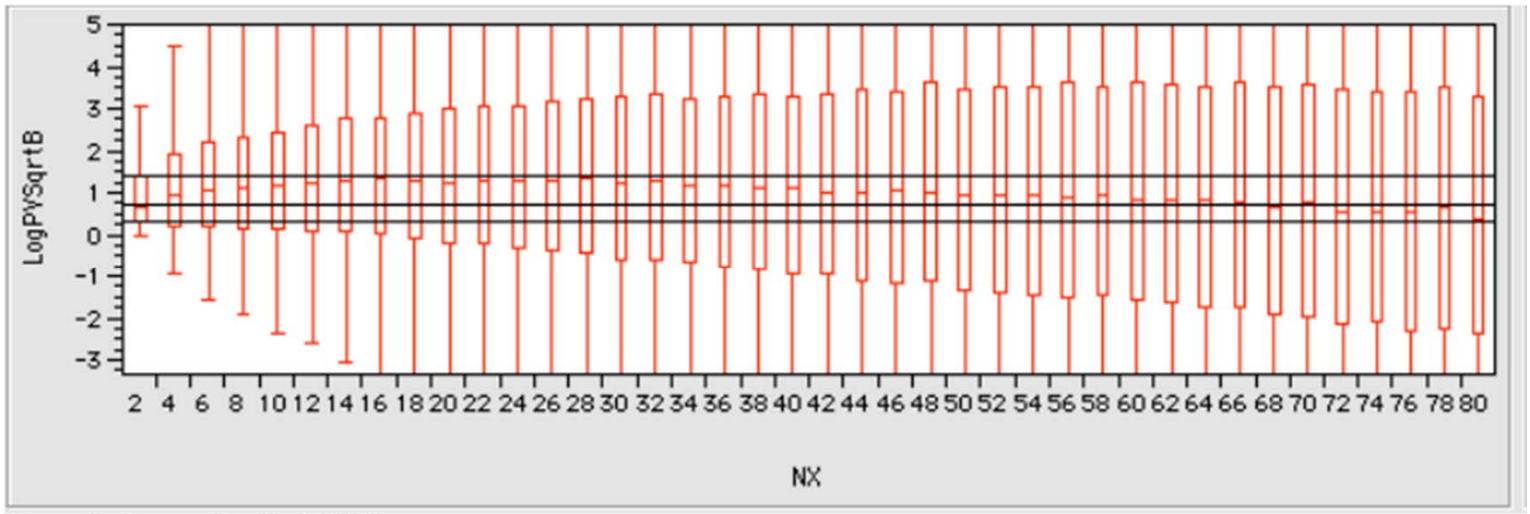
- If we use Bonferroni correction, we over-correct and more often choose the variable with fewer possible splits



Graph from Sall, 2002.

LogWorth (Cont)

- JMP uses LogWorth: the center of the distribution of the –log(adjusted p-value) is not changed as the number of potential splits increases



Graph from Sall, 2002.

Number of Splits

- More complex models will always "fit" the data better
- **Overfitting** occurs when model fits random fluctuations in the data
- Overfit models may perform well on the **training** (original) data, but may perform very poorly on **test** (new) data

Validation

- We want to know how well the tree performs in a different portion of the data not used to grow the tree
- We can set aside a random portion of the data, say 20% or 25%, for validation, and see how the tree grown on the **training** data performs on the **validation** data
- In **cross-validation**, we do this process several times

Cross-Validation

1. Randomly divide cases into (say) 10 groups
2. Leave group 1 out of the model
3. Use groups 2-10 to build a regression tree
4. Evaluate the performance of the tree on group 1
5. Repeat 10 times, leaving out each group once
6. The cross-validated R^2 is the average of the 10 "out-of-sample" R^2 values

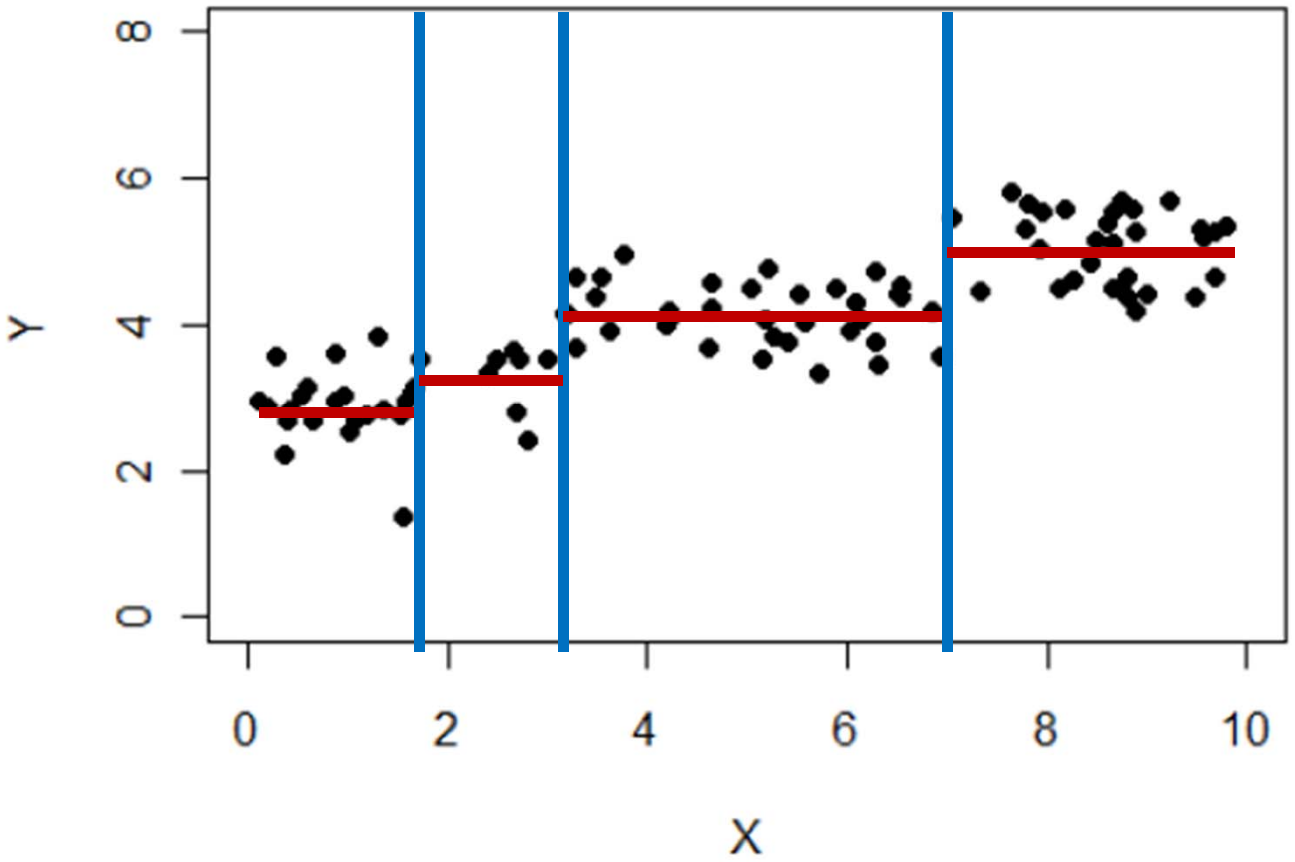
When to Stop

- If no validation method is chosen, the splitting is **interactive**
  - Exploratory method
- If a validation method is chosen, splitting can be **automatic**
  - Splitting can be set to stop if validation R-Square is better than what the next 10 splits would produce or minimum size split is reached
  - May produce complex trees, not very interpretable, but good at prediction

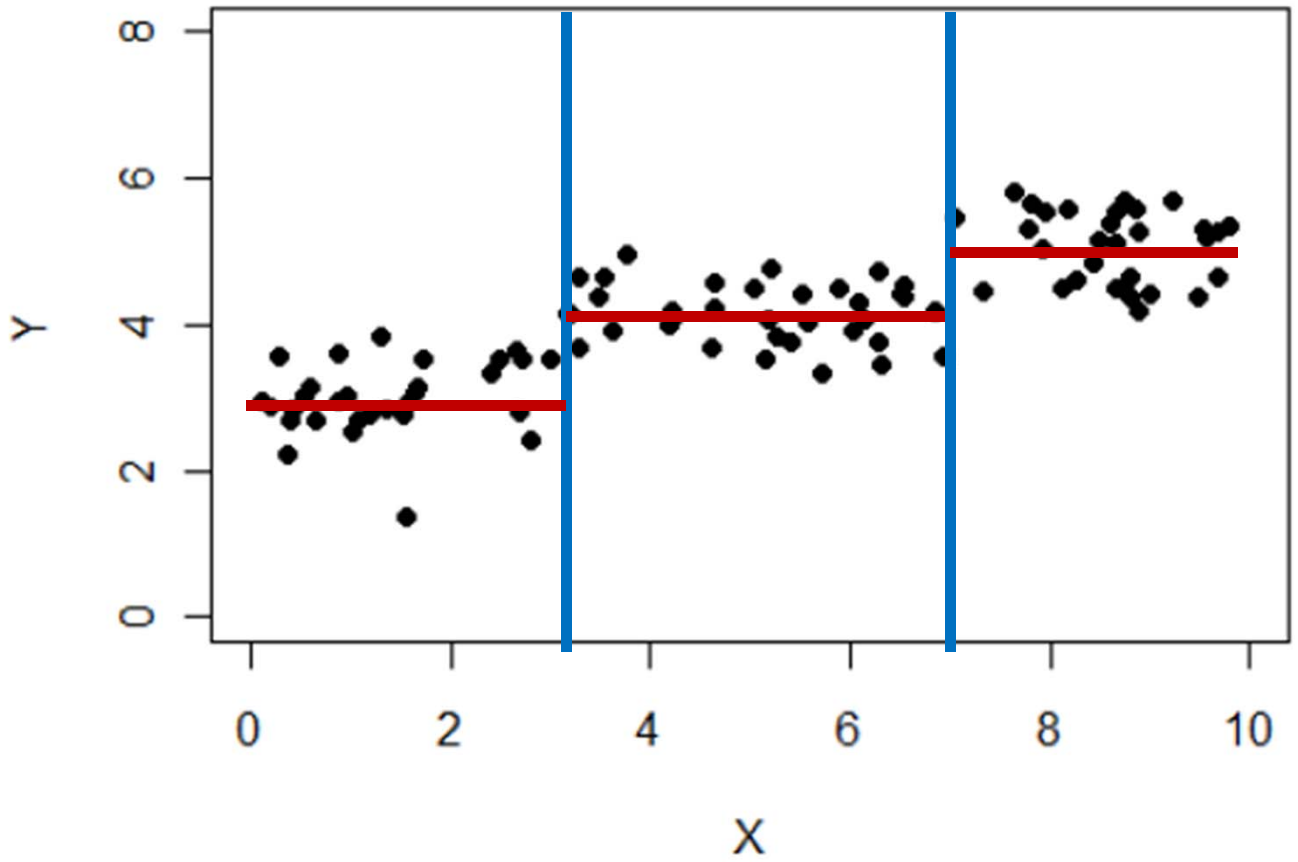
Pruning

- Tree pruning reduces complexity by combining terminal nodes

Before Pruning



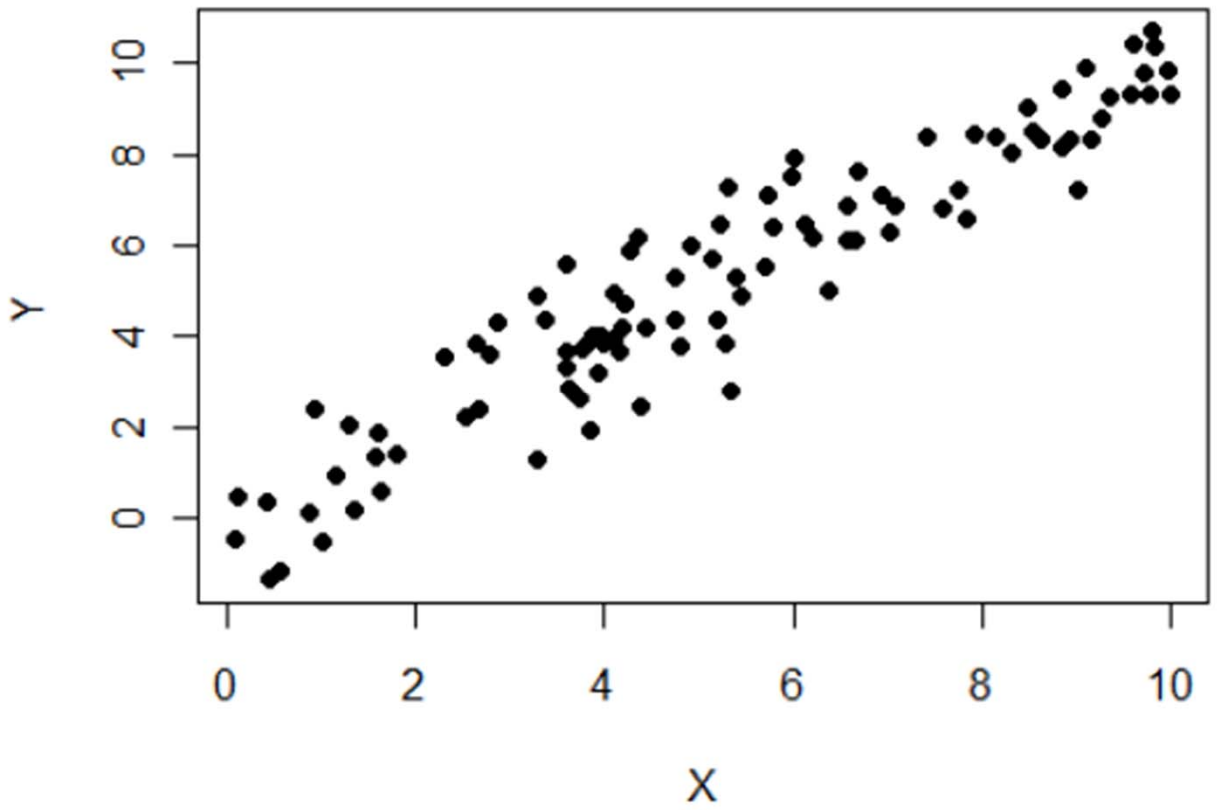
After Pruning



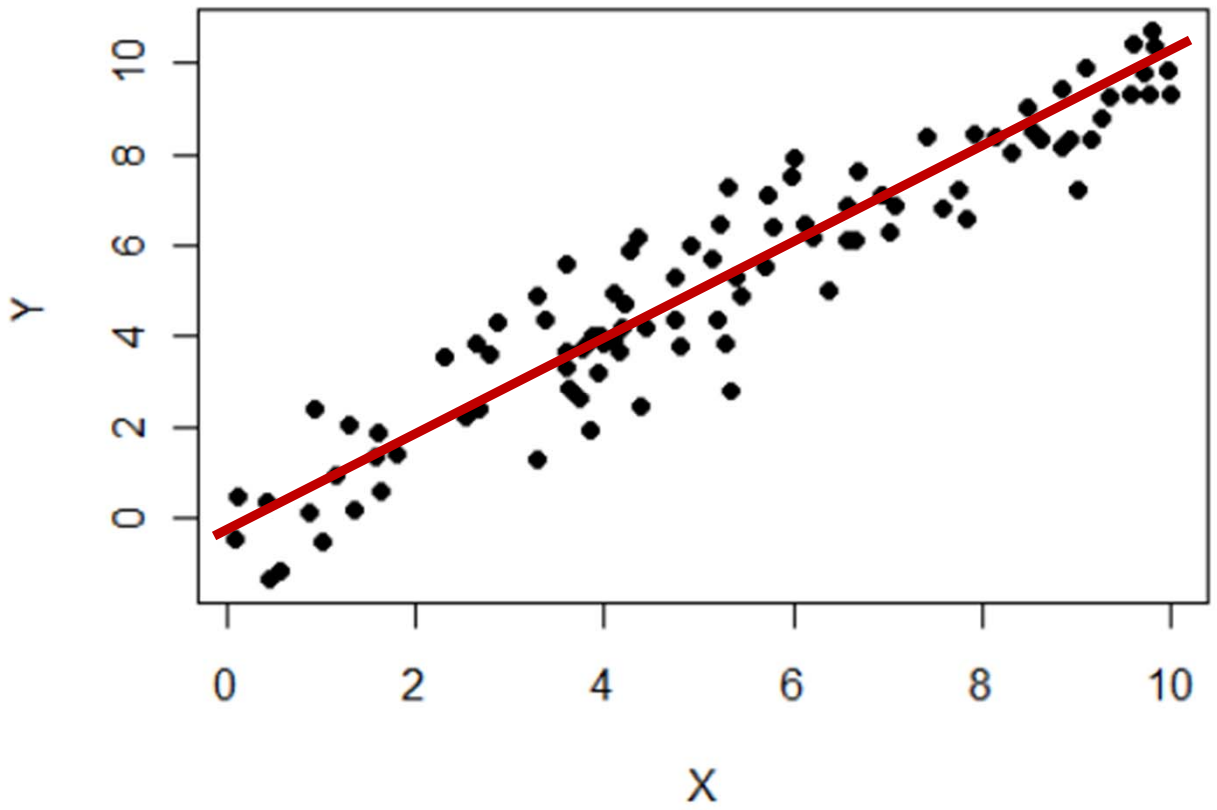
Linear Regression vs. Decision Tree

- When does it help to grow a tree, rather than using linear regression?

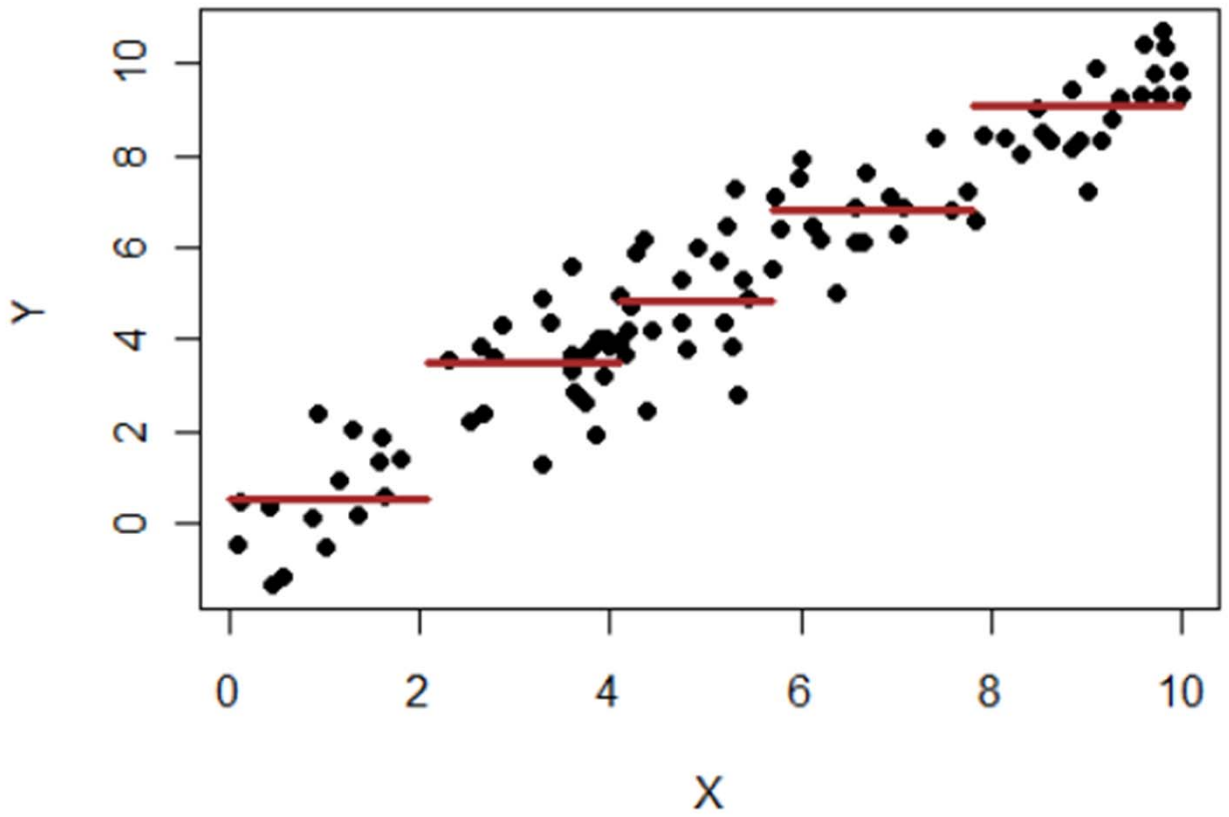
Scatterplot 1



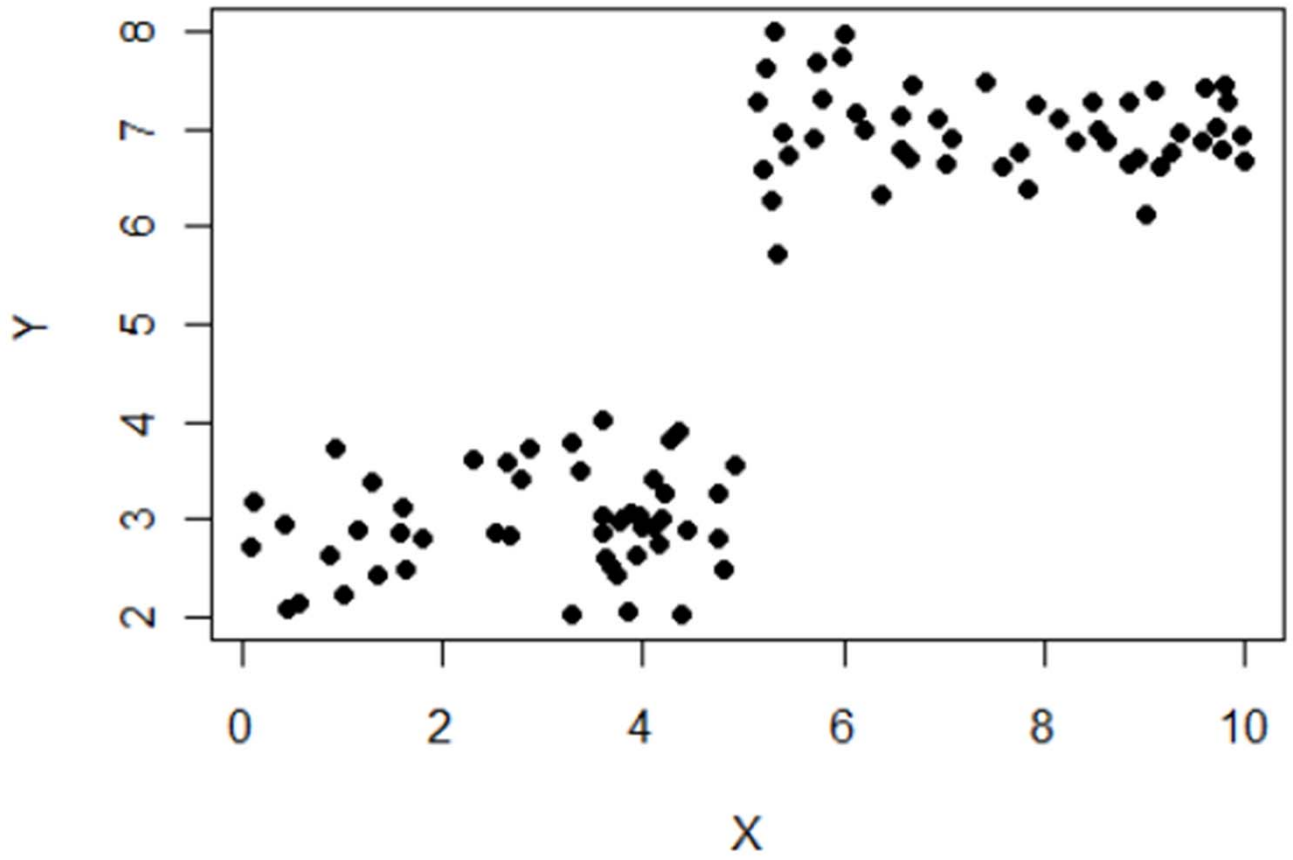
Simple Linear Regression



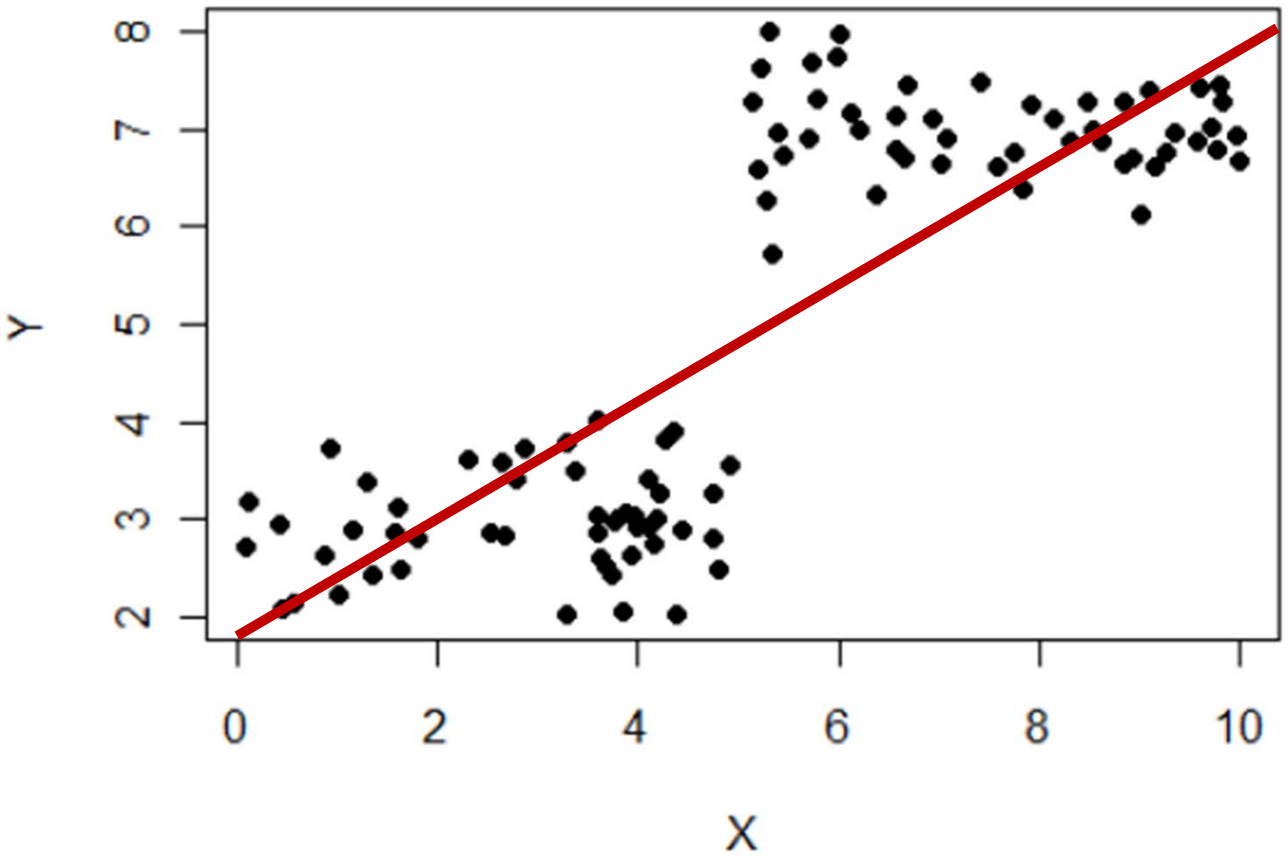
Regression Tree



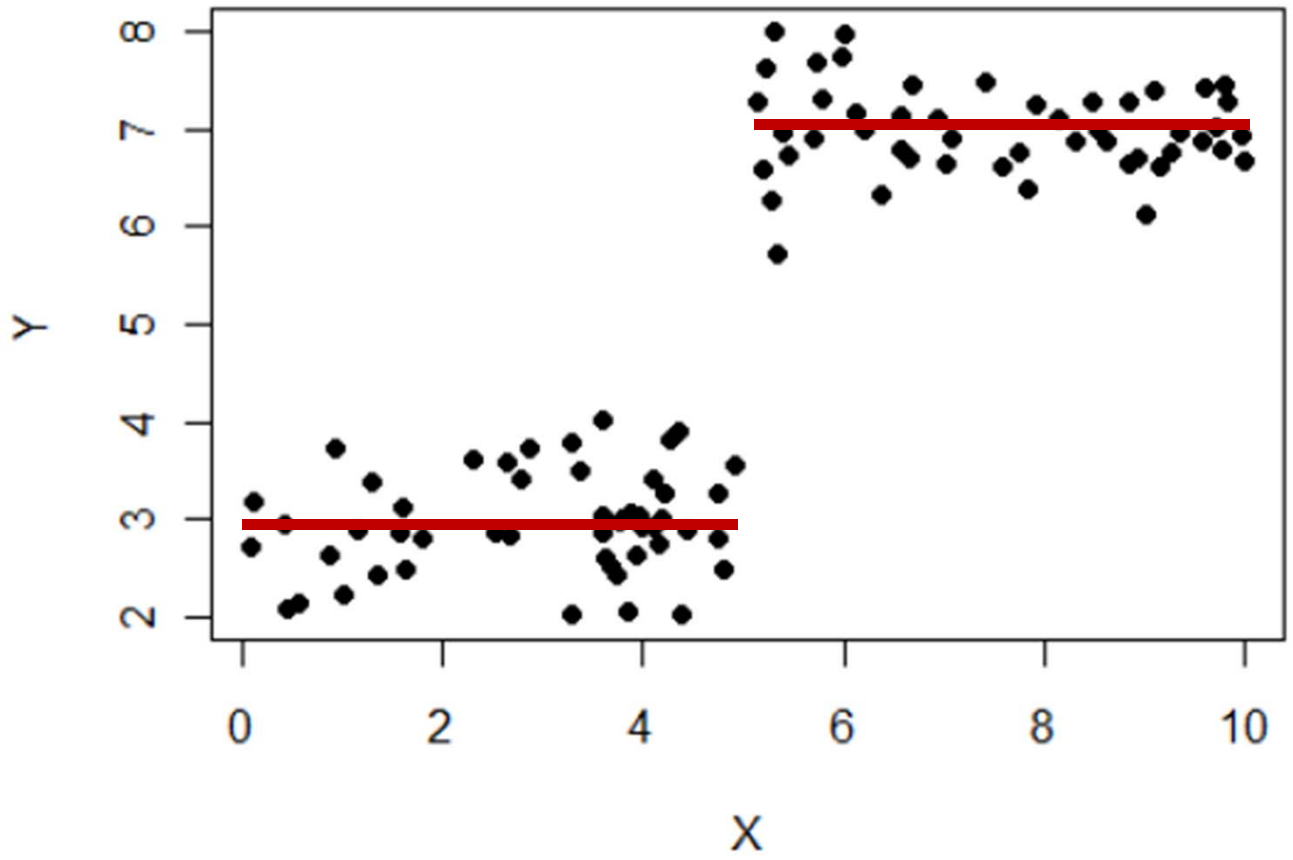
Scatterplot 2



Linear Regression



Regression Tree



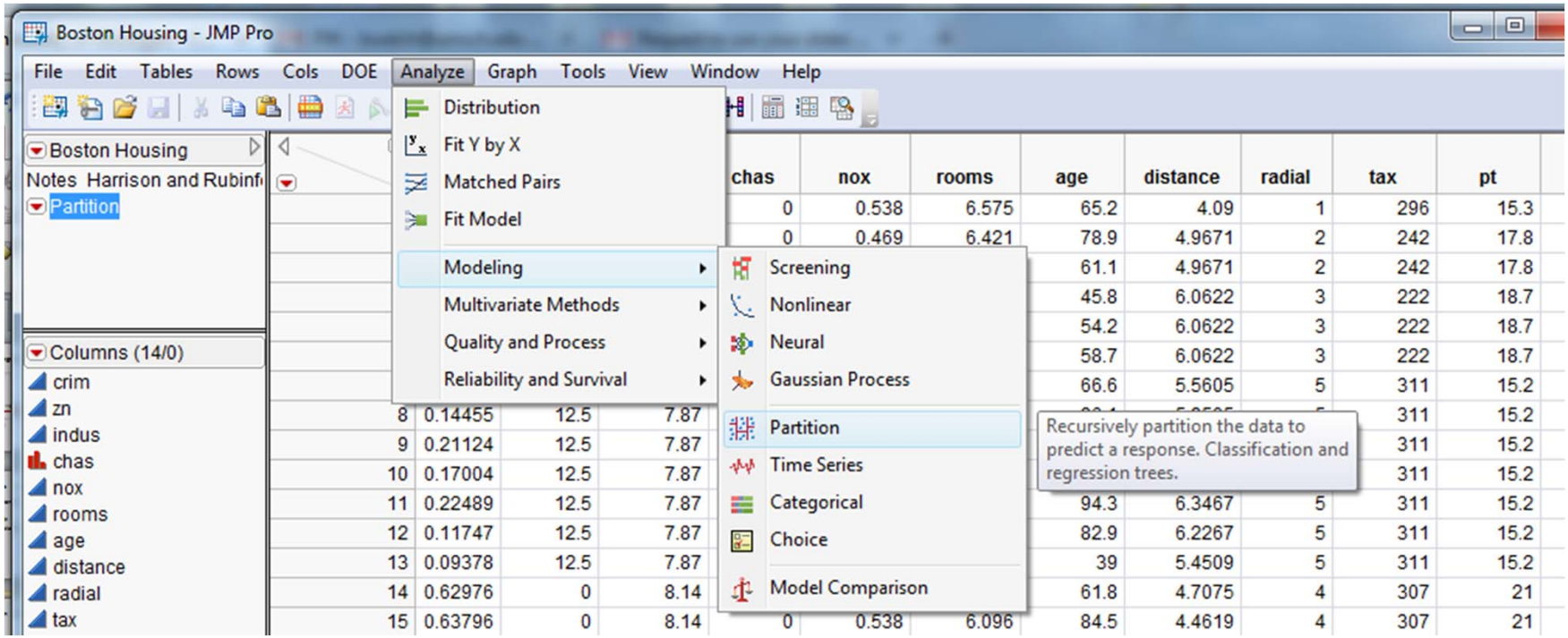
Advantages of Regression Trees

- Detection of non-linearities, change-points
- Interactions
- Non-parametric
- Family of solutions

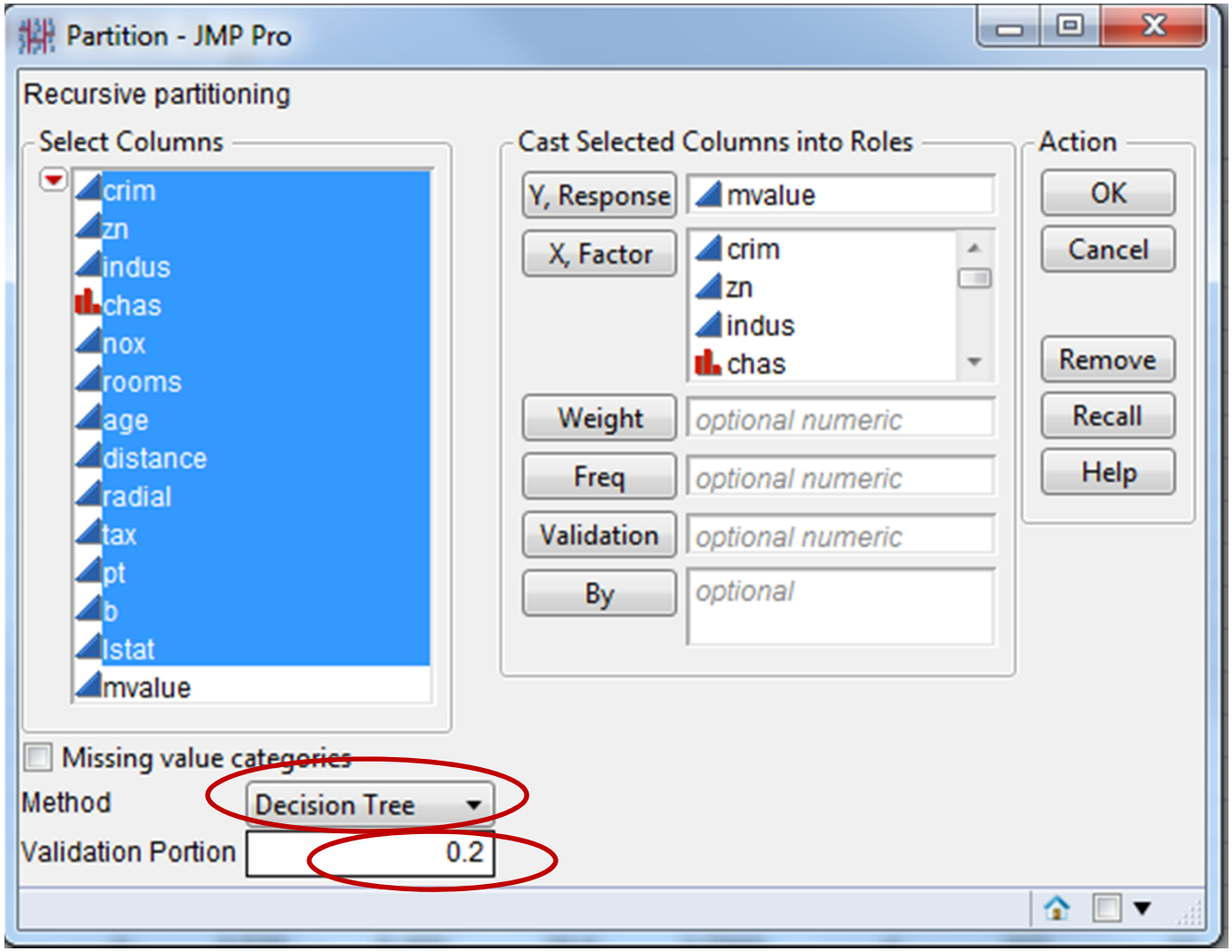
Regression Tree Setup for JMP®

Use **`Boston Housing.jmp`** data set

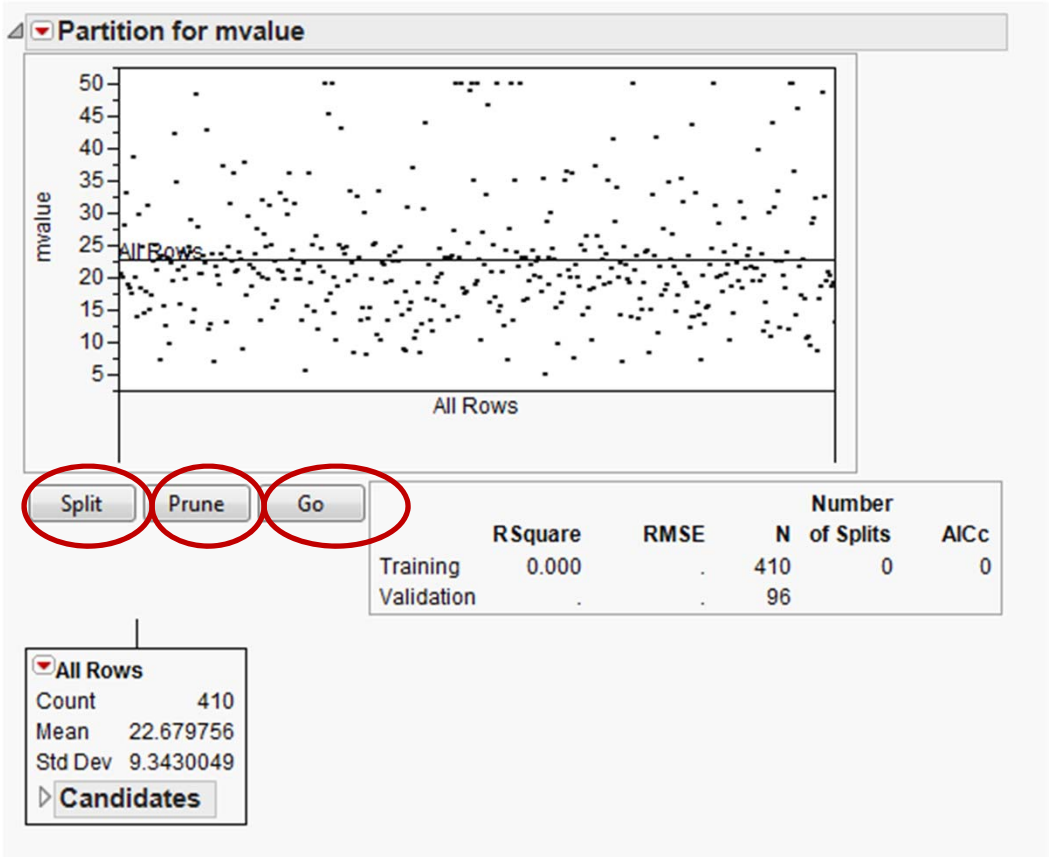
Select **Analyze > Modeling > Partition**



Regression Tree Launch Window



Decision Tree Initial Report



- **Split**-interactive
- **Prune**-interactive
- **Go**-automatic (available when using validation)

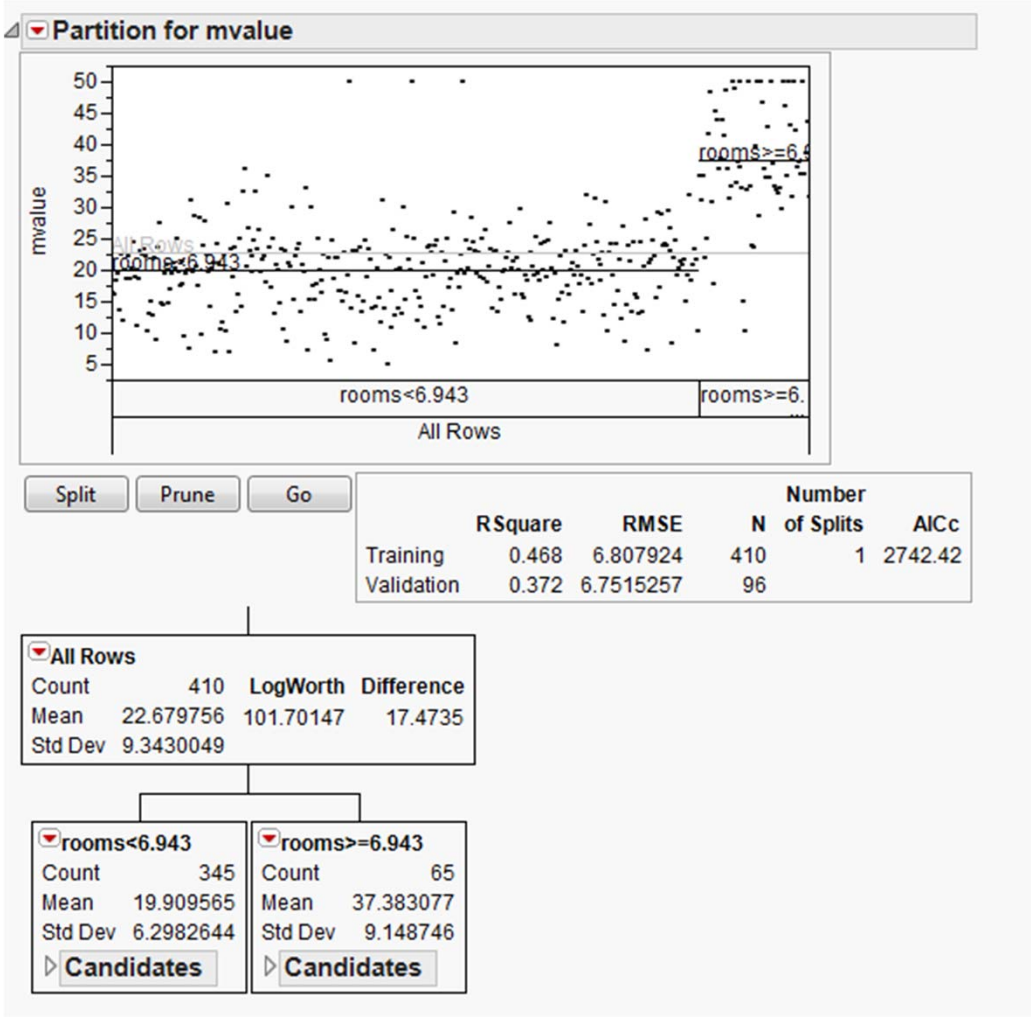
Split Candidate Information

- **LogWorth**: for each variable, the LogWorth if that variable is used for the next split
- **SS**: the difference in SS that would result from splitting at best split for that variable

$$SS_{test} = SS_{parent} - (SS_{right} + SS_{left})$$

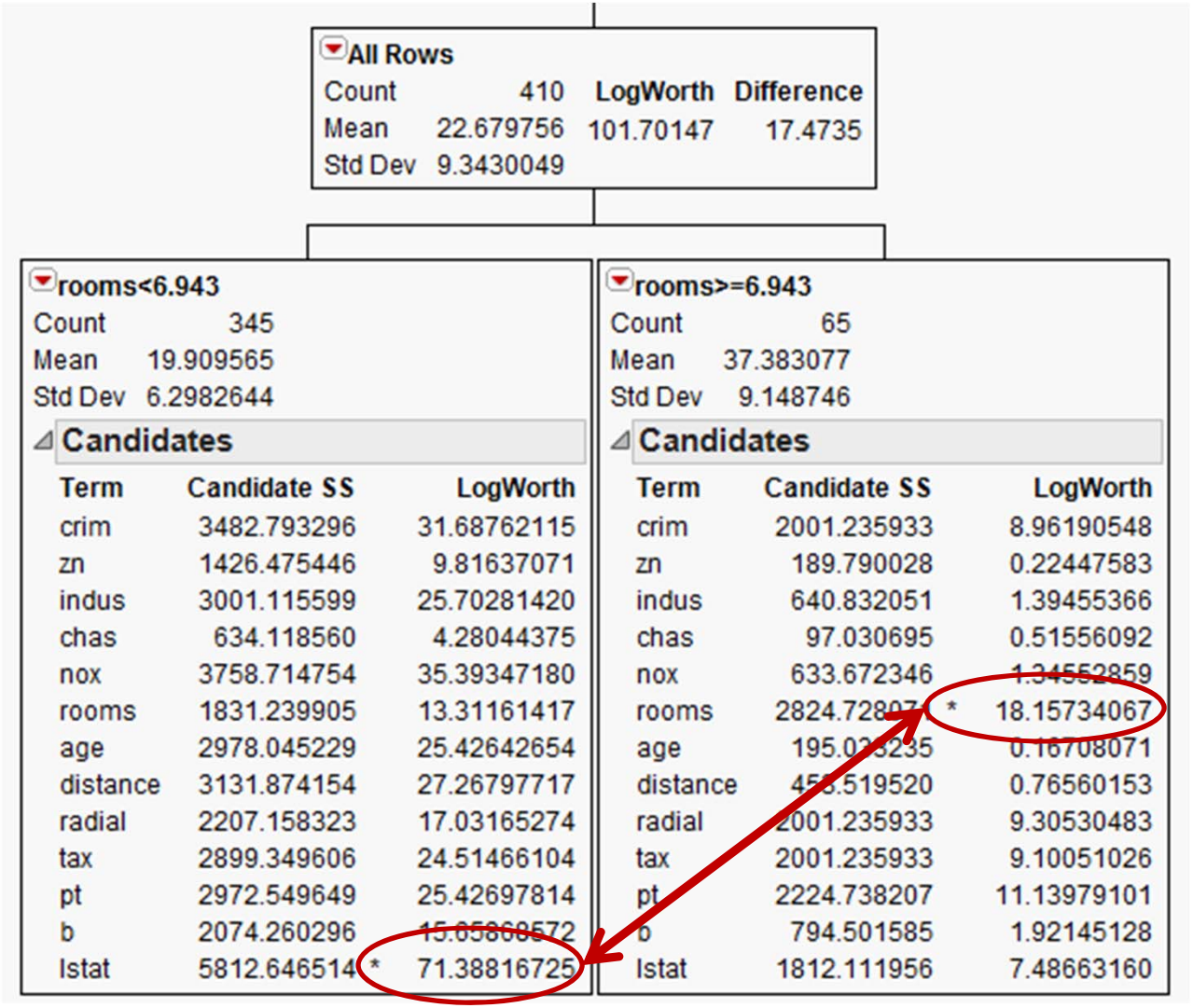
$$\text{where } SS = s^2(n\text{-}1)$$

After One Split

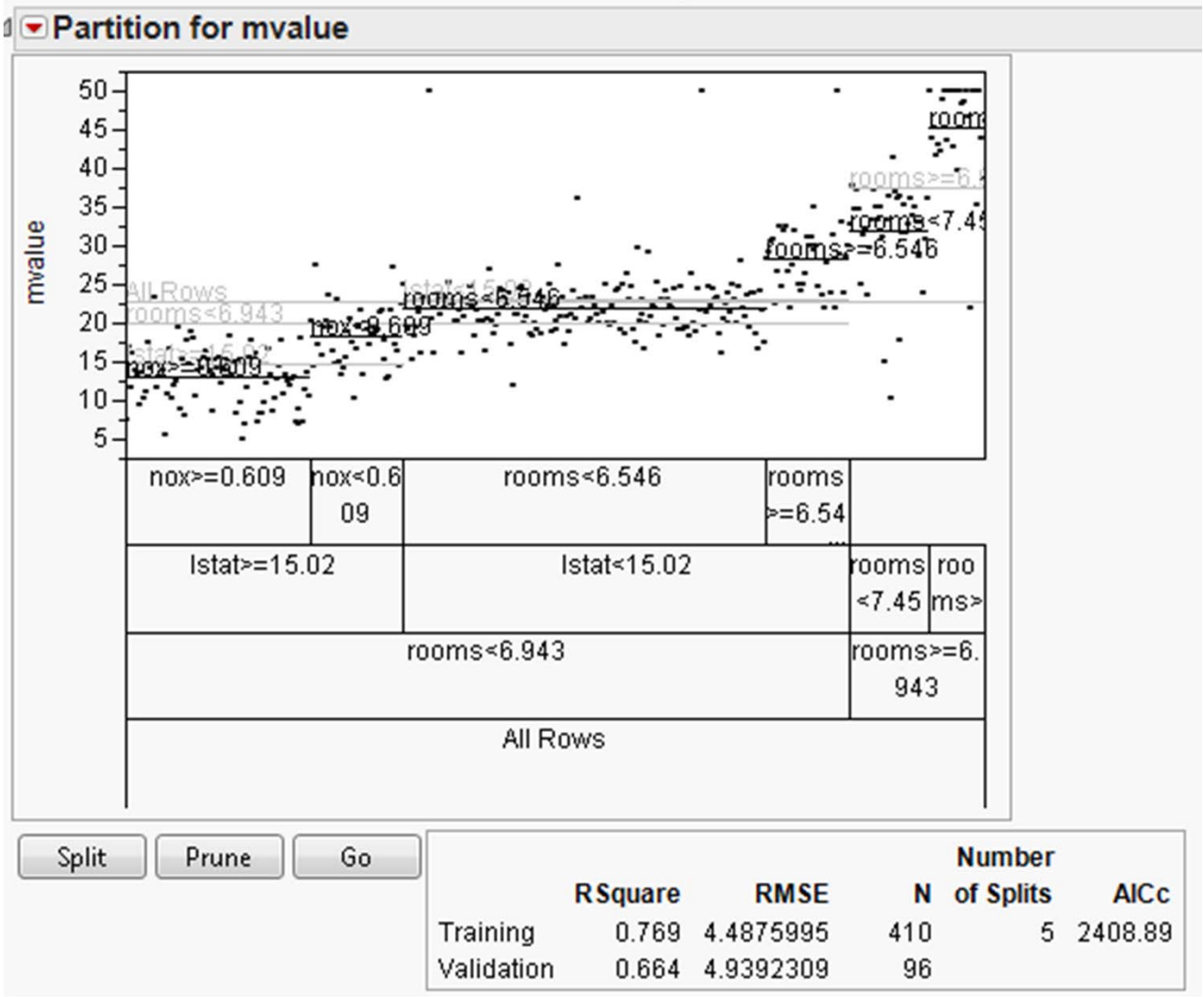


- **R-square** is shown for training and validation data
- **Mean** and **Std Dev**, standard deviation, are shown for each node
- Mean <6.943 rooms=19.9
- Mean ≥6.943 rooms=37.4

Split Candidates for Second Split



After 5 Splits



	RSquare	RMSE	N	Number of Splits	AICc
Training	0.769	4.4875995	410	5	2408.89
Validation	0.664	4.9392309	96		

Node Reports

For All Nodes

- Count (number of cases) in node
- Mean and standard deviation

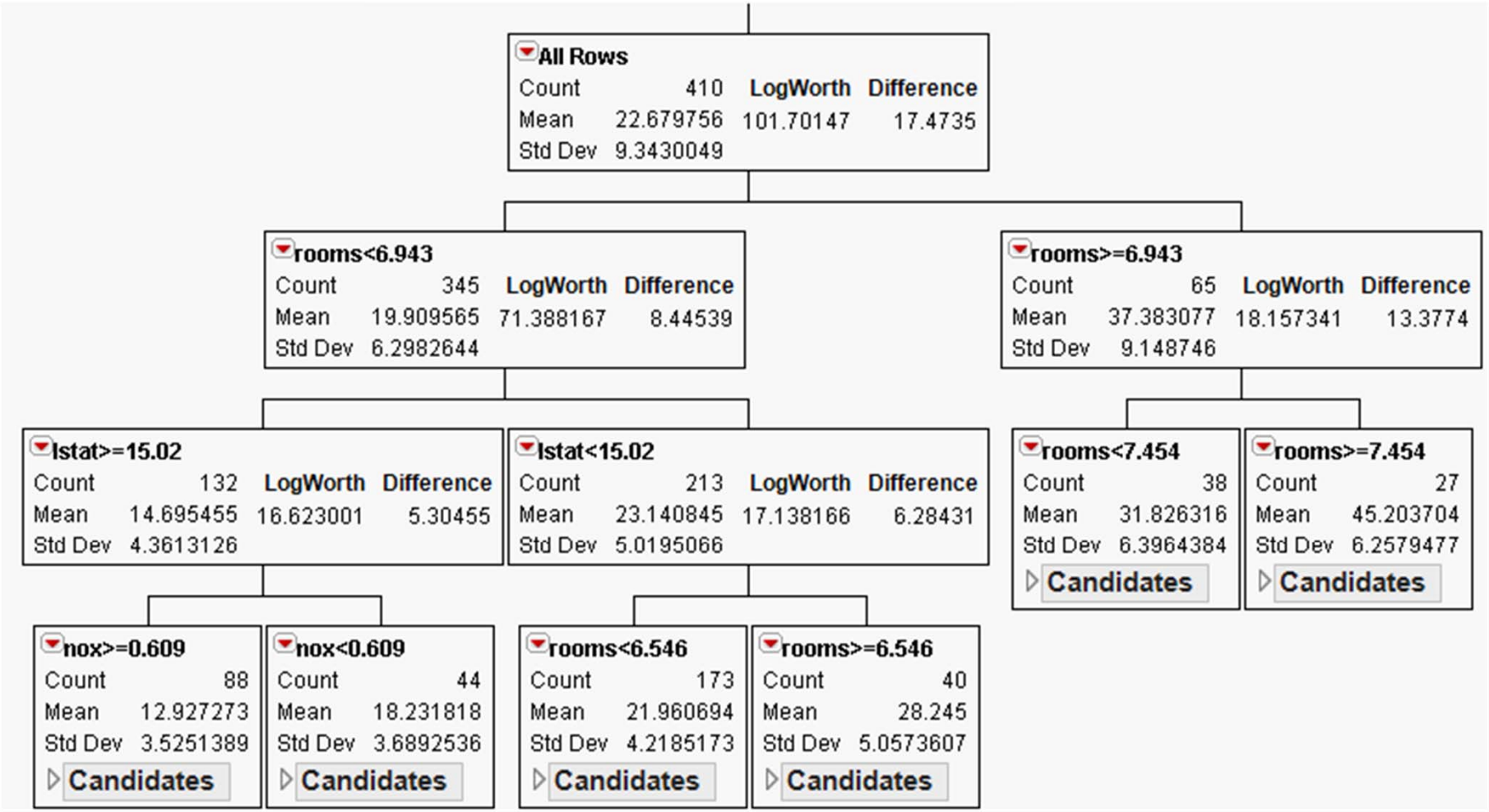
For Parent Nodes

- Difference in mean of Y for left vs. right node
- LogWorth: LogWorth for the split

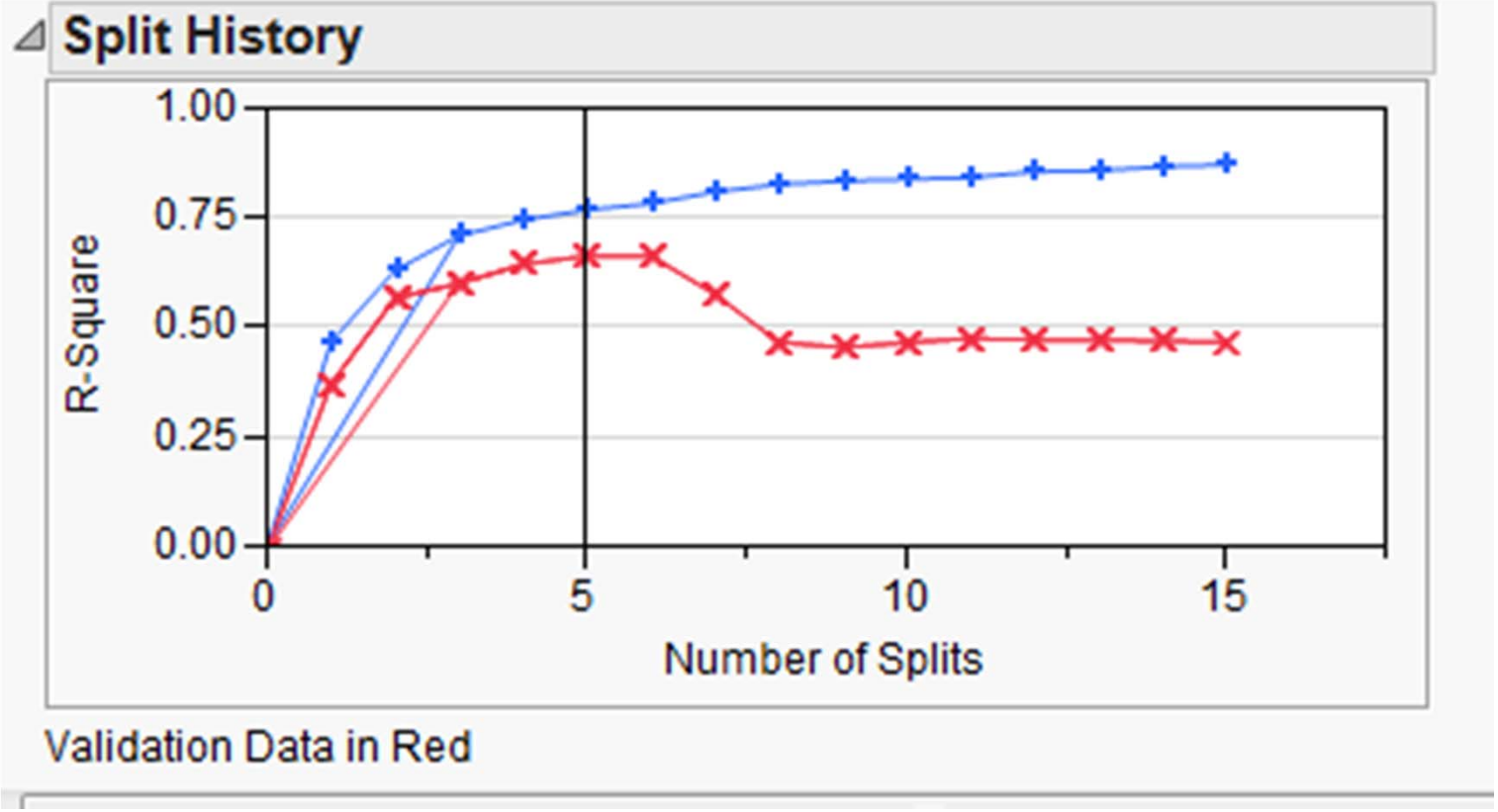
For Terminal Nodes

- Candidates for possible splitting

Tree After 5 Splits



Split History



Classification Trees

Y is categorical

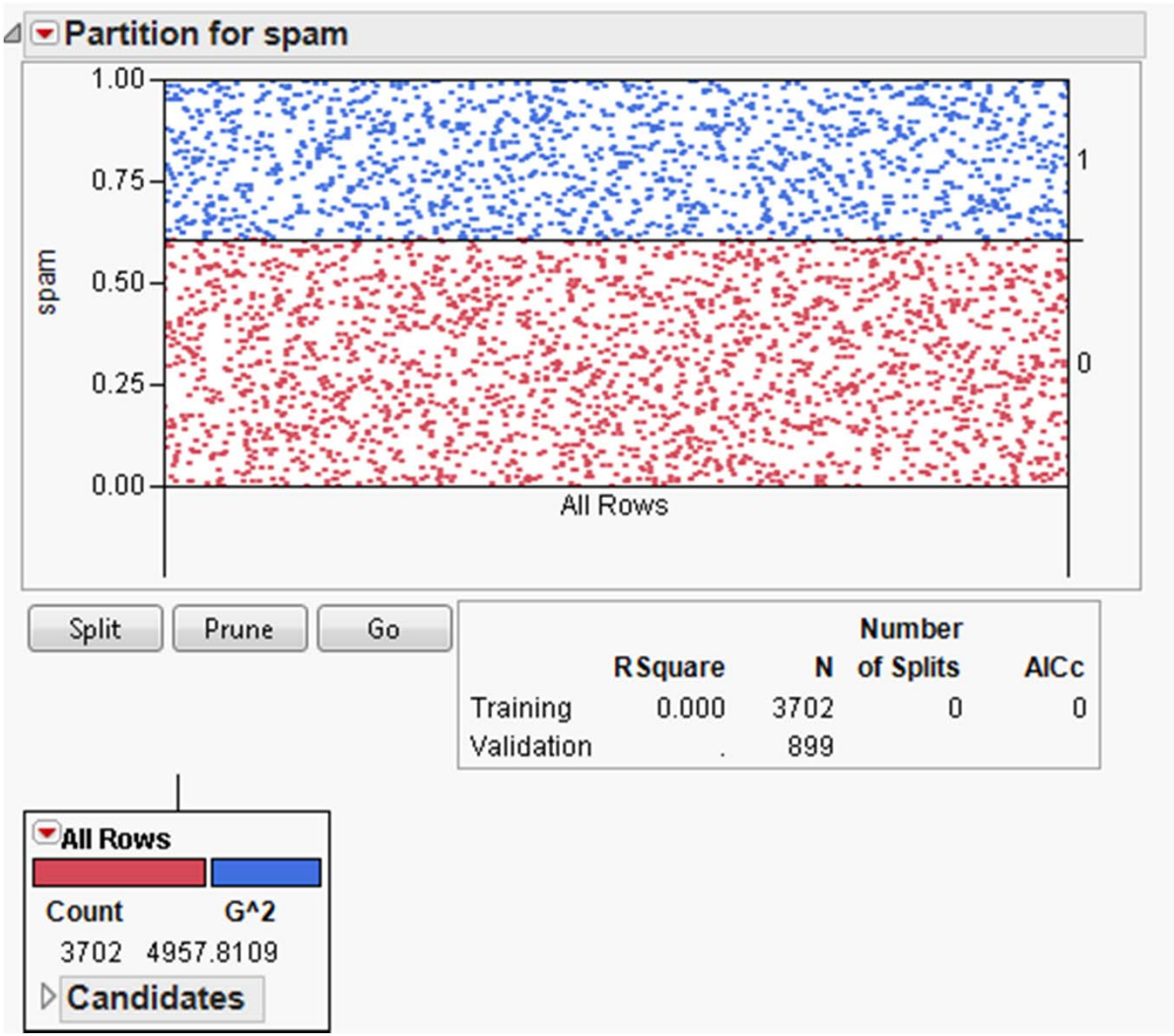
- Can have two or more categories
- Make sure attribute of Y variable is set to nominal in JMP®
- Splitting criterion is again LogWorth
- Ex: Predicting whether an inbox item is spam or email

Spam Data Example

Spam data base

- Collected June-July 1999
  - Spam 1813 (39.4%)
  - Non-Spam 2788 (60.6%)
- 57 attributes
  - Word Frequencies
  - Character Frequencies
  - Character Run Lengths
  - Used as example in Hastie, Tibshirani, and Friedman

Before Splitting



Split Candidate Information

- **LogWorth**: for each variable, the LogWorth if that variable is used for the next split
- **G^2**:The likelihood ratio chi-square. The G^2 displayed is difference in G^2 that would result from splitting at best split for that variable

$$G^2_{test} = G^2_{parentt} - (G^2_{right} + G^2_{left})$$

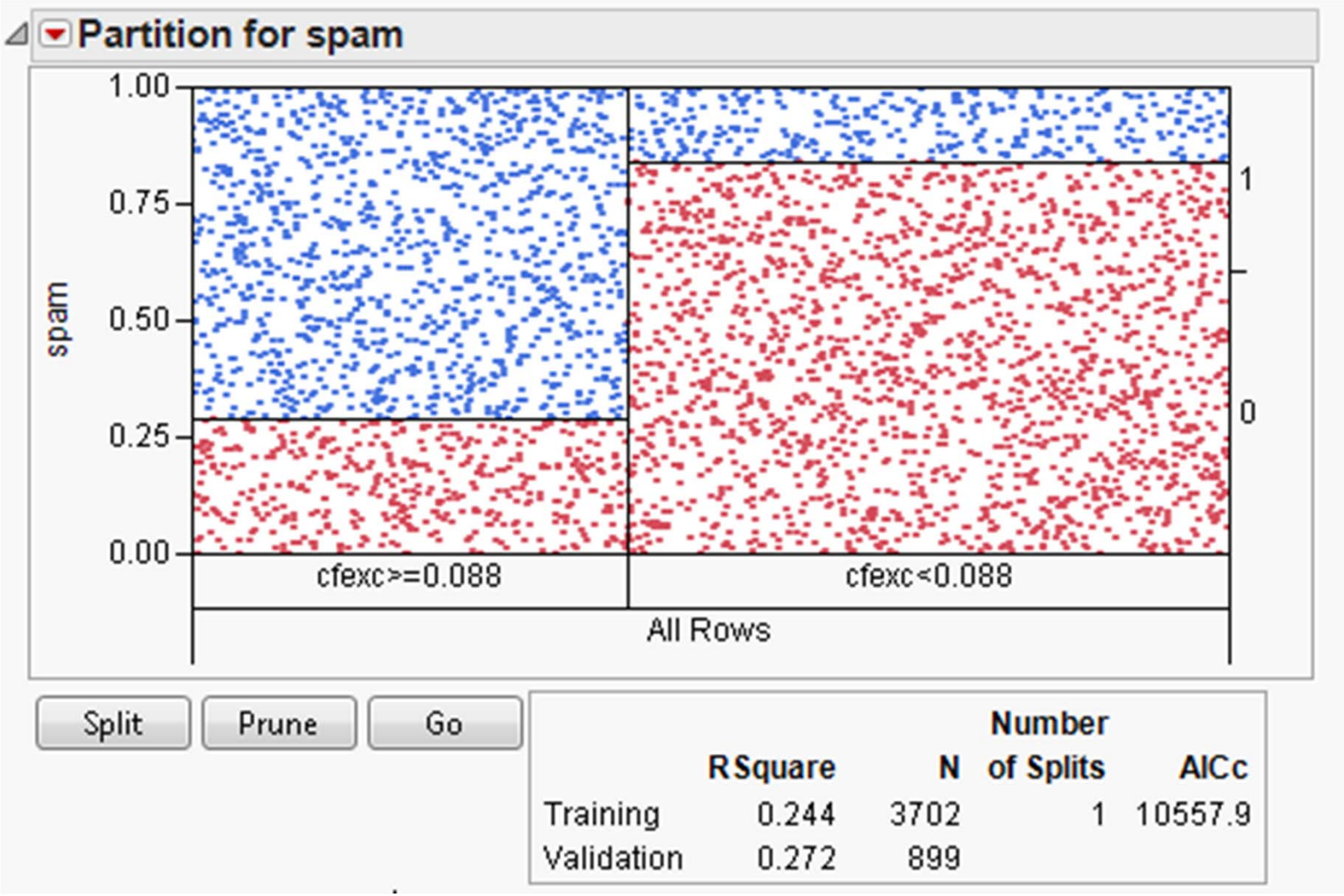
Candidates for Splitting

Candidates

Term	Candidate G^2		LogWorth
wftechnology	132.872823		47.0644159
wf1999	308.268237		123.2940567
wfparts	9.348483		1.4973276
wfpm	127.969258		44.4997174
wfdirect	30.394432		7.3765562
wfcs	119.904568		38.2444549
wfmeeting	205.245533		77.1669910
wforiginal	154.608671		53.3626595
wfproject	127.108621		43.9780226
wfre	124.004592		46.2377733
wfedu	251.948039		101.8807440
wftable	11.965113		2.2096502
wfconference	75.498041		22.2498082
cfsc	50.314576		14.4137528
cfpar	106.133042		37.9927846
cfbrack	69.368769		22.4745847
cfexc	1209.481043	>	711.4237070
cfdollar	1216.417500	<	656.0148635
cfpound	273.272635		119.2329560
crlaverage	777.906788		364.6878894
crllongest	765.879865		352.9020890
crltotal	567.938149		307.7881499

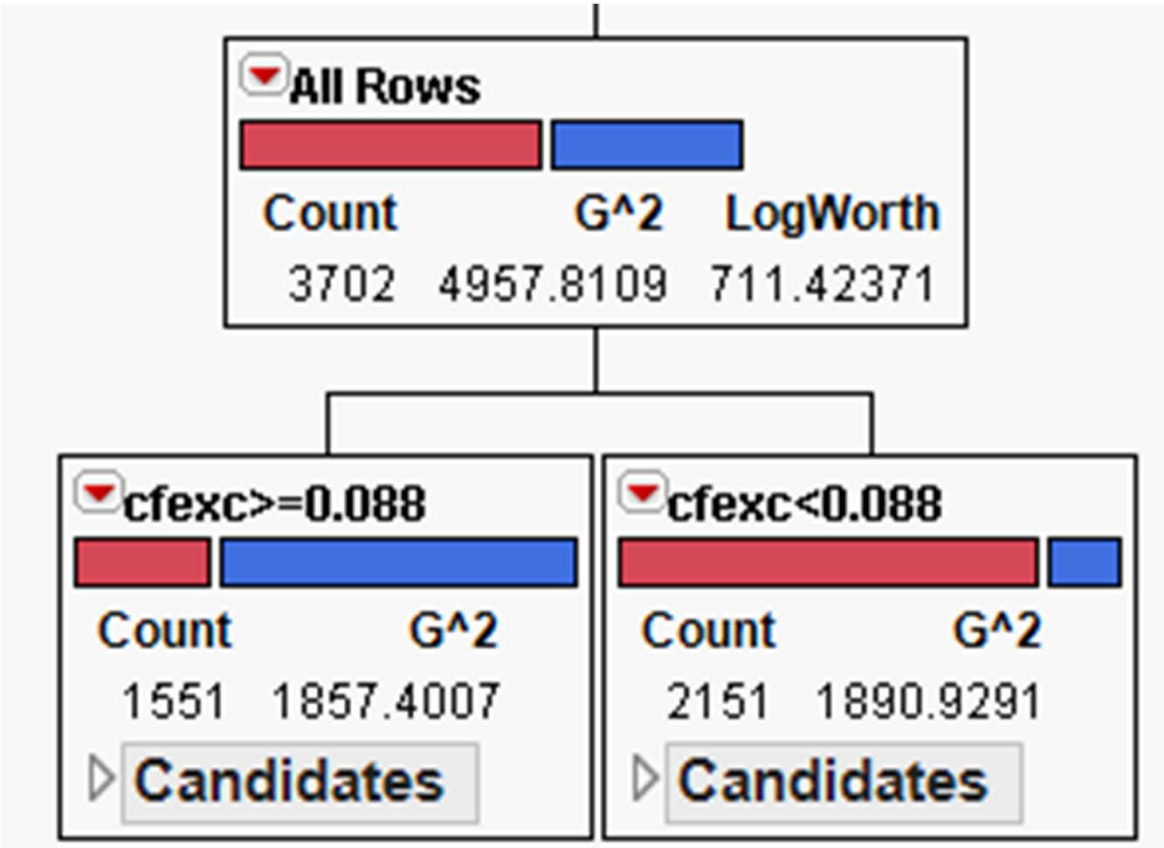
- G^2 is larger for cfexc
- LogWorth is larger for cfdollar
- Split is based on LogWorth so choose cfexc

After One Split

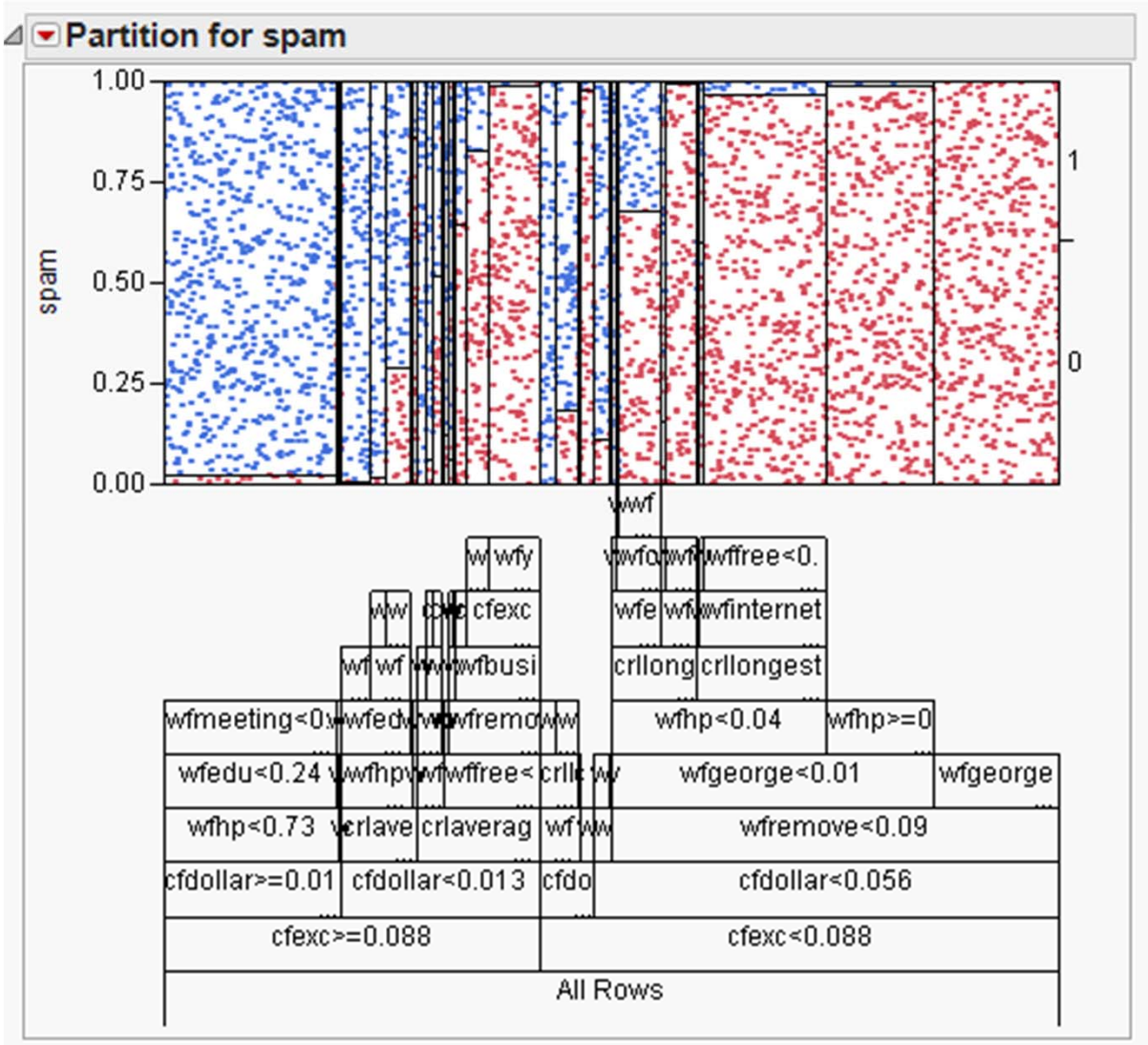


	RSquare	N	Number of Splits	AICc
Training	0.244	3702	1	10557.9
Validation	0.272	899		

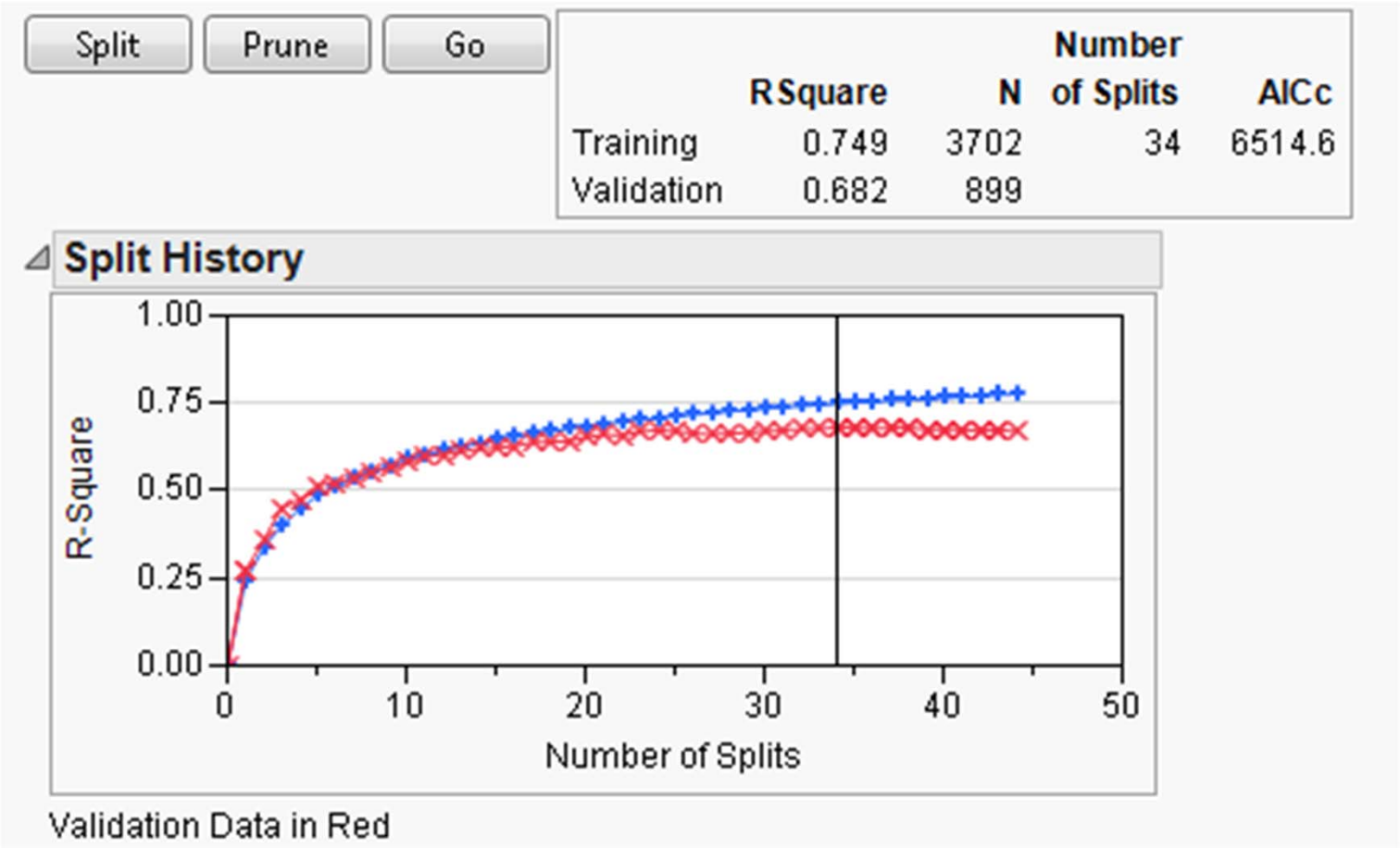
Tree After One Split



After Automatic Splitting



After Automatic Splitting



Fitting Criteria

JMP® uses several measures of fit

- Larger is Better
  - **Entropy RSquare:** compares the log-likelihoods from the fitted model and the constant probability model
  - **Generalized Rsquare**: Value is 1 for a perfect model, and 0 for a model no better than a constant model

Fitting Criteria

Smaller is Better

- p is fitted probability for event that occurred
  - **Mean -Log p:** average of -log(p)
  - **RMSE:** root mean square error, where the differences are between the response and p
  - **Mean Abs Dev** : average of the absolute values of the differences between the response and p

Fit Details for Spam Decision Tree

Fit Details

Measure	Training	Validation	Definition
Entropy RSquare	0.7487	0.6824	1-Loglike(model)/Loglike(0)
Generalized RSquare	0.8580	0.8125	(1-(L(0)/L(model))^(2/n))/(1-L(0)^(2/n))
Mean -Log p	0.1682	0.2141	$\sum$ -Log(p[j])/n
RMSE	0.2187	0.2434	$\sqrt{\sum(y[j]-p[j])^2/n}$
Mean Abs Dev	0.0982	0.1100	$\sum$ |y[j]-p[j]|/n
Misclassification Rate	0.0629	0.0768	$\sum$ (p[j]≠pMax)/n
N	3702	899	n

Missclassification Rate

Missclassification

- If we classify a case as being in the category with the highest predicted probability, we will sometimes be wrong
- Smaller misclassification rate is better
- Sometimes we prefer one kind of error over another
  - Diseased person classified as non-diseased may be worse than the other way around

Confusion Matrix

- **Predicted category**: category with highest predicted probability
- Confusion matrix compares Actual Category to Predicted Category for **training** and **validation** data

Confusion Matrix for Spam Data

- Specificity Validation: 511/(511+26) = 0.95
- False Positive Validation: 1-Specificity = 0.05
- Sensitivity Validation: 319/(319+43) = 0.88

Confusion Matrix

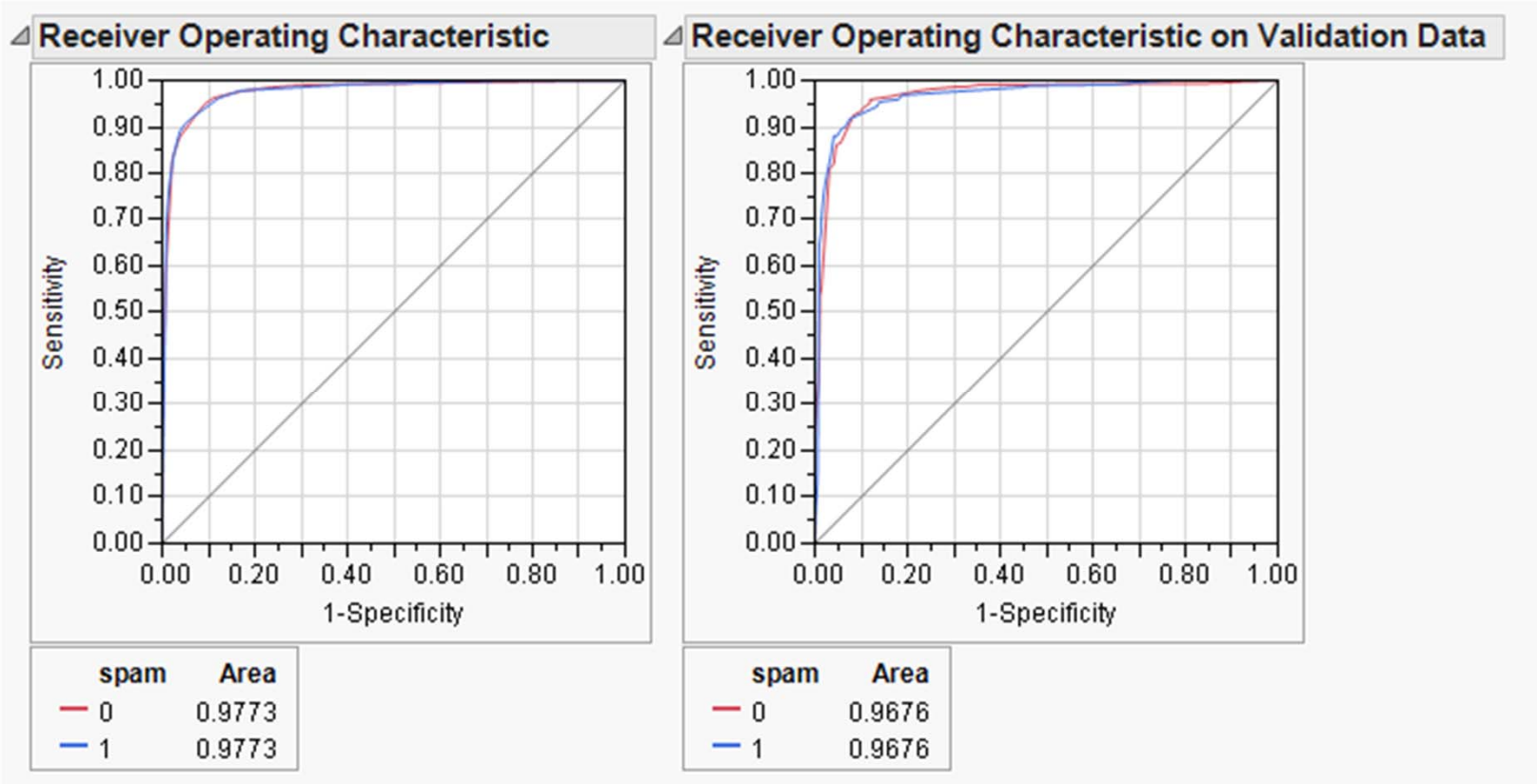
Actual	Predicted	
Training	**0**	**1**
0	2167	84
1	149	1302

Actual	Predicted	
Validation	**0**	**1**
0	511	26
1	43	319

ROC Curve

- True Positive y-axis is labeled “Sensitivity” and the False Positive X-axis is labeled “1-Specificity”
- Diagonal line is chance
- Partition creates an ROC curve for each response level versus the other levels.
- If there are only two levels, one is the diagonal reflection of the other
- Area under the curve: higher is better

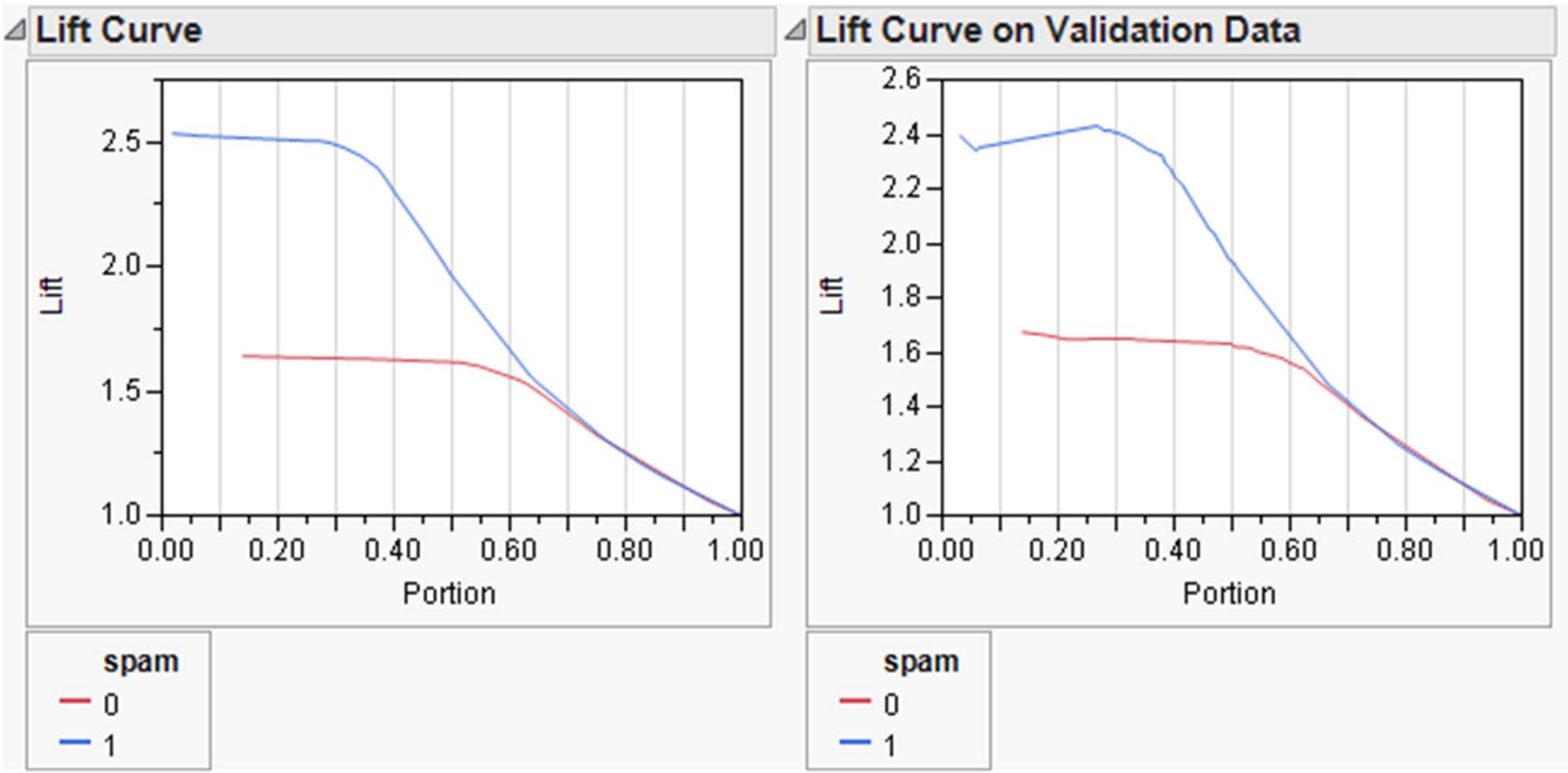
ROC Curve for Spam Data



Lift Curve

- Same information as ROC curve
- Dramatizes the richness of the ordering at the beginning.
- The Y-axis shows the ratio of how rich that portion of the population is in the chosen response level compared to the rate of that response level as a whole.
- All lift curves reach (1,1) at the right, as the population as a whole has the general response rate.

Spam Data Lift Curve



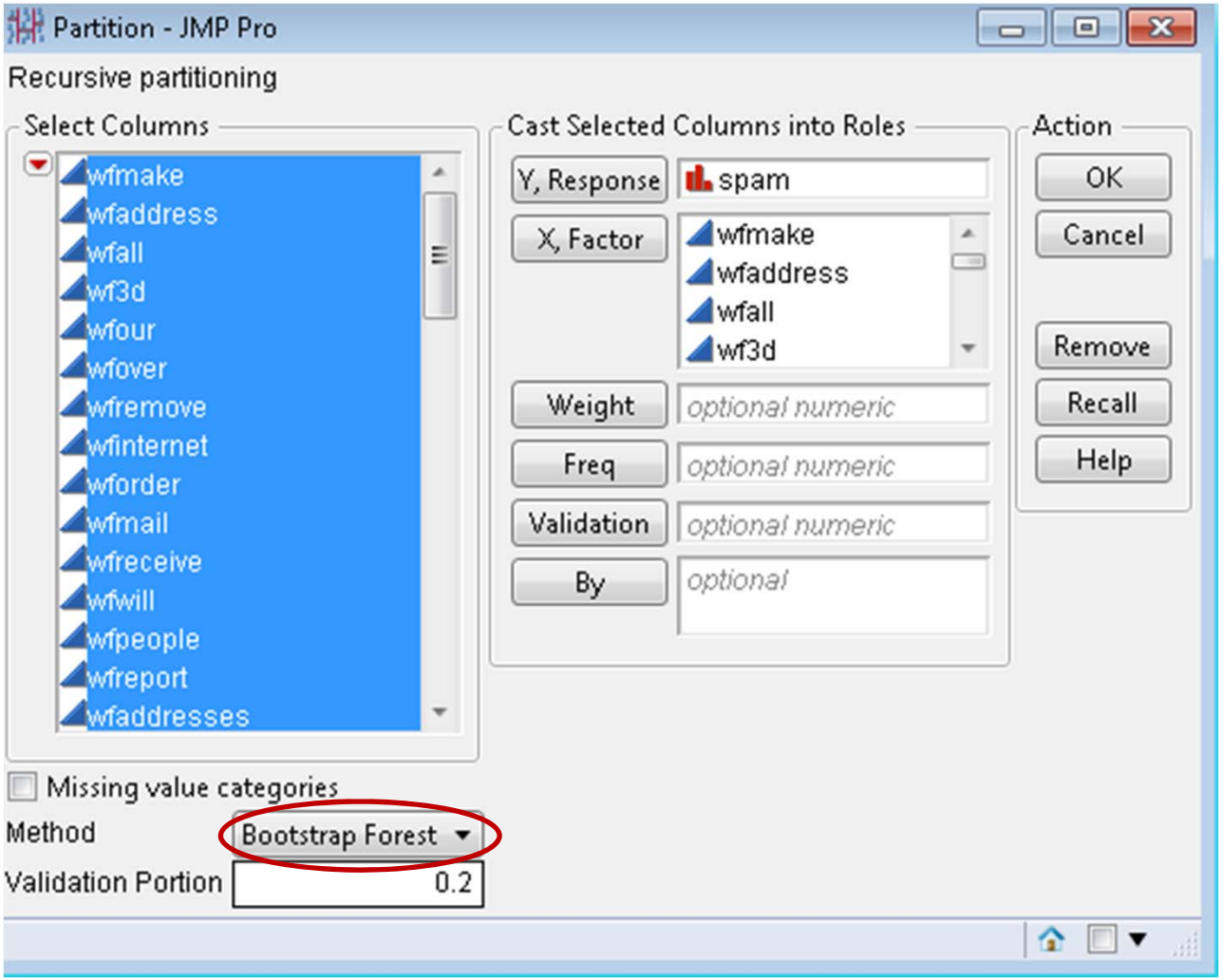
Random Forests

- AKA Bootstrap Forests, Bagging (Bootstrap Aggregating)
- Combines the outputs of many "weak" classifiers to make a powerful "committee"
- No longer have a single "Tree" that you can view
- The constituent trees can be viewed

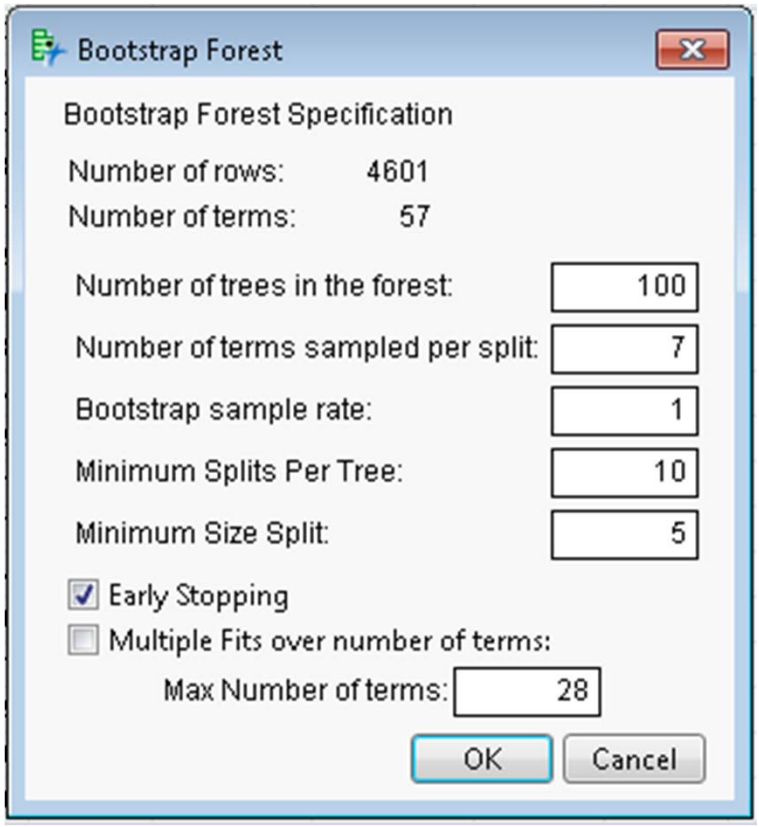
Random Forests Steps

- Choose a bootstrap sample of data (with replacement)
- Choose a random sample of predictors
- Grow a tree until reach stopping criterion
- Do this many times and combine the information from all trees by averaging

Bootstrap Forest for Spam Data

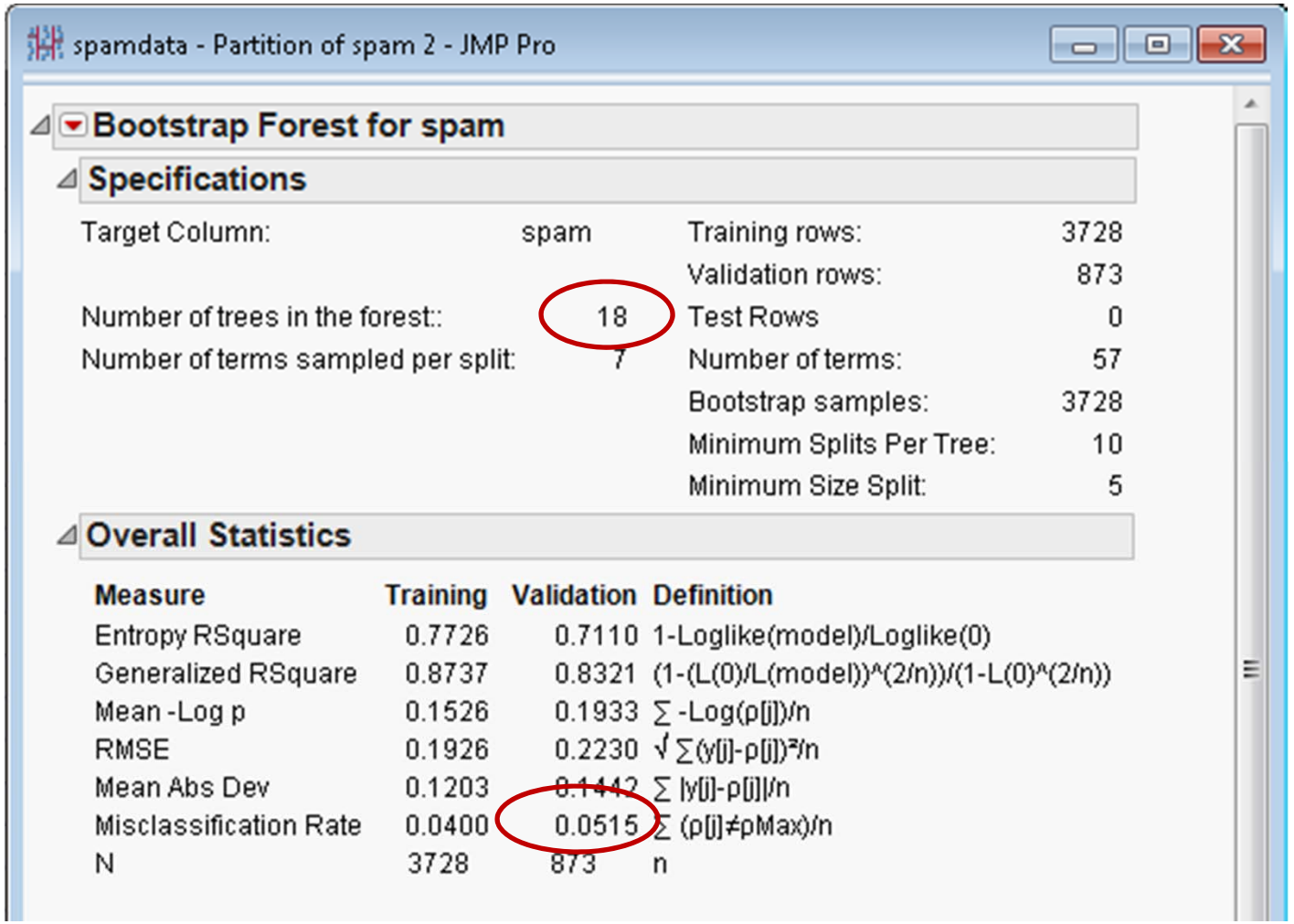


Bootstrap Forest for Spam Data

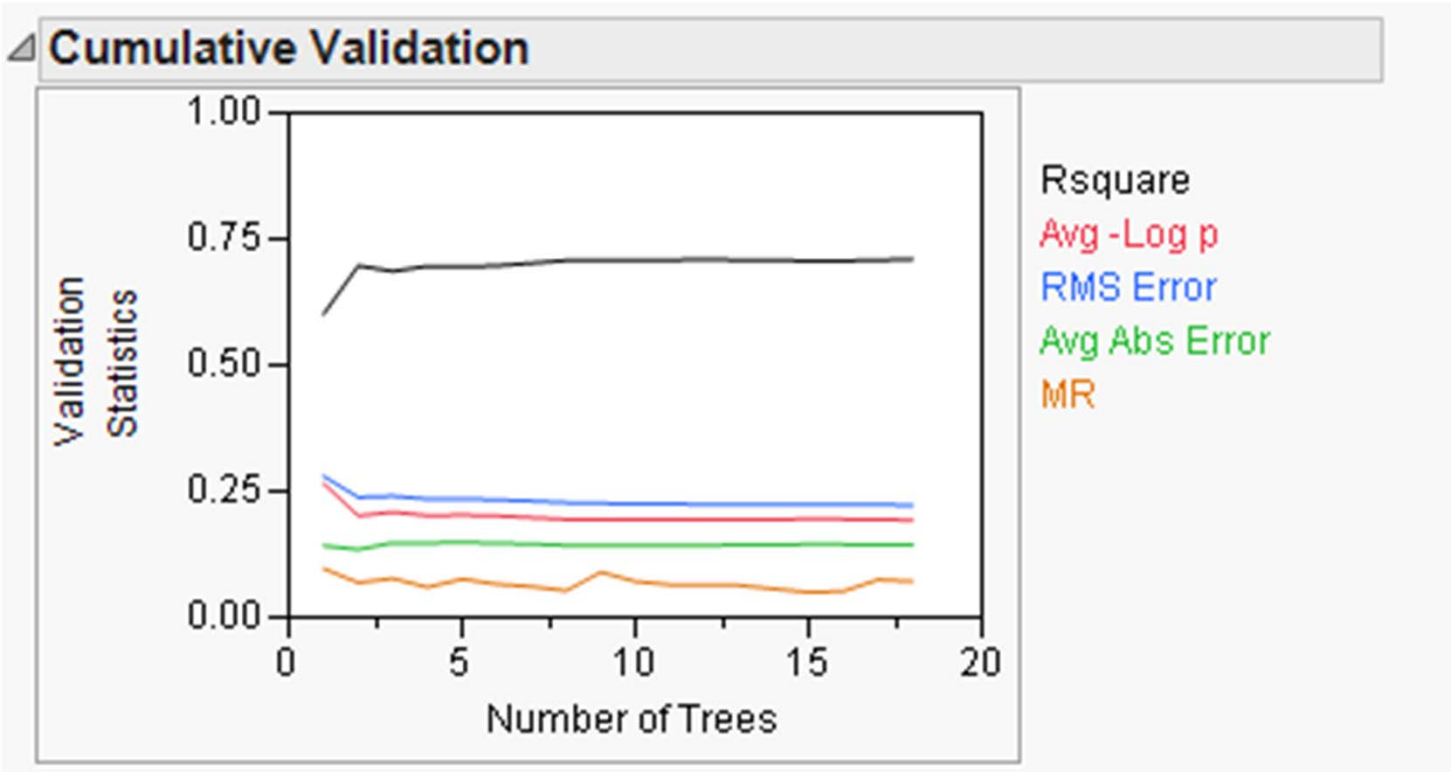


- Number of trees
- Number of items, recommend choosing square root of total predictors
- Early stopping, if validation data shows no improvement

Bootstrap Forest for Spam Data



Bootstrap Forest for Spam Data



Bootstrap Forest for Spam Data

Cumulative Details

NTree	Validation RSquare	Avg -Log p	RMSE	Avg Abs Error	Misclassification Rate
1	0.602118	0.26619	0.28084	0.142659	0.097365
2	0.697818	0.202165	0.238869	0.13545	0.069874
3	0.686992	0.209408	0.241064	0.14845	0.077892
4	0.697107	0.202641	0.235315	0.148116	0.06071
5	0.695493	0.203721	0.235713	0.149261	0.076747
6	0.697719	0.202231	0.233916	0.147745	0.066438
7	0.702982	0.198711	0.231569	0.146206	0.061856
8	0.708304	0.19515	0.2282	0.143596	0.053837
9	0.70873	0.194865	0.226981	0.143609	0.090493
10	0.70954	0.194323	0.225369	0.143117	0.072165
11	0.70928	0.194497	0.22538	0.143922	0.065292
12	0.710403	0.193746	0.224947	0.143444	0.065292
13	0.709189	0.194558	0.224862	0.144287	0.064147
14	0.709487	0.194358	0.224543	0.144283	0.057274
15	0.706697	0.196225	0.225373	0.146069	0.050401
16	0.707643	0.195592	0.224973	0.145647	0.052692
17	0.708639	0.194926	0.224402	0.144927	0.075601
18	0.711032	0.193325	0.223023	0.144231	0.072165

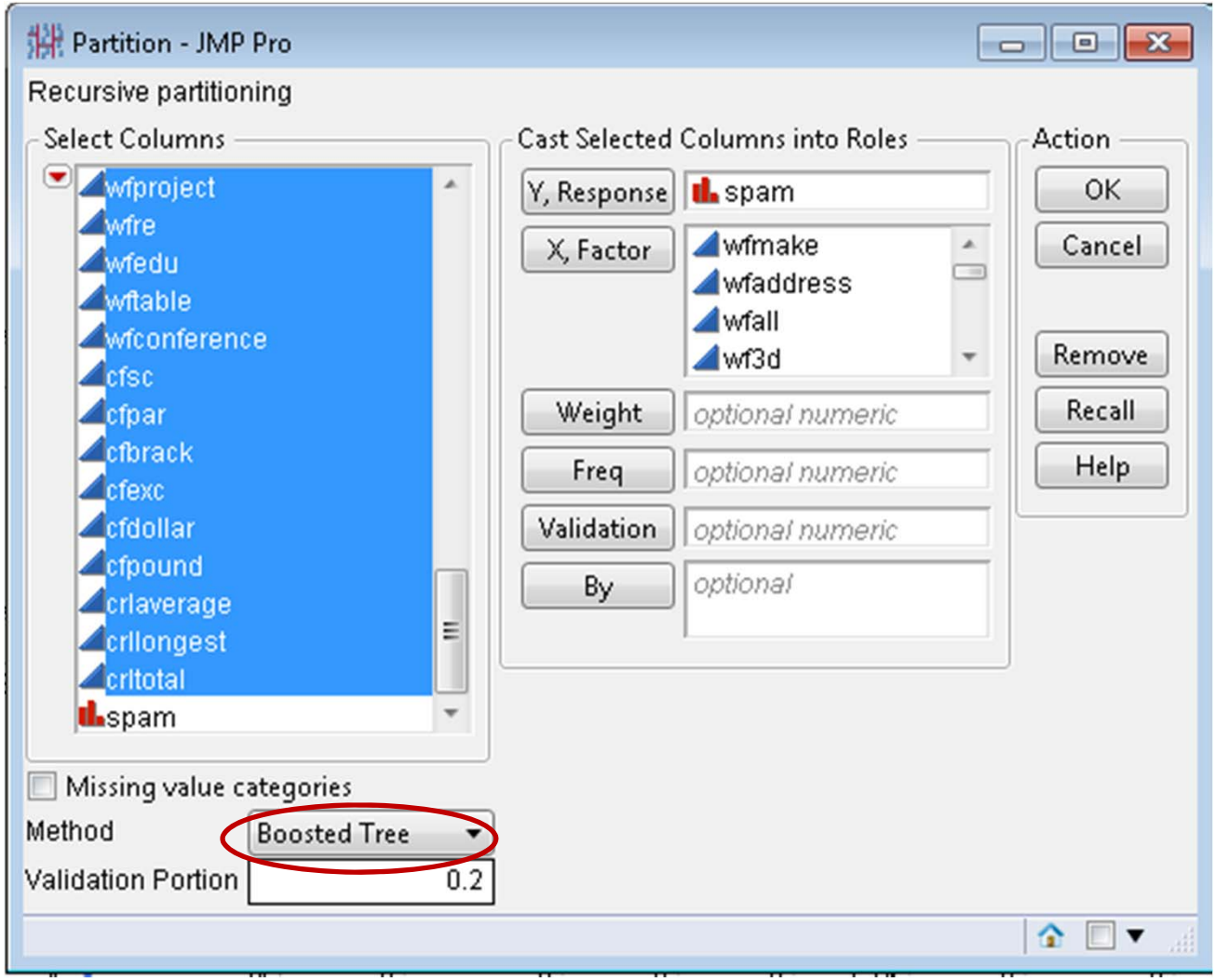
Boosted Trees

- Like bagging, boosting combines the outputs of many "weak" classifiers to make a powerful "committee"
- Sequentially apply the weak classification scheme to modified versions of the data
- At each stage, modify the data by giving more weight to the misclassified cases
- Combine the information across all stages of the process
- Idea is to improve accuracy

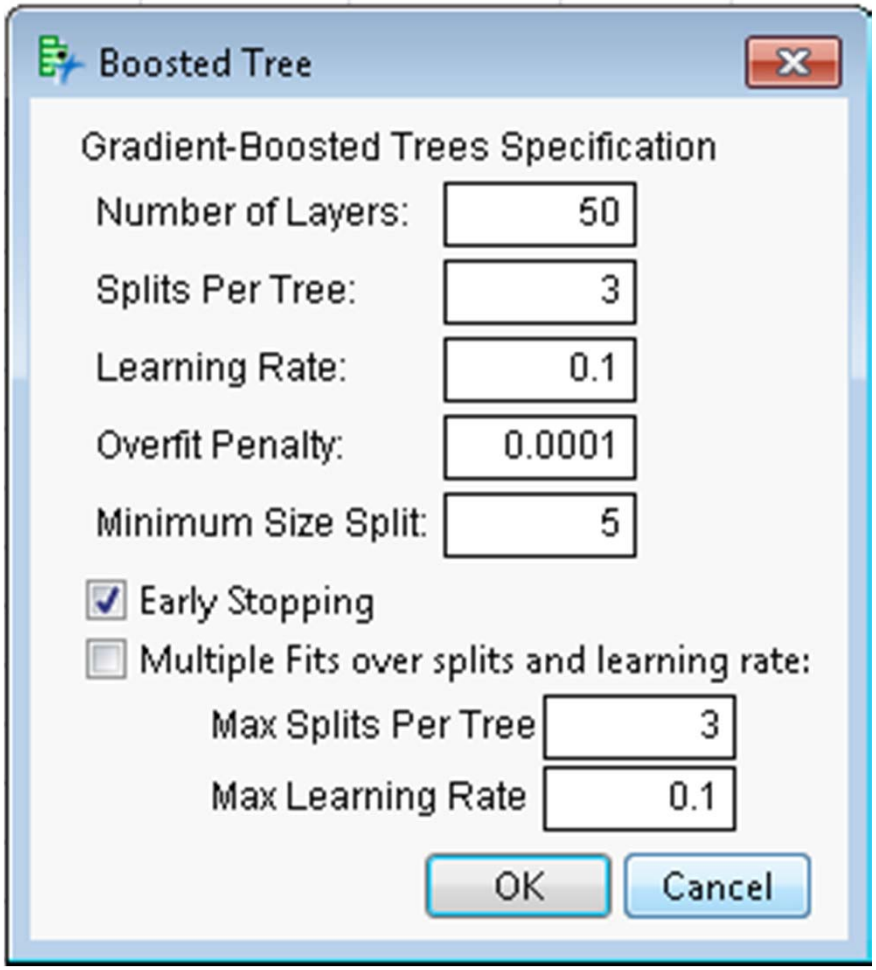
Boosted Tree Steps in JMP®

- Tree at each stage is short, typically 1-5 splits
- After the initial tree, each stage fits the residuals from the previous stage
- Process continues until the specified number of stages is reached, or, if validation is used, until fitting an additional stage no longer improves the validation statistic
- Final prediction is the sum of the estimates for each terminal node over all the stages

Boosted Tree for Spam Data

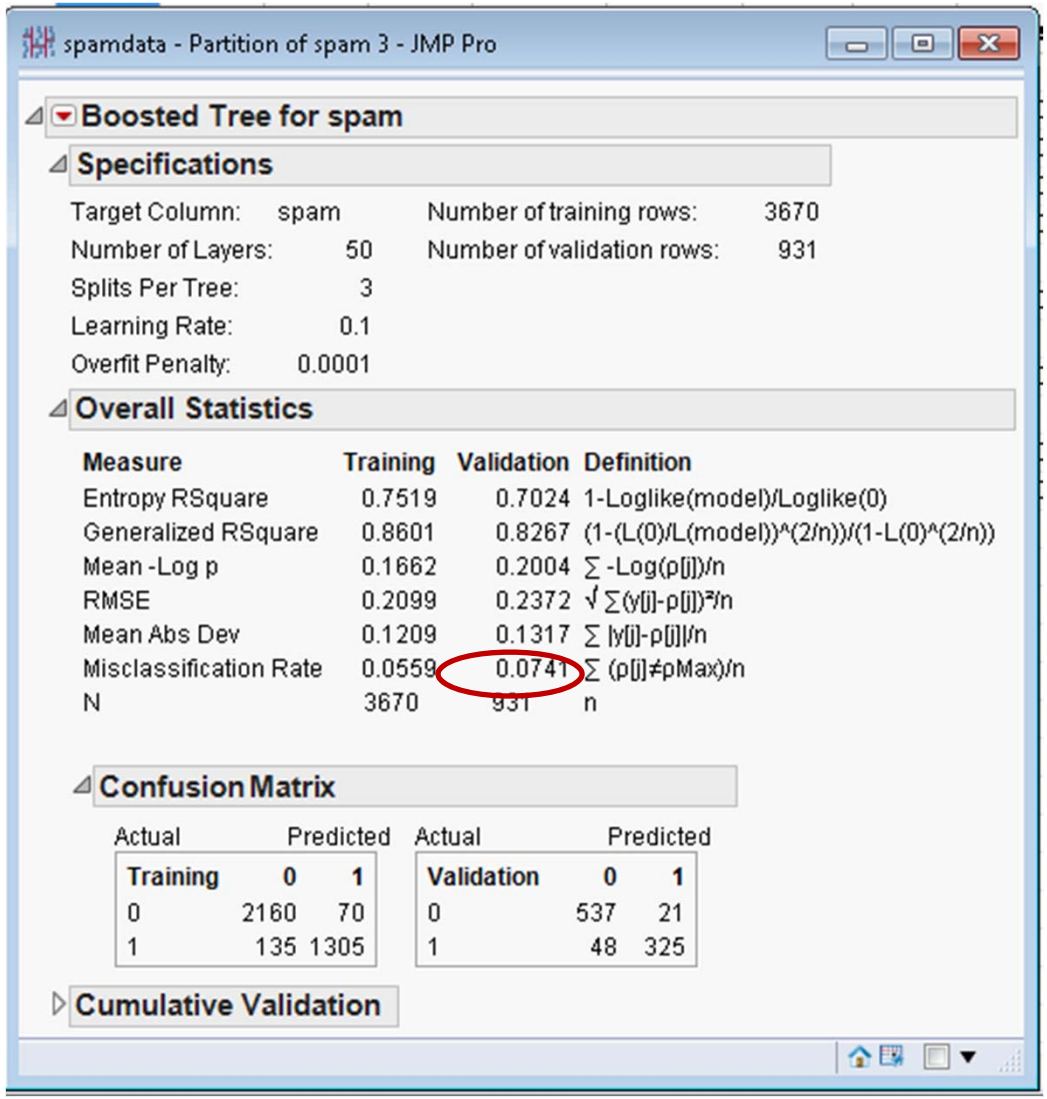


Boosted Tree for Spam Data

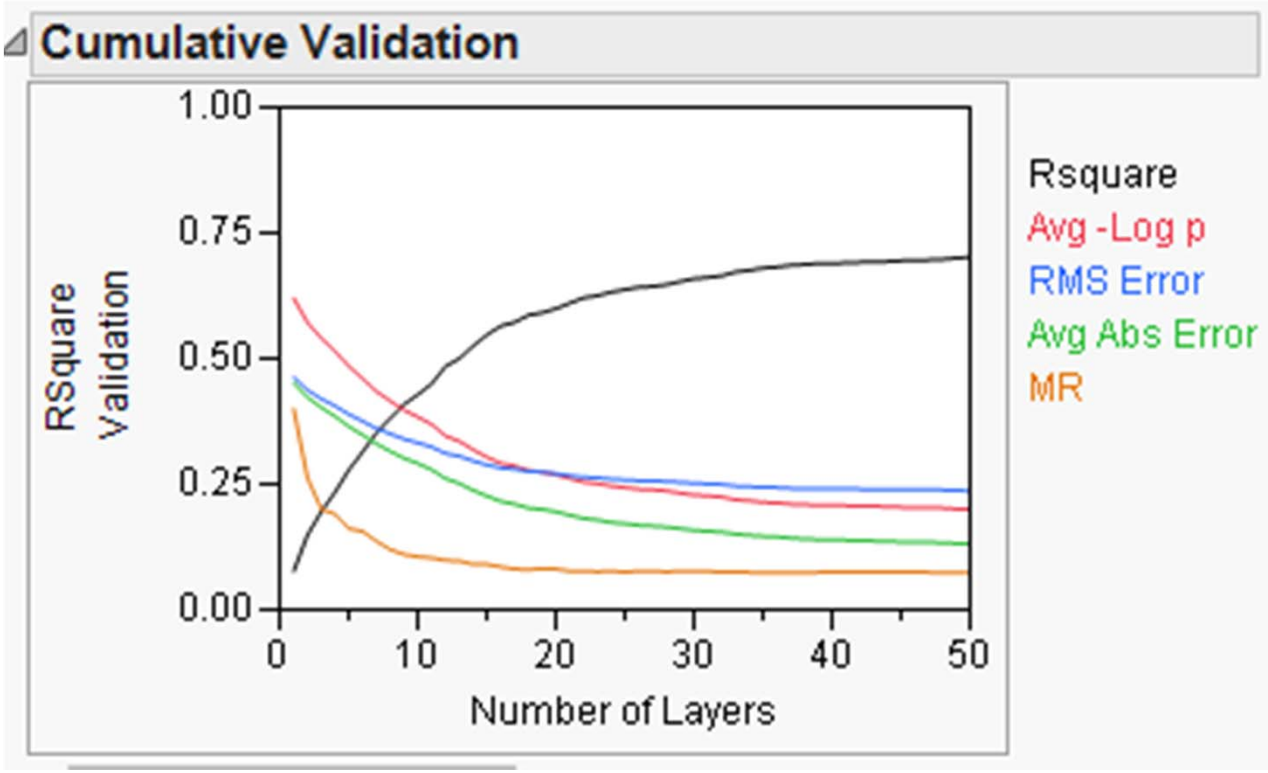


- Number of layers
- Splits per tree
- Early stopping, if validation data shows no improvement

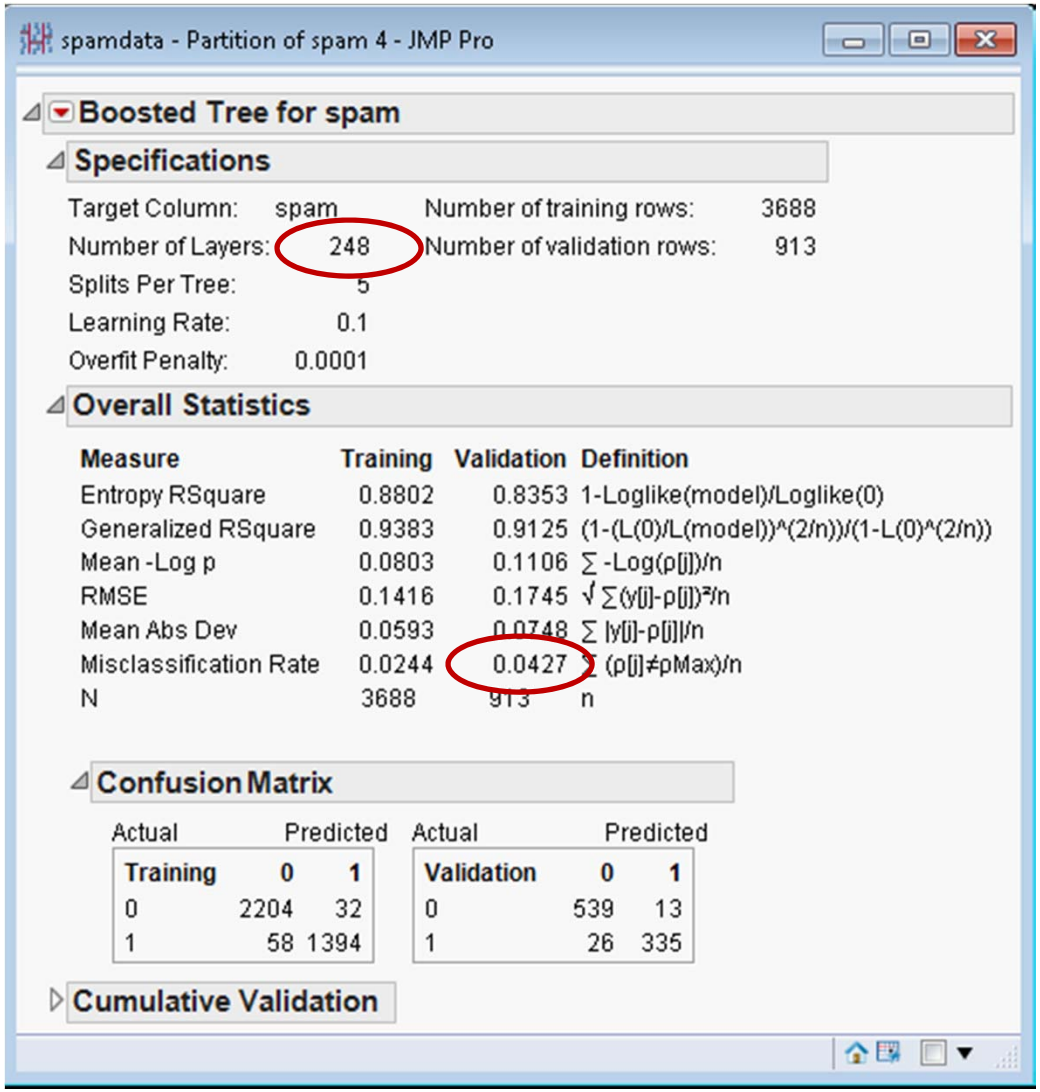
Boosted Tree Results



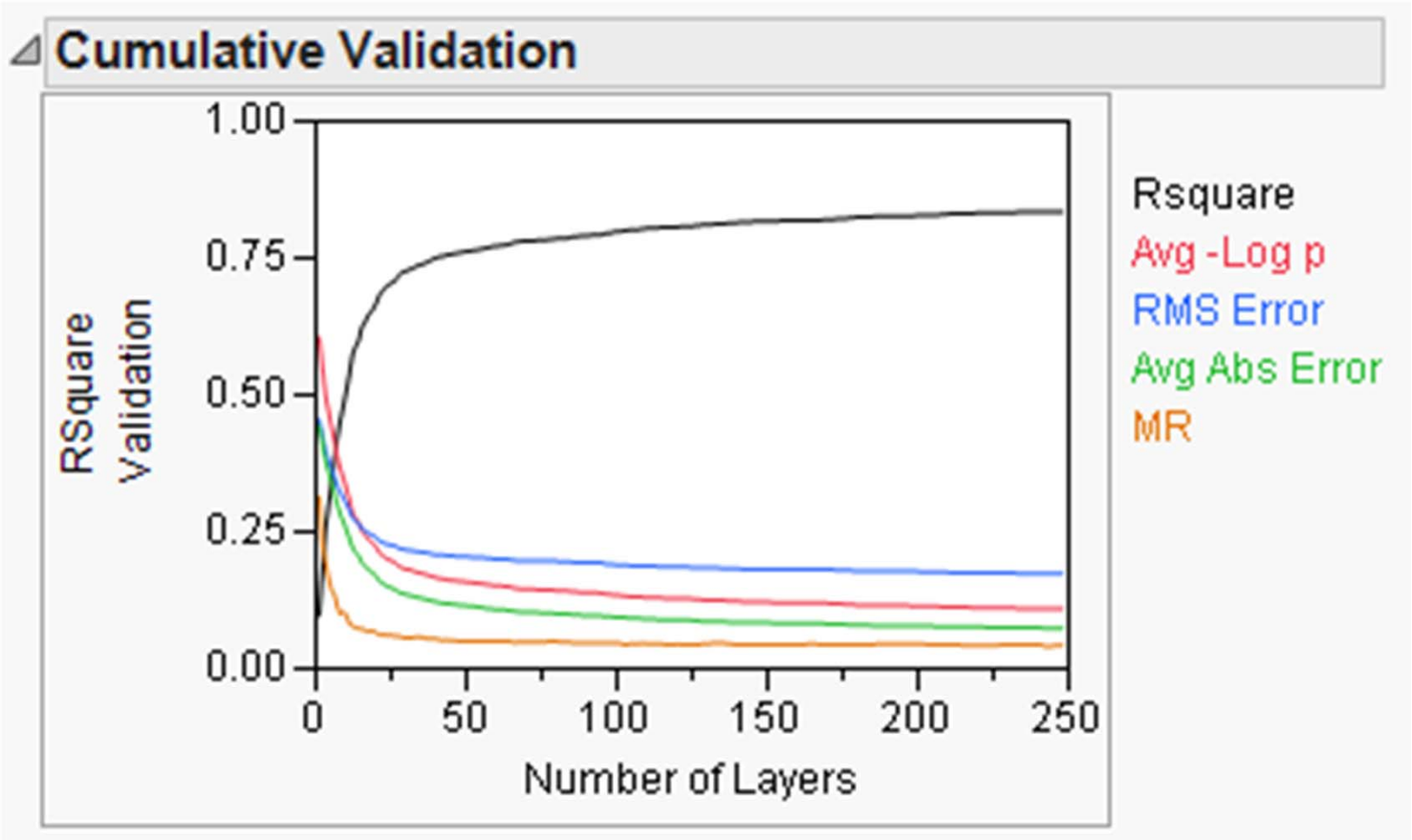
Boosted Tree for Spam Data



Boosted Tree for Spam Data
More Layers



Boosted Tree for Spam Data
More Layers



Advantages of JMP® for Recursive Partitioning

- Can use SAS® data sets directly
- Ease of access to partition platform
- Beautiful graphics
- Highly interactive
- Great documentation

References

- JMP®, Version 10, SAS Institute Inc., Cary, NC, 2012.
- SAS Institute Inc. 2012. *JMP® 10 Modeling and Multivariate Methods*. Cary, NC: SAS Institute Inc.
- Monte Carlo Calibration of Distributions of Partition Statistics, John Sall, SAS Institute, Nov 18, 2002.
- spambase.DOCUMENTATION at the UCI Machine Learning Repository http://www.ics.uci.edu/~mlearn/MLRepository.html
- Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition, Springer, 2009, 745 pp.