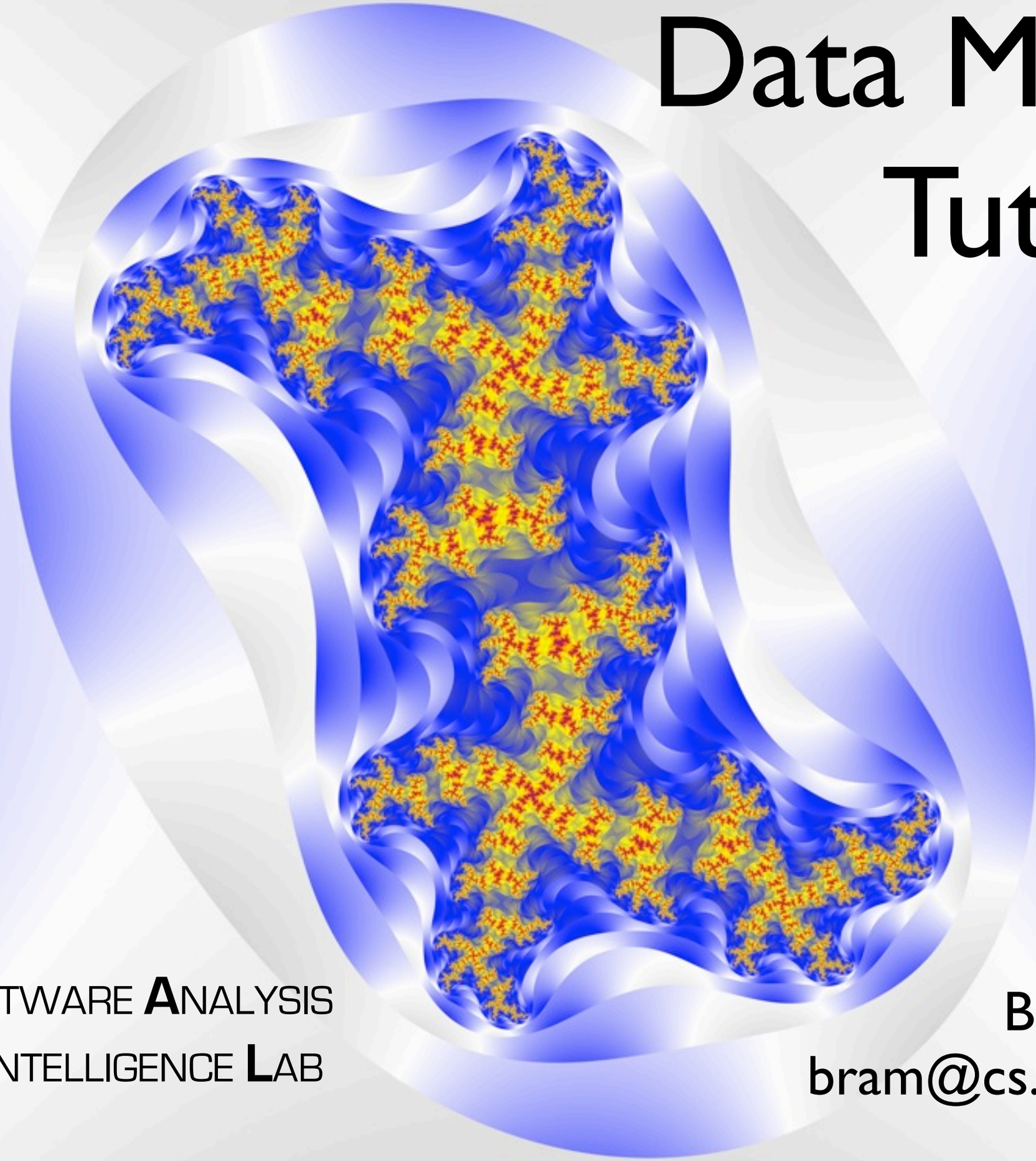
Data Mining Tutorial



SOFTWARE ANALYSIS & INTELLIGENCE LAB

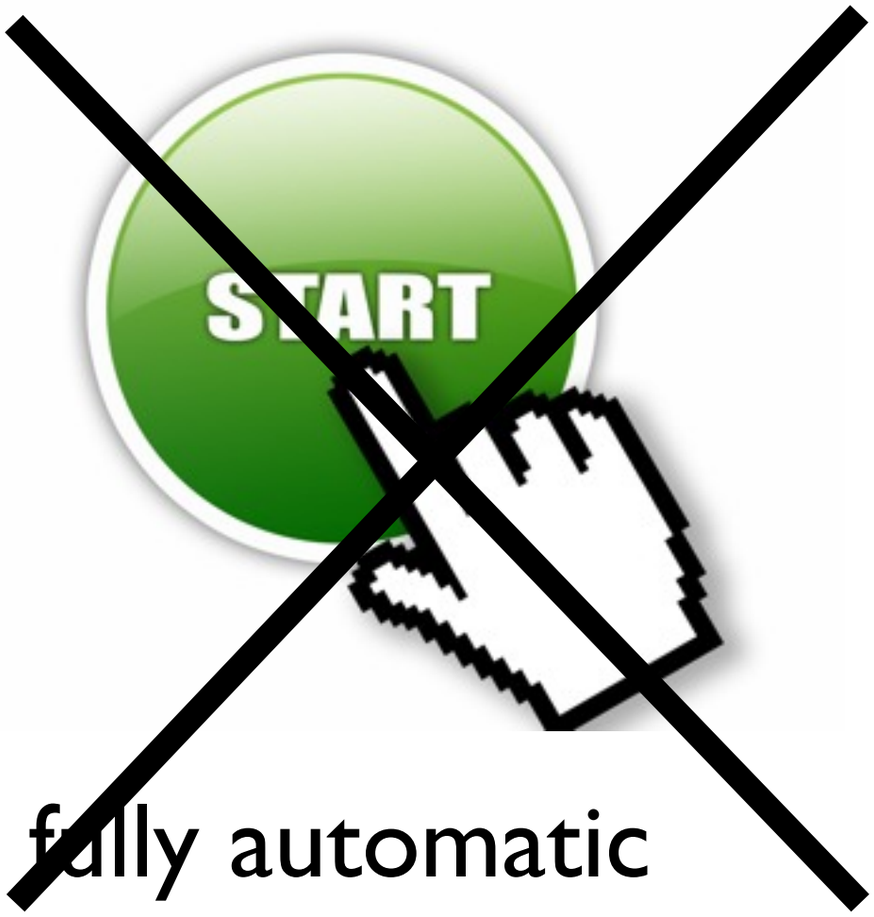
Bram Adams
bram@cs.queensu.ca

Data Mining is ... ?

Data Mining is ... !

{basic principles}

Data Mining is ... !



fully automatic

human judgment

Data Mining is ... ?

Data Mining is ... !

simplicity

Mining Raw Data ...

bug repository

mailing list

chat logs

source code repository

wiki

manual

requirements documents

package repository

design documents

...to Detect Patterns...

bug prediction model

bug fixing effort model

duplicate bug reports

change coupling

bug triaging patterns

test output classification

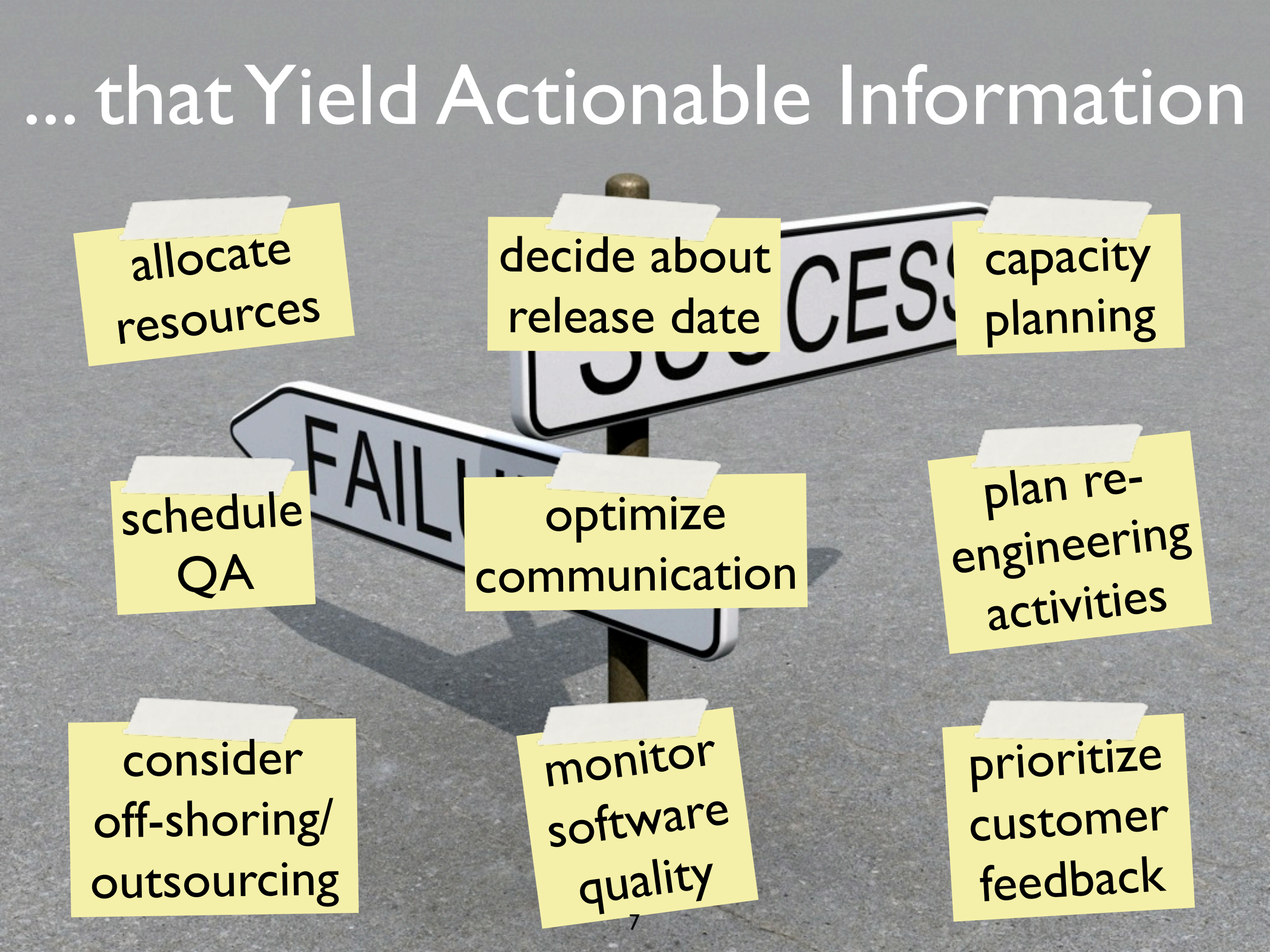
email classifier

invariants

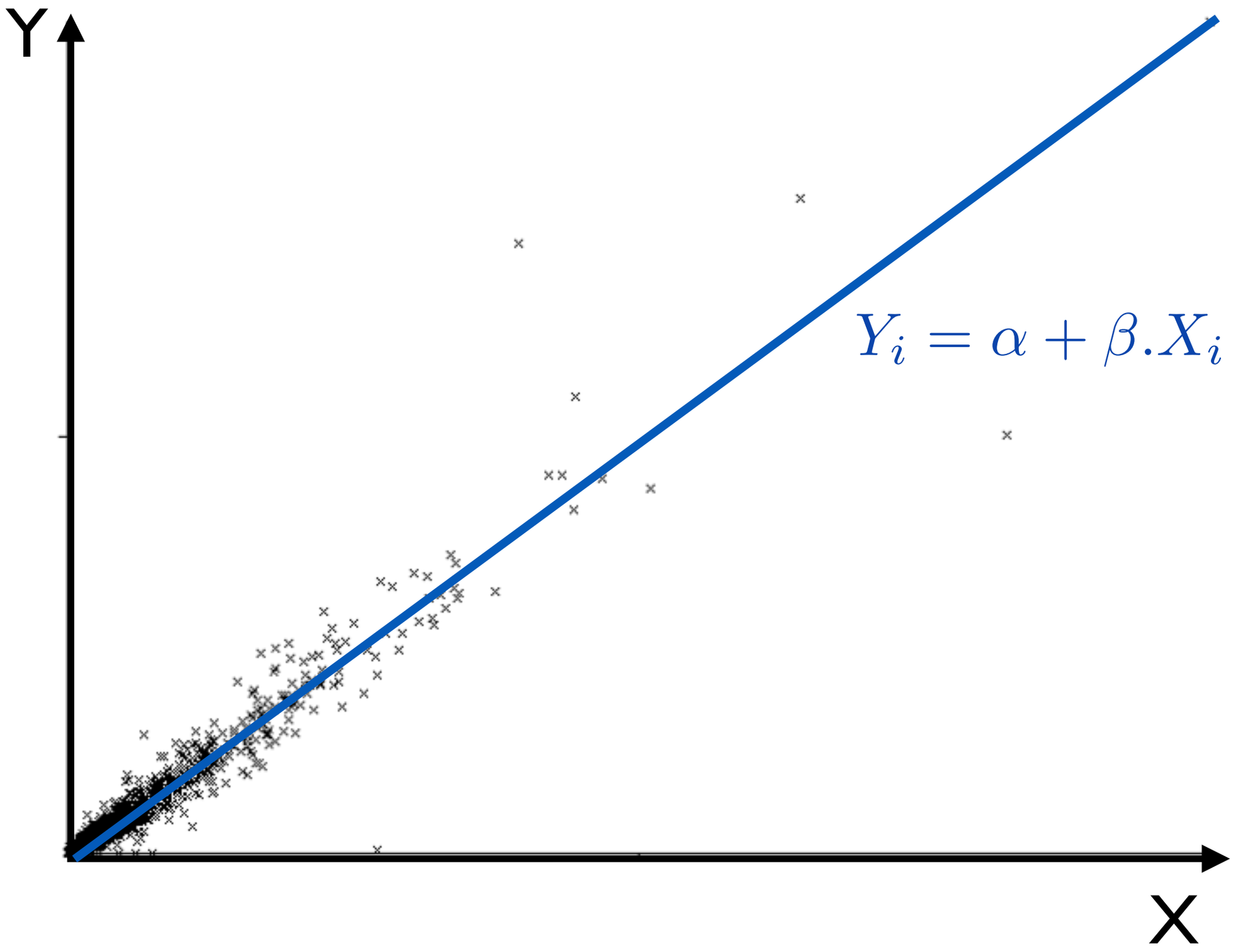
subsystem clustering

http://lunatic.no/ol/img2aschtml.php

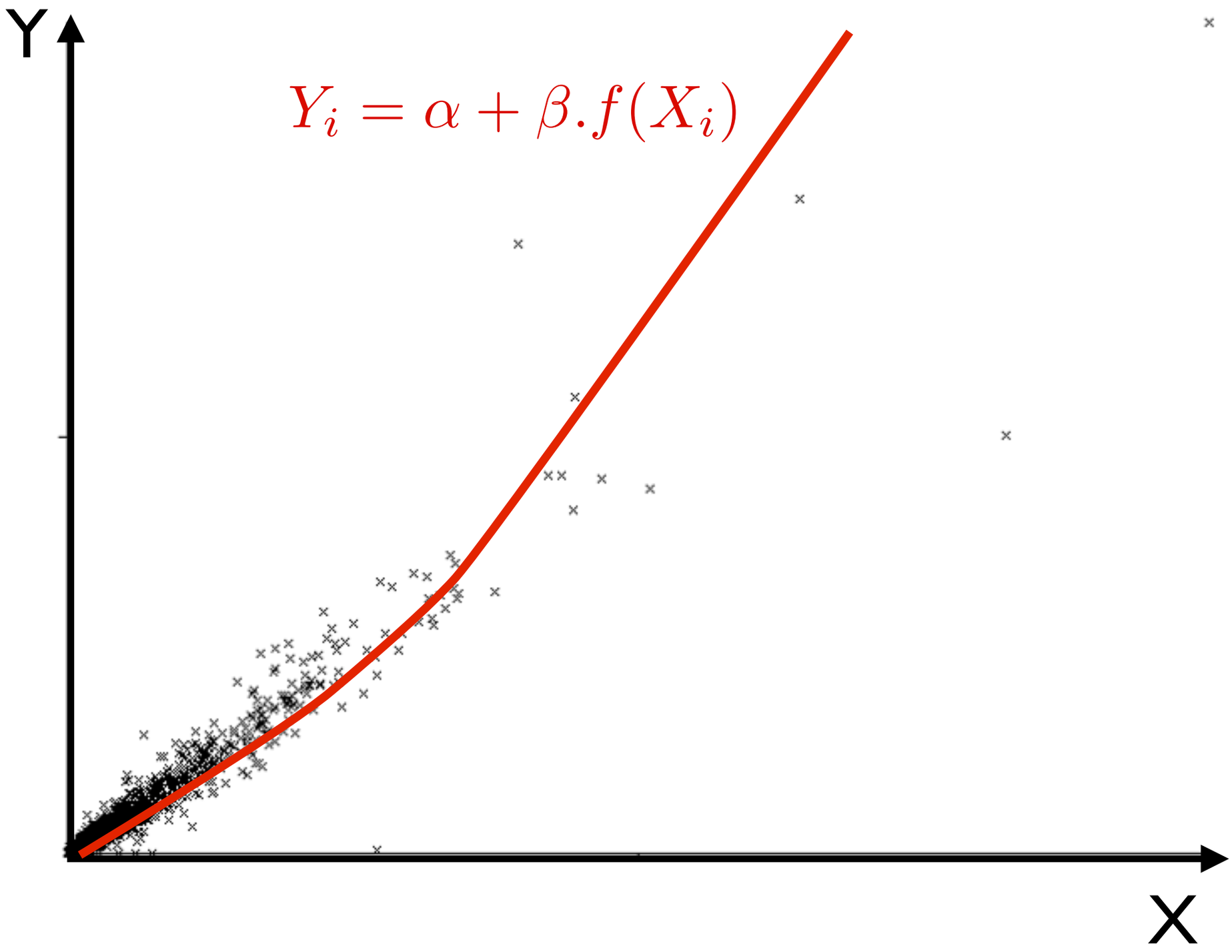
... that Yield Actionable Information



Linear Regression

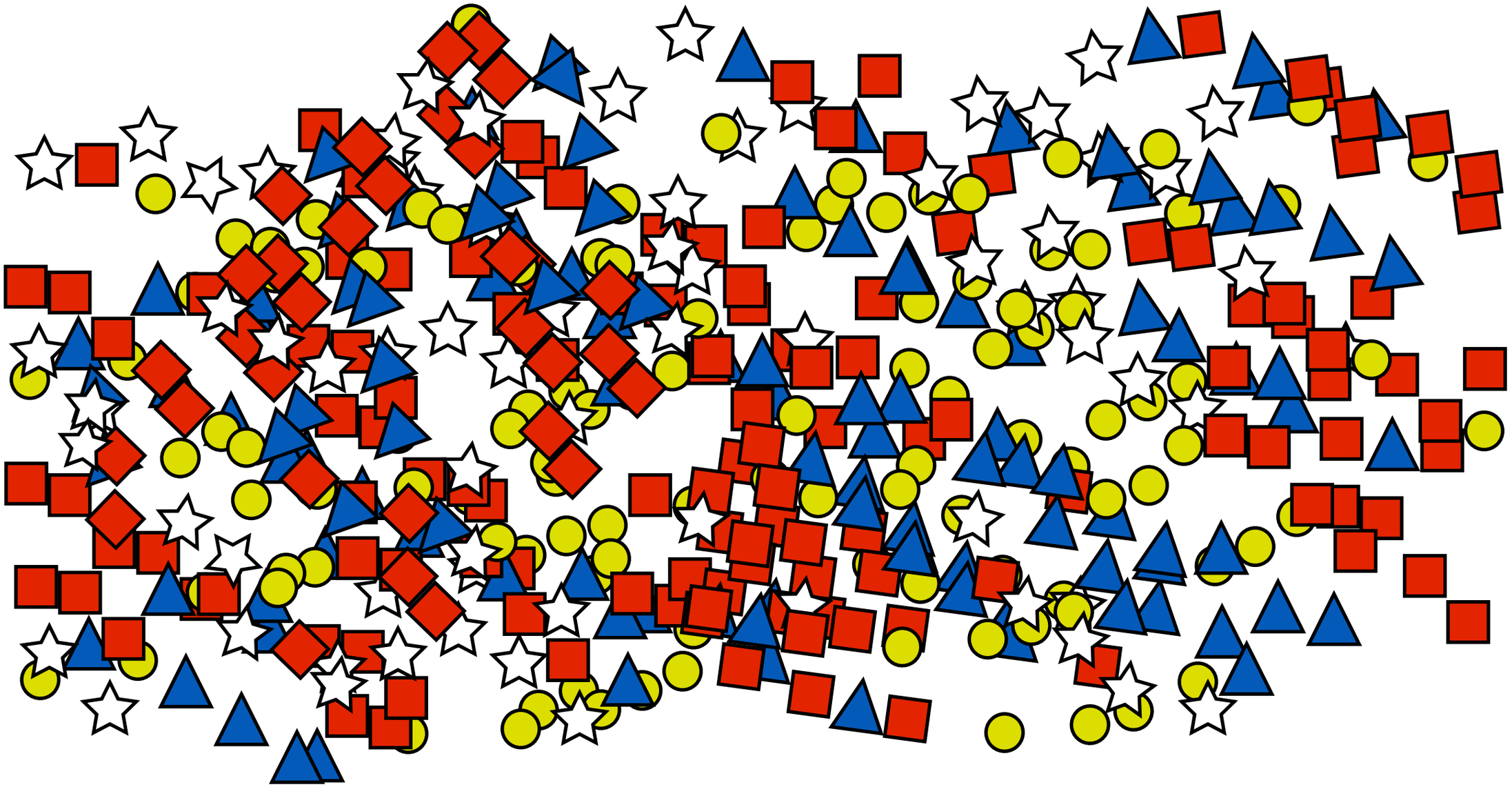


Linear Regression

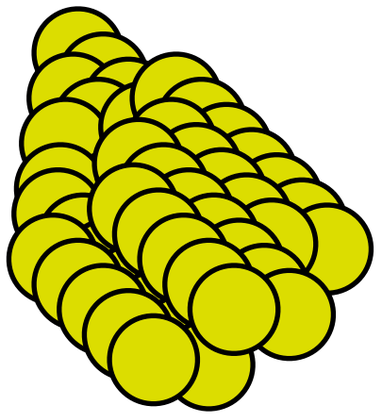


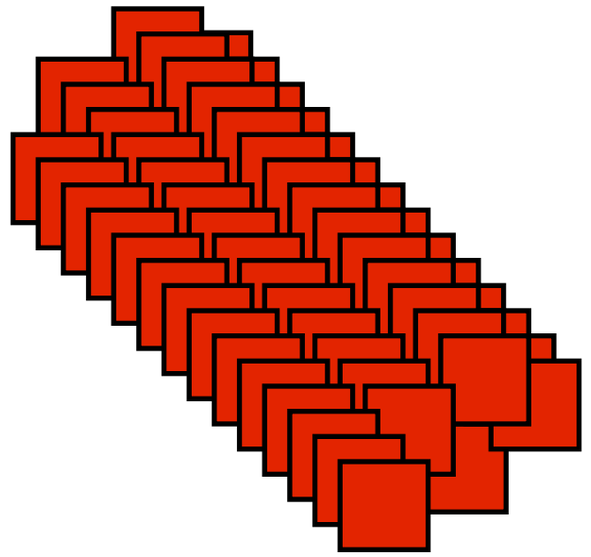
Classification

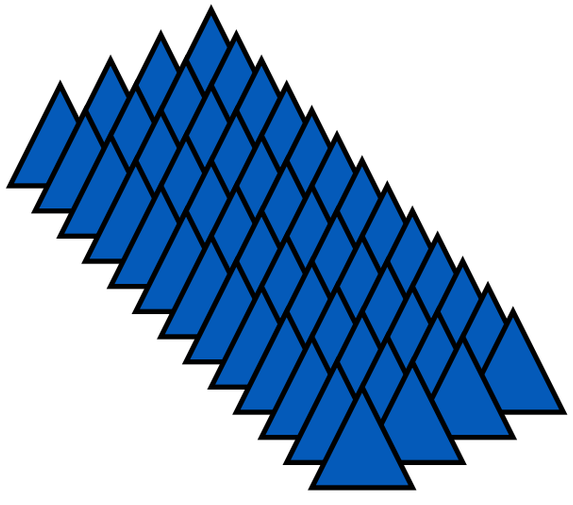
Clustering

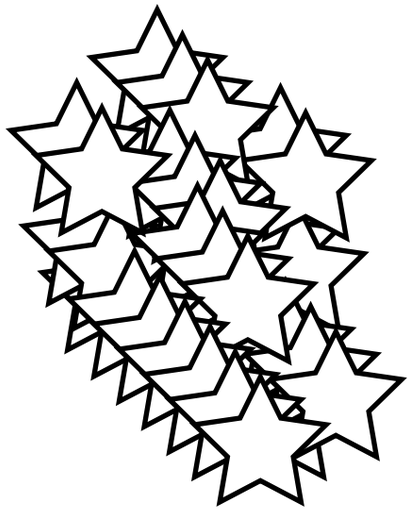


Clustering









Association Rule Mining

???

THIS TUTORIAL:

BINARY CLASSIFICATION

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Open file... | Open URL... | Open DB... | Generate... | Undo | Edit... | Save...

Filter

Choose **Discretize** -R 2-7 | Apply

Current relation

Relation: eclipse_files_2.0-weka.filters.unsupervised.attribute.Numeric...
Instances: 2220 Attributes: 200

Attributes

All | None | Inv...

No.	Name
180	NORM_BlockComment
181	NORM_TagElement
182	NORM_TextElement
183	NORM_MemberRef
184	NORM_MethodRef
185	NORM_MethodRefParameter
186	NORM_EnhancedForStatement
187	NORM_EnumDeclaration
188	NORM_EnumConstantDeclaration
189	NORM_TypeParameter
190	NORM_ParameterizedType
191	NORM_QualifiedType
192	NORM_WildcardType
193	NORM_NormalAnnotation
194	NORM_MarkerAnnotation
195	NORM_SingleMemberAnnotation
196	NORM_MemberValuePair
197	NORM_AnnotationTypeDeclaration
198	NORM_AnnotationTypeMemberDeclaration
199	NORM_Modifier
200	post

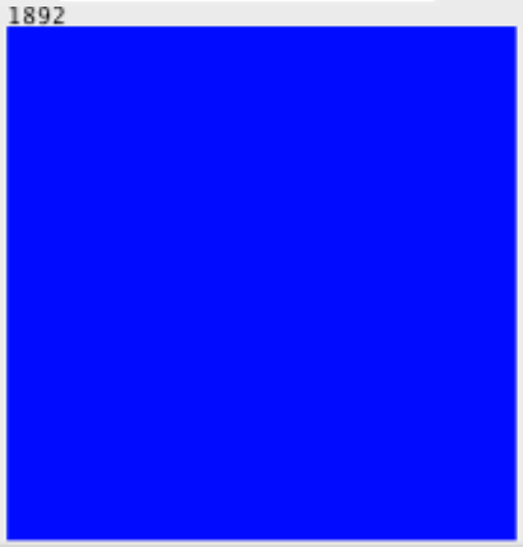
Remove

Selected attribute

Name: post Type: Nominal
Missing: 0 (0%) Distinct: 2 Unique: 0 (0%)

No.	Label	Count
1	0	1892
2	1	328

Visualize All



Weka GUI Chooser

Program Visualization Tools Help

WEKA
The University of Waikato

Waikato Environment for Knowledge Analysis
Version 3.6.4
(c) 1999 - 2010
The University of Waikato
Hamilton, New Zealand

Applications

Explorer
Experimenter
KnowledgeFlow
Simple CLI

Status
OK

http://www.cs.waikato.ac.nz/ml/weka/

Log

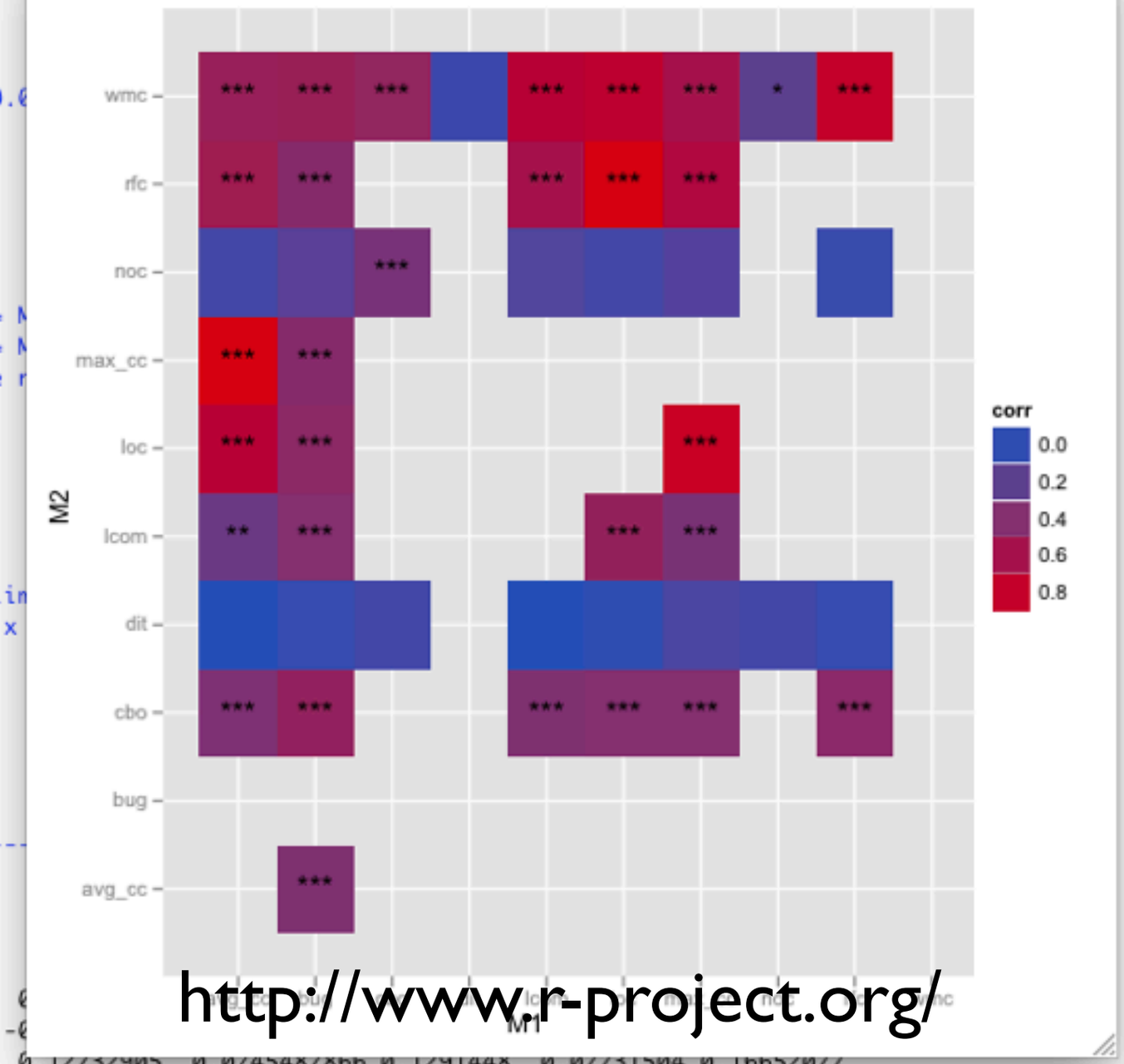
14
x 0

R Console

~/Library/Mail Downloads/Material_Lab5

```
There were 45 warnings (use warnings() to see them)
>
>
> #----Treatment for plotting
> stars <- as.character(symnum(correl_pval, cutpoints=c(0,0.001,0.0
+                       symbols=c('***', '**', '*', '' ),
+                       legend=F))
> molten.log4j10_CK <- cbind(melt(correl), stars)
> names(molten.log4j10_CK) <- c("M1", "M2", "corr", "pvalue")
> x <- ggplot(molten.log4j10_CK, aes(M1, M2, fill=corr))
> mi.ids <- subset(molten.log4j10_CK, M1 == M2)
> mi.lower <- subset(molten.log4j10_CK[lower.tri(correl),], M1 != M
> mi.upper <- subset(molten.log4j10_CK[upper.tri(correl),], M1 != M
> # now plot just these values, adding labels (geom_text) for the r
> (p1 <- x +  geom_tile(data=mi.lower) +
+      geom_text(data=mi.lower, aes(label=paste(pvalue)))
+ )
> meas <- as.character(unique(molten.log4j10_CK$M2))
> pdf(file="correlations.pdf", height=10, width=18)
> (p2 <- p1 + scale_colour_identity() +
+      scale_fill_gradientn(colours= c("red", "white", "blue"), lim
+      scale_x_discrete(limits=meas[length(meas):1]) + #flip the x
+      scale_y_discrete(limits=meas)
+ )
>
> dev.off() #clear plot
quartz
      2
> #------------------------------------------------------------------
>
> #display simple correlation matrix
> correl
              wmc          dit         noc         cbo         rfc
wmc     1.00000000  0.0373988235 0.19991274 0.49237314 0.79230411  0
dit     0.03739882  1.0000000000 0.05963903 0.05441939 0.01590755 -0
noc     0.19991274  0.0596390318 1.00000000 0.34249165 0.03416653  0.12732905  0.0745487866 0.1291448  0.07731504 0.16652077
cbo     0.49237314  0.0544193930 0.34249165 1.00000000 0.43961550  0.38417545  0.3994276803 0.4021495  0.37451605 0.50059149
rfc     0.79230411  0.0159075513 0.03416653 0.43961550 1.00000000  0.60240308  0.9140018081 0.6705735  0.58051895 0.42646834
lcom    0.73483385 -0.0383181799 0.12732905 0.38417545 0.60240308  1.00000000  0.5088939760 0.3540510  0.26698940 0.40043199
loc     0.77376127 -0.0008793353 0.07454879 0.39942768 0.91400181  0.50889398  1.0000000000 0.8226454  0.73859337 0.45236210
max_cc  0.60167207  0.1058438914 0.12914484 0.40214949 0.67057345  0.35405096  0.8226453902 1.0000000  0.92337862 0.42409957
avg_cc  0.53037724 -0.0333186054 0.07731504 0.37451605 0.58051895  0.26698940  0.7385933696 0.9233786  1.00000000 0.38397107
bug     0.53780258  0.0149125524 0.16652077 0.50059149 0.42646834  0.40043199  0.4523621018 0.4240996  0.38397107 1.00000000
>
```

Quartz 2 [*]



http://www.r-project.org/

zeroR, an Educated Guess

Can we Predict the Popularity of a Song?

Favorite Curl

Only Time

Leave Me

One More Lonely Girl

Bobby

Cry

I Smile

Never Let you Enter

Somebody to Move

Sea Sick

Trust the Majority

One

```
@attribute title string
@attribute style {up-tempo,ballad,techno,schlager,camp}
@attribute nr_yeahs numeric
@attribute nr_profanity numeric
...
```

nominal

attributes

```
@attribute success {no,yes}
```

class attribute

```
"Favorite Curl",up-tempo,6,34,...,yes
"Only Time",ballad,34,5,...,yes
"Leave Me",techno,1,1,...,yes
"One More Lonely Girl",up-tempo,0,4,...,yes
"Bobby",up-tempo,3,2,...,yes
"Cry",ballad,2,0,...,yes
"I smile",schlager,0,0,...,yes
"Never Let you Enter",techno,0,4,...,no
"Somebody to Move",up-tempo,3,3,...,no
"Sea Sick",camp,1,6,...,no
```

instances

Classification Process

Favorite Curl
Only Time
Leave Me
One More Lonely Girl
Bobby
Cry
I Smile

Never Let you Enter
Somebody to Move
Sea Sick

training set

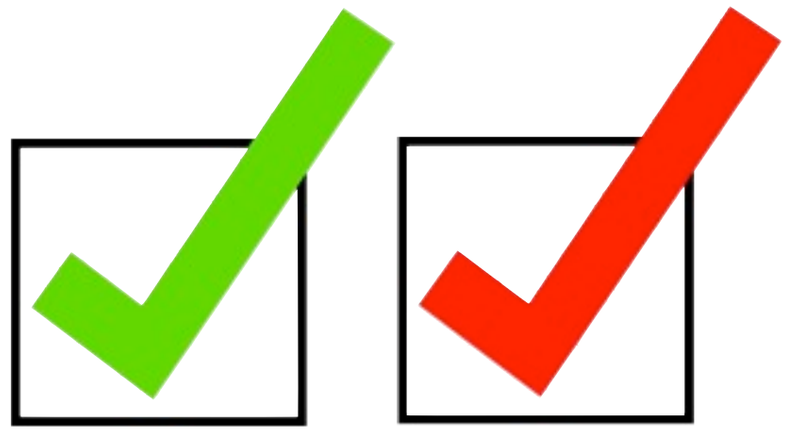
1. train model

model

Never Sing, Never
Your World

test set

2. evaluate model



Applying your Model

Favorite Curl
Only Time
Leave Me
One More Lonely Girl
Bobby
Cry
I Smile
Your World

Never Let you Enter
Somebody to Move
Sea Sick
Never Sing, Never

What patterns exist in our **current** data set?

explanation

New Song
Another New Song

Can we classify **future** instances?

Britney's New Song
Britney's Other New Song

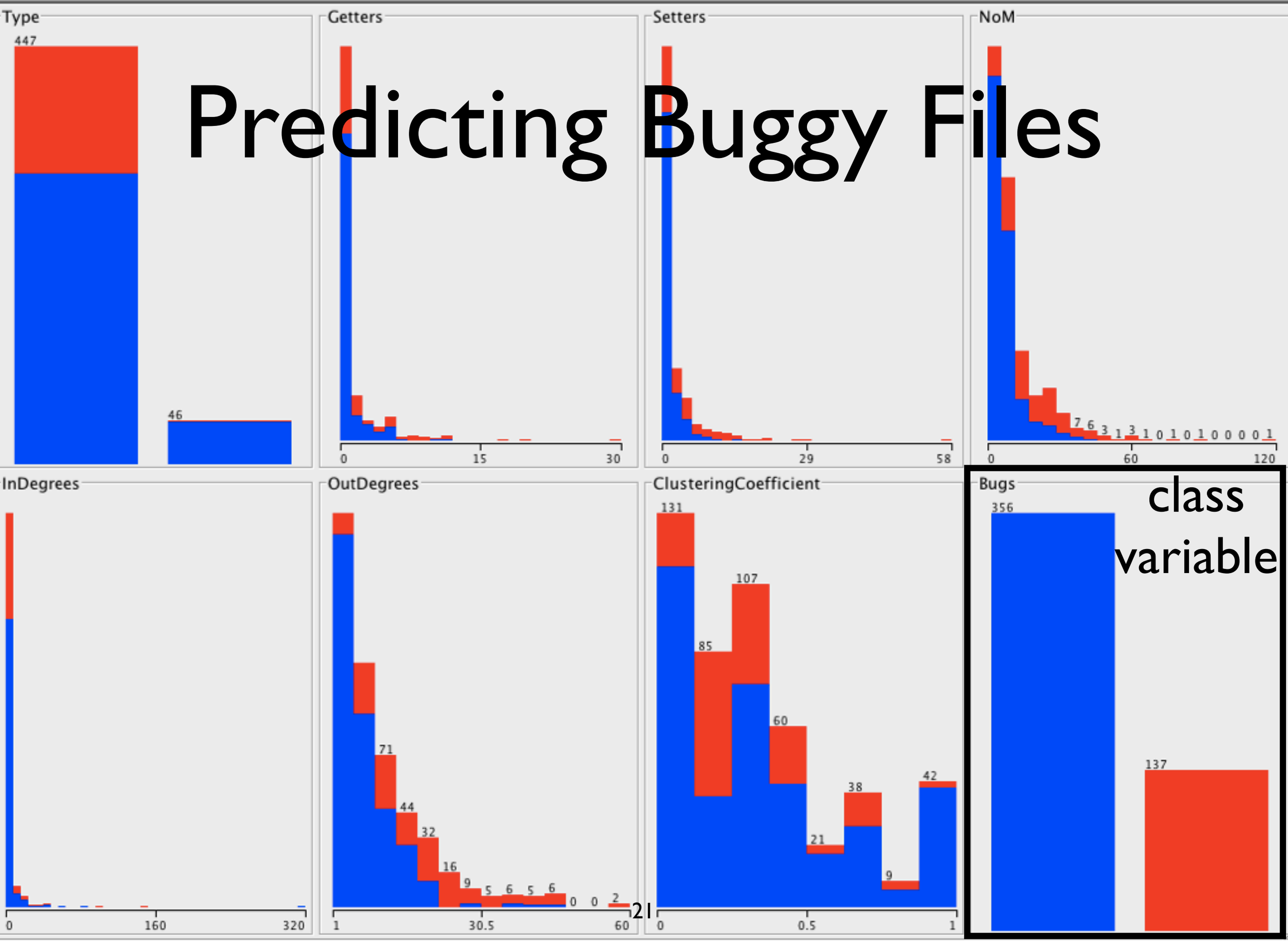
Can we classify **other** data sets?

prediction

A More Realistic Data Set

http://promisedata.org/repository/data/popularity/ant/

Predicting Buggy Files



Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

pick evaluation strategy

Classifier

Choose ZeroR

Test options

- Use training set
- Supplied test set — Set...
- Cross-validation — Folds 10
- Percentage split — % 66

More options...

(Nom) Bugs

Start | Stop

Result list (right-click for options)

22:29:38 - rules.ZeroR

Classifier output

```
Relation:     ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attri
Instances:    493
Attributes:   8
              Type
              Getters
              Setters
              NoM
              InDegrees
              OutDegrees
              ClusteringCoefficient
              Bugs
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

ZeroR predicts class value: 0

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         356               72.211  %
Incorrectly Classified Instances       137               27.789  %
Kappa statistic                          0
Mean absolute error                      0.4018
Root mean squared error                  0.448
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances              493

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         1          0.722     1         0.839      0.489    0
                 0         0          0         0         0          0.489    1
Weighted Avg.    0.722     0.722      0.521     0.722     0.606      0.489

=== Confusion Matrix ===

   a   b   <-- classified as
 356   0 |   a = 0
 137   0 |   b = 1
```

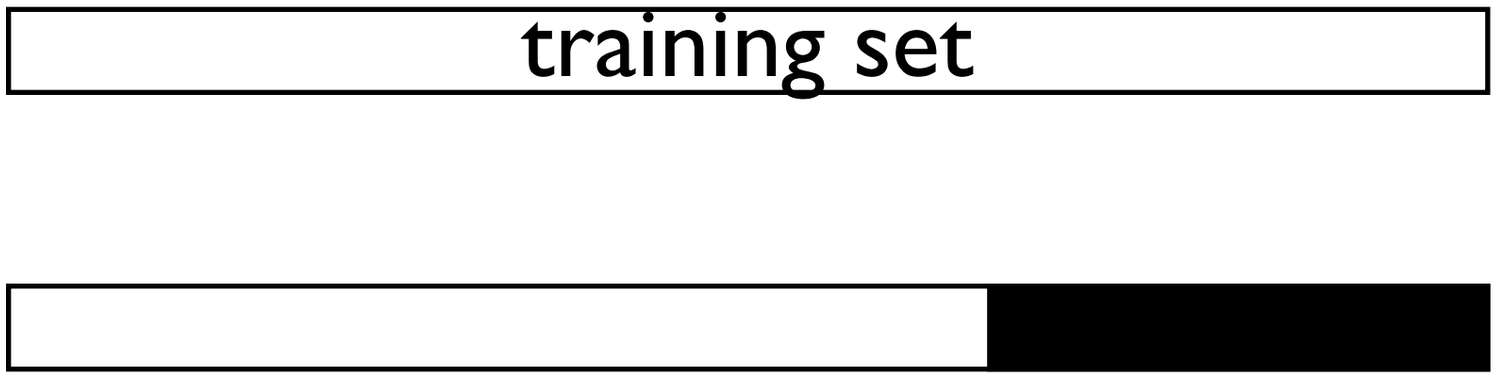
Status

OK

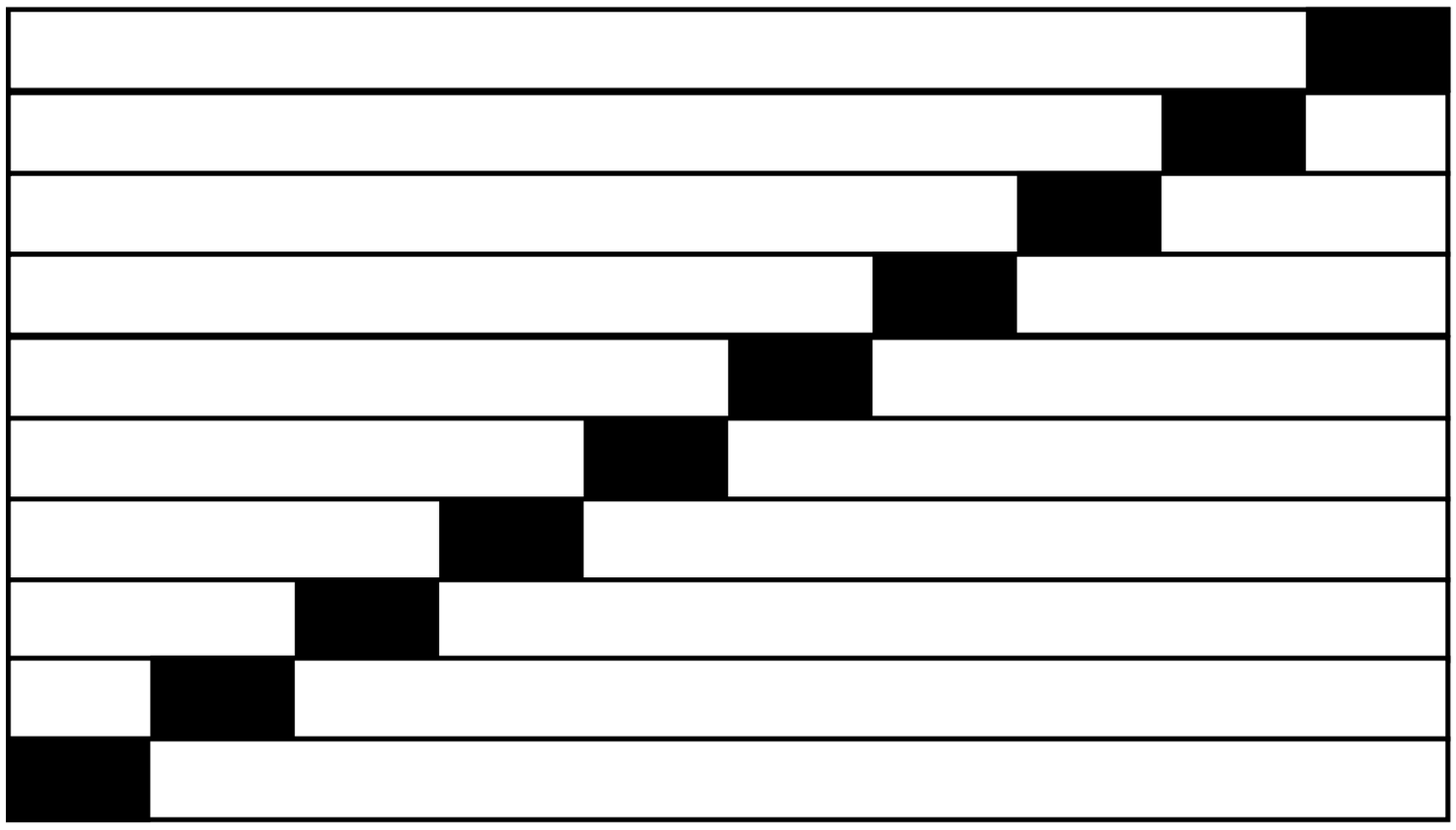
Log

x 0

Cross-validation



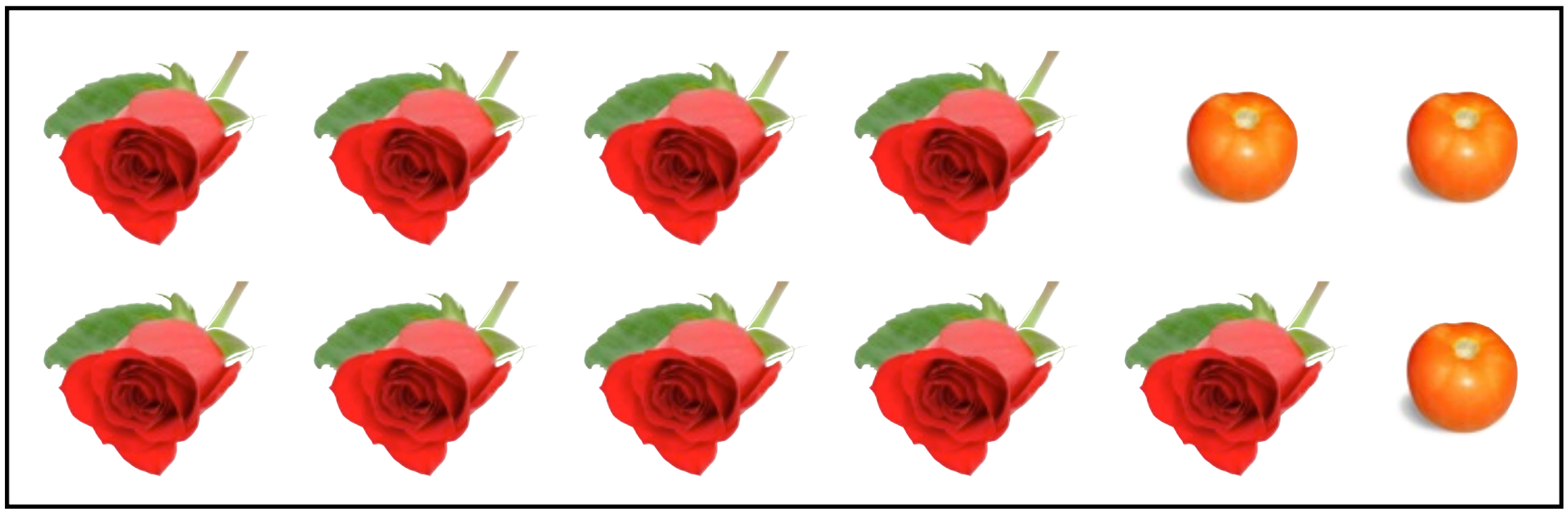
2:1 hold-out



x10

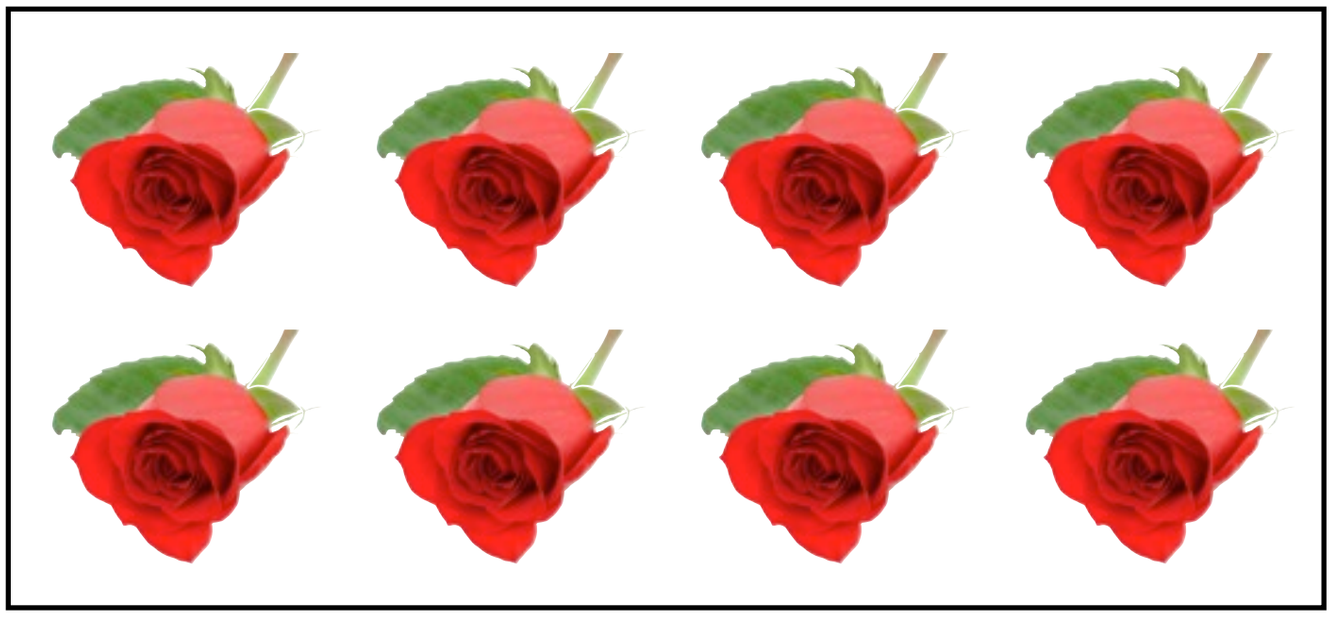
10-fold
cross-validation

Use Representative Sets!

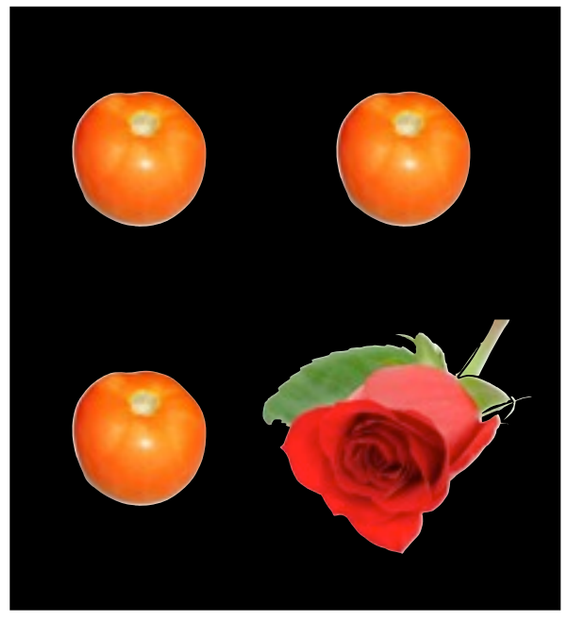


training set

test set



no in training set!

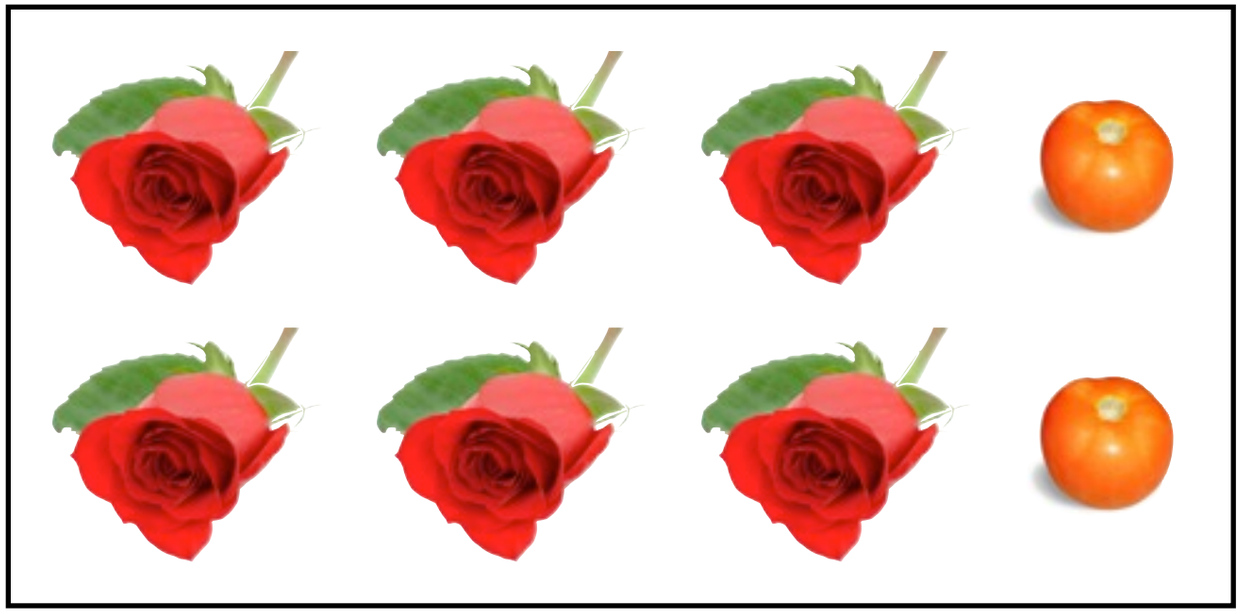


Stratified Cross-validation

3/12

training set

test set



2/8

1/4

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Classifier

Choose **ZeroR**

Test options

- Use training set
- Supplied test set — Set...
- Cross-validation — Folds 10
- Percentage split — % 66

More options...

(Nom) Bugs

Start | Stop

Result list (right-click for options)

22:29:38 - rules.ZeroR

Classifier output

```
Relation:     ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attri
Instances:    493
Attributes:   8
              Type
              Getters
              Setters
              NoM
              InDegrees
              OutDegrees
              ClusteringCoefficient
              Bugs
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

ZeroR predicts class value: 0

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===
```

performance results

```
Correctly Classified Instances         356               72.211  %
Incorrectly Classified Instances       137               27.789  %
Kappa statistic                          0
Mean absolute error                      0.4018
Root mean squared error                  0.448
Relative absolute error                100      %
Root relative squared error            100      %
Total Number of Instances              493

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 1         1          0.722     1         0.839      0.489    0
                 0         0          0         0         0          0.489    1
Weighted Avg.    0.722     0.722      0.521     0.722     0.606      0.489

=== Confusion Matrix ===

   a   b   <-- classified as
 356   0 |   a = 0
 137   0 |   b = 1
```

Status

OK

Log

x 0

```
   a   b   <-- classified as
 356   0 |  a = 0
 137   0 |  b = 1
```

FP_rate(a) = what % of "b" did you falsely classify as "a"?

$$\frac{137}{137+0}$$

```
   a   b   <-- classified as
 356   0 |  a = 0
 137   0 |  b = 1
```

recall(a) = what % of "a" did you correctly classify? = **TP_rate(a)**

$$\frac{356}{356+0}$$

```
   a   b   <-- classified as
 356   0 |  a = 0
 137   0 |  b = 1
```

precision(a) = what % of "a" classifications is correct?

$$\frac{356}{356+137}$$

$$\textbf{F-measure(a)} = 2 \cdot \frac{\text{precision(a)} \cdot \text{recall(a)}}{\text{precision(a)} + \text{recall(a)}}$$

Weka Classifier Visualize: ThresholdCurve. (Class value 0)

X: False Positive Rate (Num) | Y: True Positive Rate (Num)

Colour: Threshold (Num) | Select Instance

Reset | Clear | Open | Save

recall(a) = how **good** is model at identifying this class?

Plot (Area under ROC = 0.4889)

X optimal performance

ROC curve

0.7

0.39

0.41 0.7 1

X
Y

FP_rate(a) = how **bad** is model at discriminating from other class?

Class colour

0.72 0.72 0.72

Weka Classifier Visualize: ThresholdCurve. (Class value 0)

X: False Positive Rate (Num) | Y: True Positive Rate (Num)

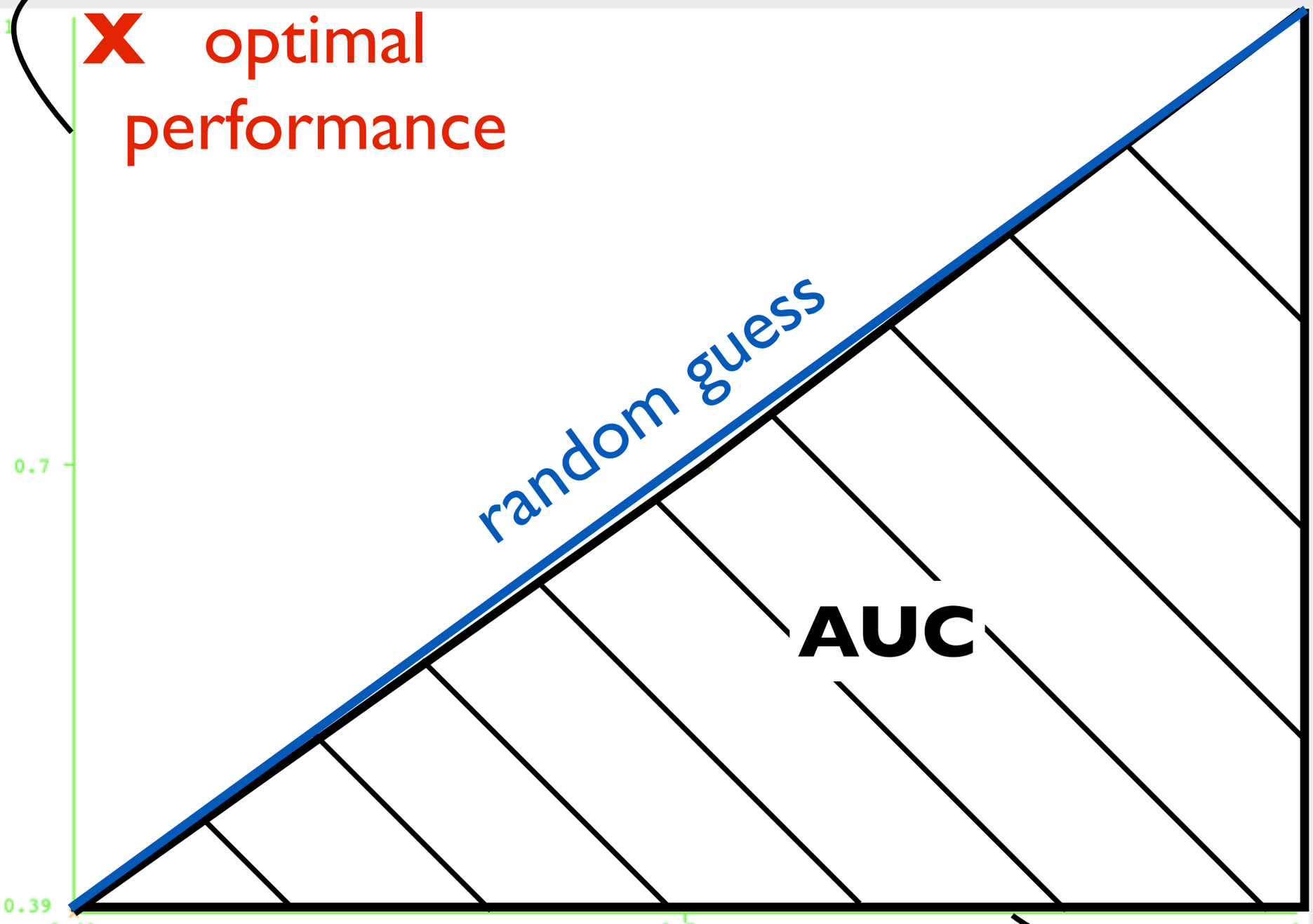
Colour: Threshold (Num) | Select Instance

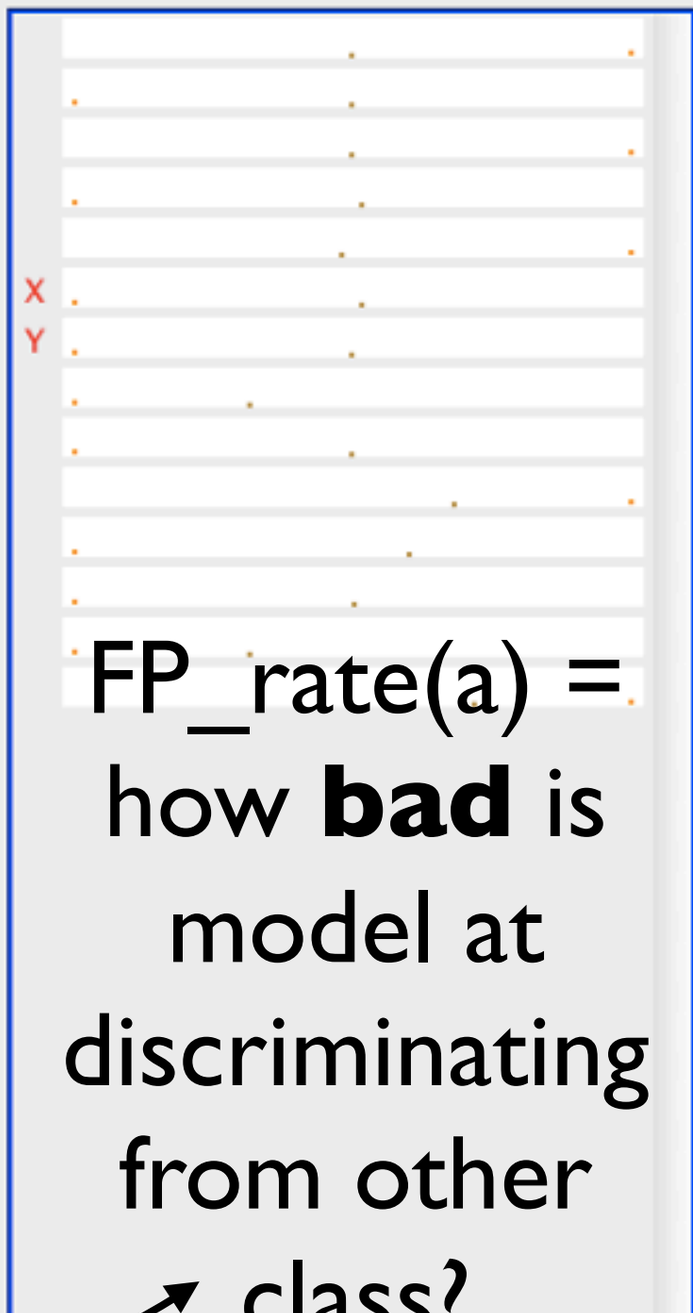
Reset | Clear | Open | Save

recall(a) = how **good** is model at identifying this class?

Plot (Area under ROC = 0.4889)

X optimal performance





FP_rate(a) = how **bad** is model at discriminating from other class?

Class colour

0.72 | 0.72 | 0.72

So, How does zeroR Perform?

```
  a   b   <-- classified as
356   0 |  a = 0
137   0 |  b = 1
```

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
1	1	0.722	1	0.839	0.489	0
0	0	0	0	0	0.489	1
0.722	0.722	0.521	0.722	0.606	0.489	

weighted average

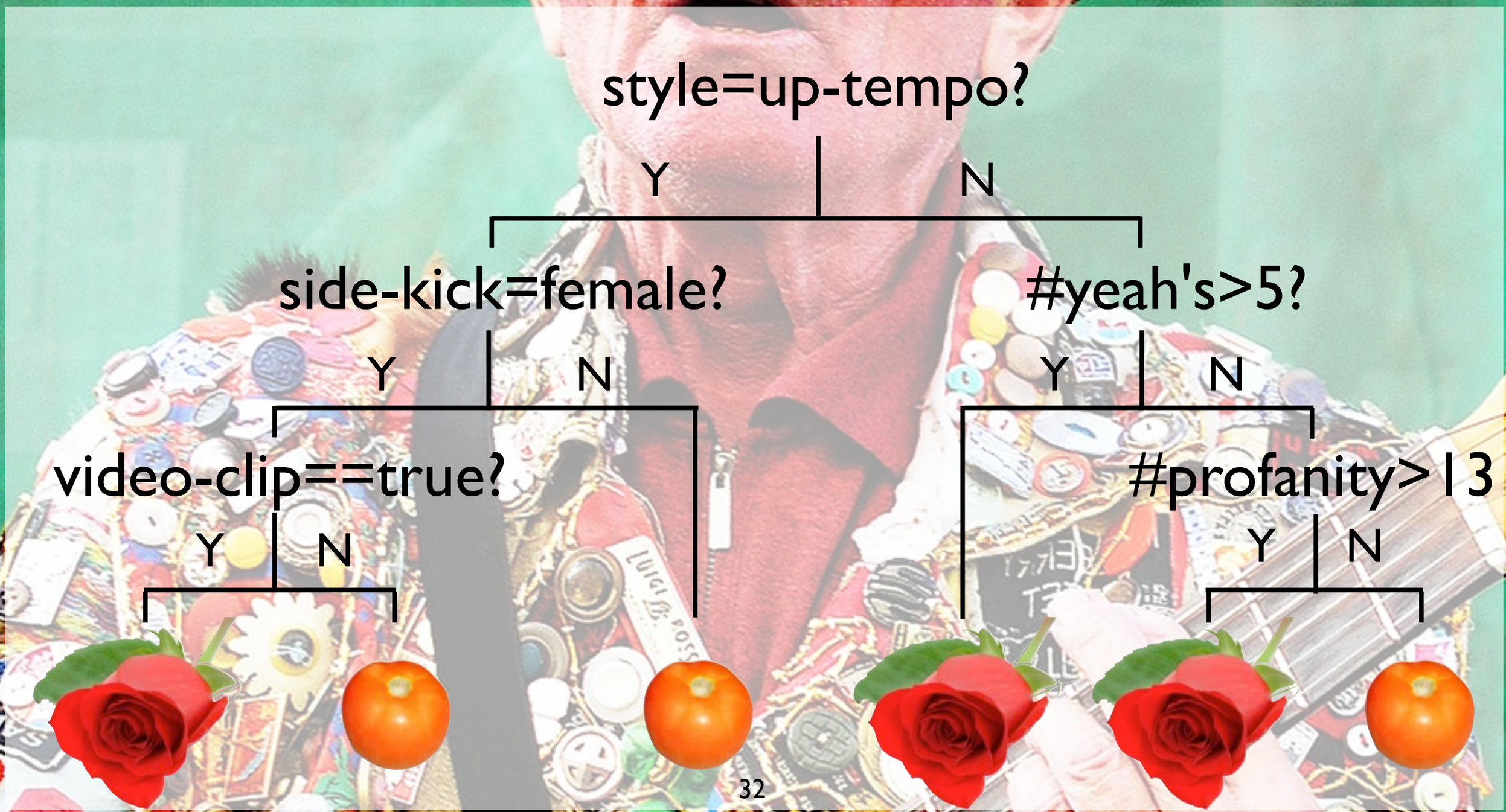
Points in Favour/Against

- extremely simple
- popular baseline

- totally ignores minority class
- ignores all attributes

Decision Trees provide an Explanation

Can we Predict the Popularity of a Song? (2)



Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Classifier

Choose **J48** -C 0.25 -M 2

J48 decision tree learner

Test options

- Use training set
- Supplied test set — Set...
- Cross-validation — Folds 10
- Percentage split — % 66

More options...

(Nom) Bugs

Start | Stop

Result list (right-click for options)

22:29:38 - rules.ZeroR
22:38:34 - trees.J48

Classifier output

```
NoM <= 10: 0 (339.0/39.0)
NoM > 10
|   OutDegrees <= 19
|   |   Getters <= 7
|   |   |   InDegrees <= 8
|   |   |   |   OutDegrees <= 7: 0 (9.0/1.0)
|   |   |   |   OutDegrees > 7: 1 (65.0/28.0)
|   |   |   InDegrees > 8: 0 (14.0/2.0)
|   |   Getters > 7: 1 (8.0/1.0)
|   OutDegrees > 19: 1 (58.0/7.0)

Number of Leaves  :     6

Size of the tree :      11


Time taken to build model: 0.11 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         386               78.2961 %
Incorrectly Classified Instances       107               21.7039 %
Kappa statistic                          0.427
Mean absolute error                      0.285
Root mean squared error                  0.3952
Relative absolute error                 70.9308 %
Root relative squared error             88.2108 %
Total Number of Instances              493

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.885     0.482      0.827     0.885     0.855      0.747    0
                 0.518     0.115      0.634     0.518     0.57       0.747    1
Weighted Avg.    0.783     0.38       0.773     0.783     0.776      0.747

=== Confusion Matrix ===

   a   b   <-- classified as
 315  41 |   a = 0
  66  71 |   b = 1
```

Status

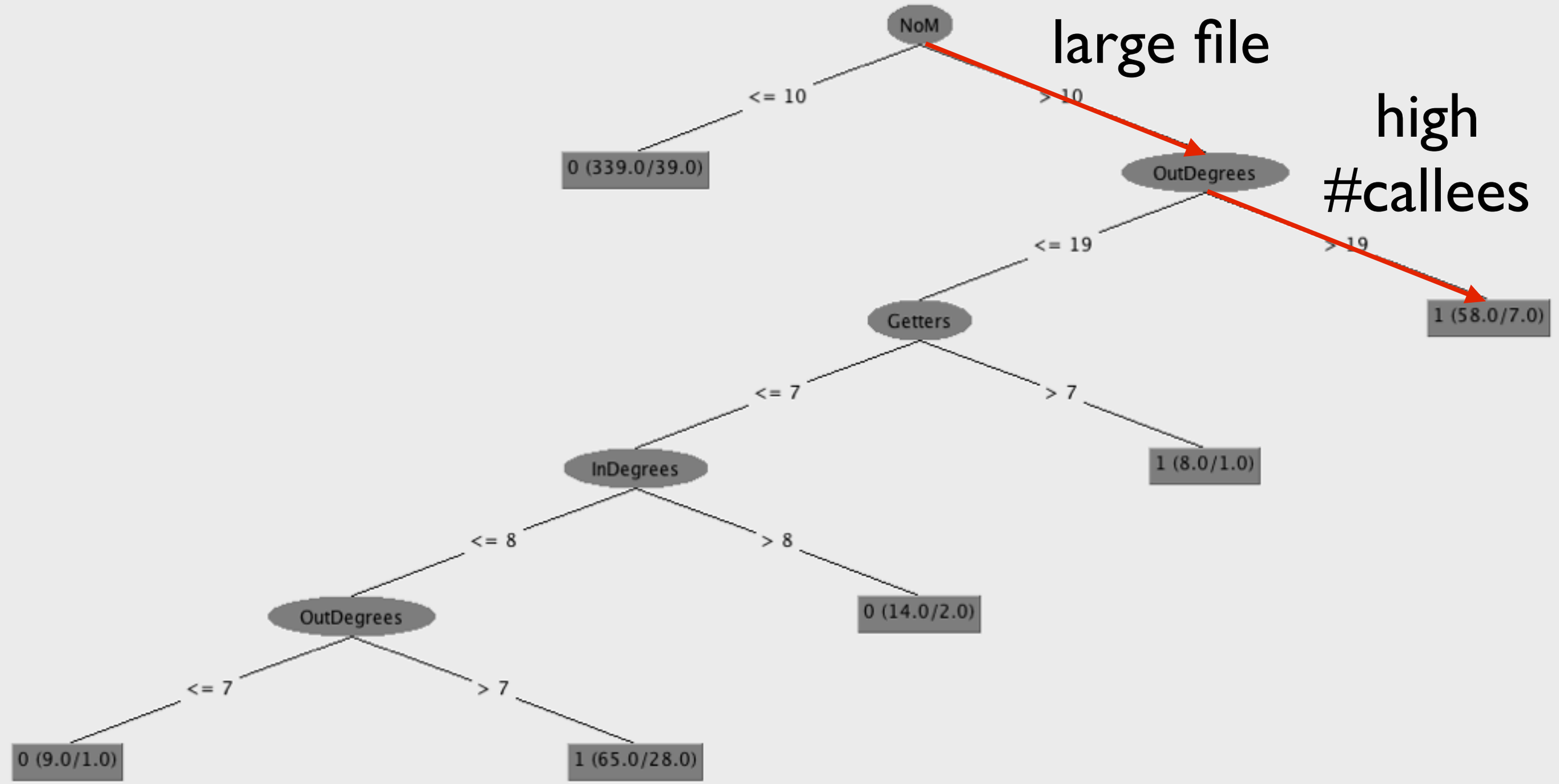
OK

Log

x 0

Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View

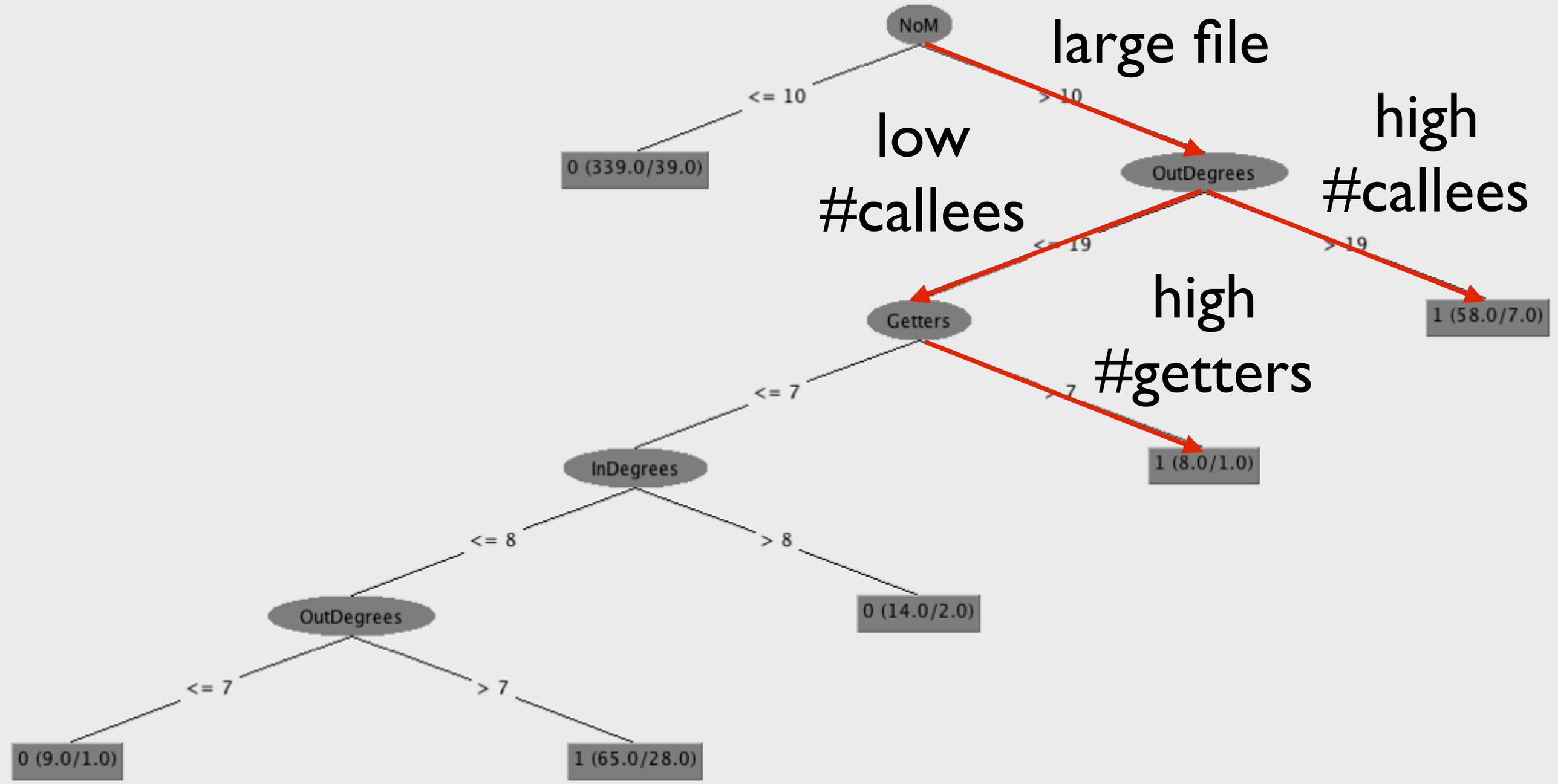


large file

high #callees

Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View



large file

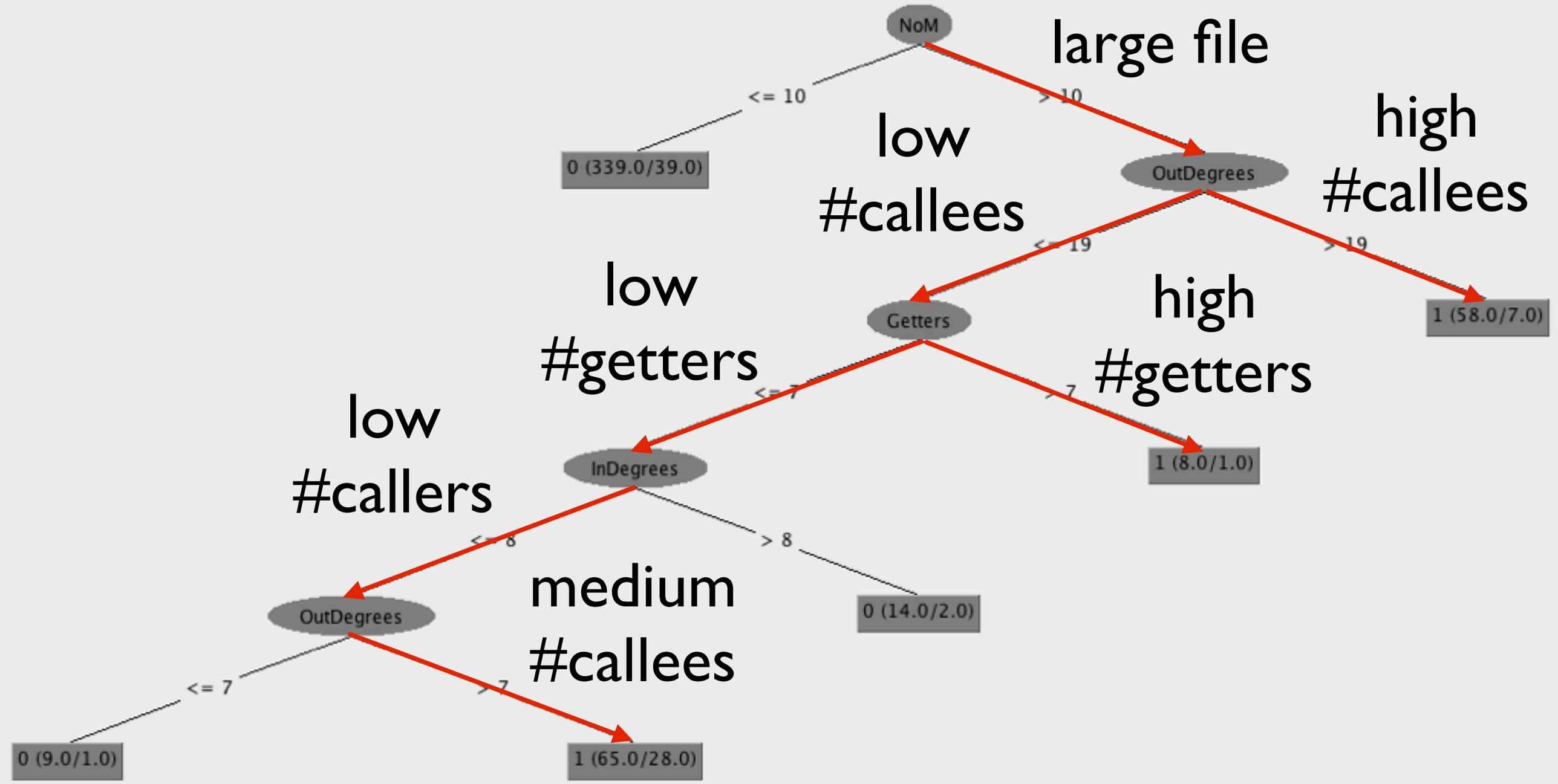
low #callees

high #callees

high #getters

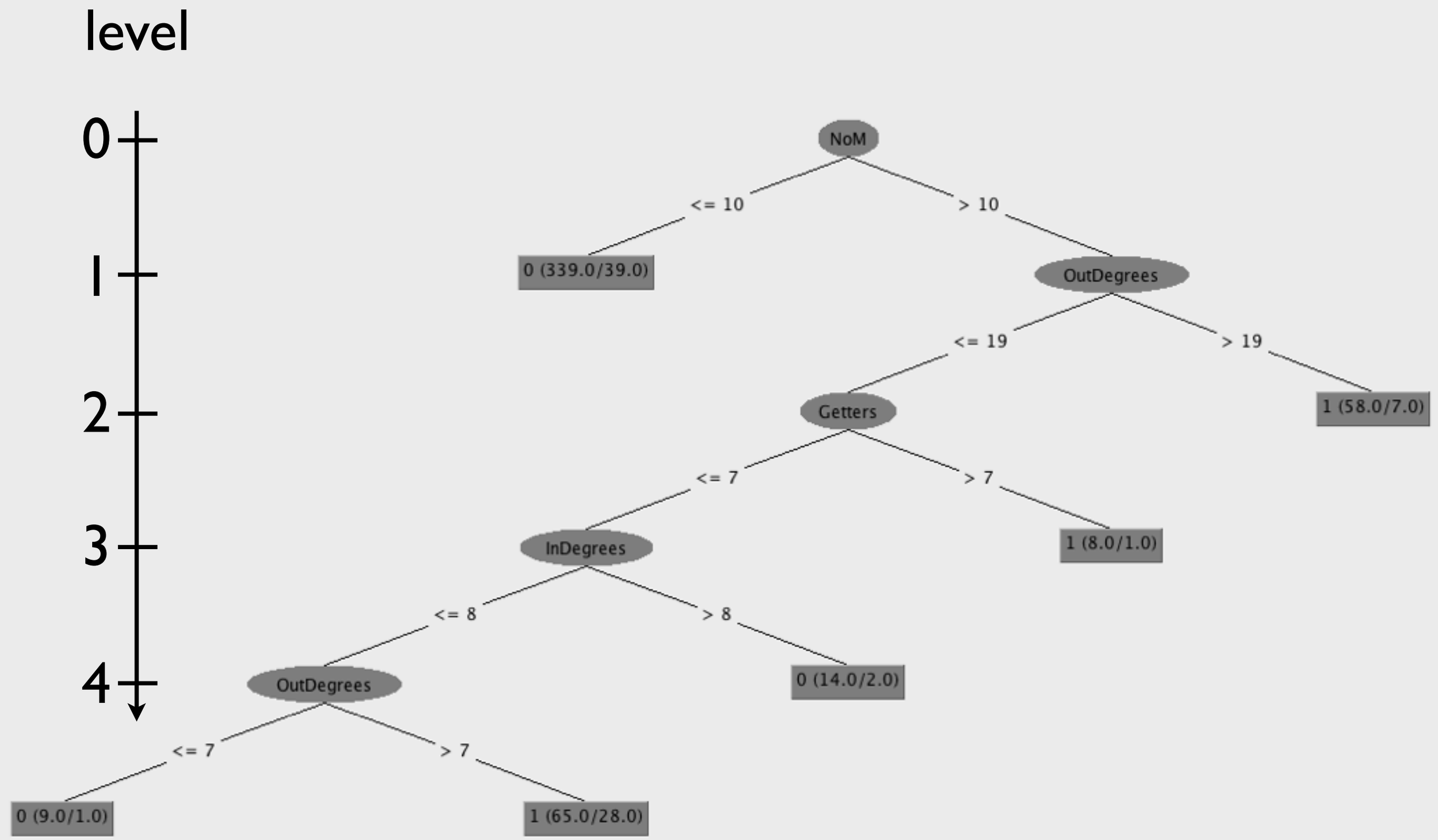
Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View



Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View



Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View

level

0

1

2

3

4

350
262.5
175
87.5
0

N

135
101.25
67.5
33.75
0

B

NoM

<= 10

> 10

0 (339.0/39.0)

339 x "not buggy"

OutDegrees

<= 19

> 19

131 x "buggy"

23 x "not buggy"

Getters

1 (58.0/7.0)

<= 7

> 7

InDegrees

1 (8.0/1.0)

<= 8

> 8

OutDegrees

0 (14.0/2.0)

<= 7

> 7

0 (9.0/1.0)

1 (65.0/28.0)

Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View

level

0

1

2

3

4

350
262.5
175
87.5
0

N

135
101.25
67.5
33.75
0

B

NoM

<= 10

> 10

0 (339.0/39.0)

OutDegrees

<= 19

> 19

1 (58.0/7.0)

Getters

<= 7

> 7

1 (8.0/1.0)

8 x "buggy"

InDegrees

<= 8

> 8

0 (14.0/2.0)

OutDegrees

<= 7

> 7

0 (9.0/1.0)

1 (65.0/28.0)

65 x "buggy"

23 x "not buggy"

65
32.5
0

35

B

10
5
0

B

Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View

level

0

1

2

3

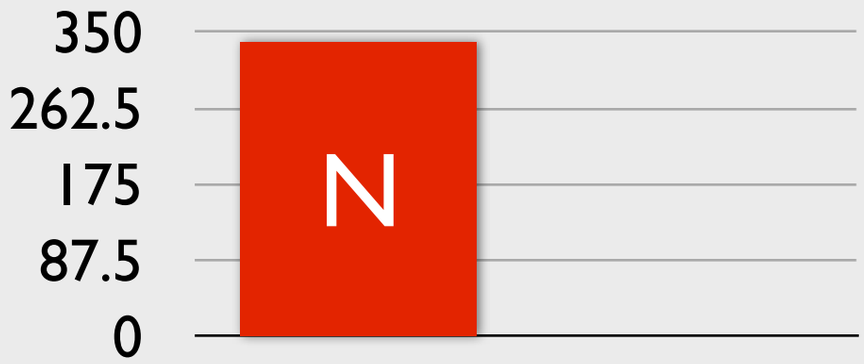
4

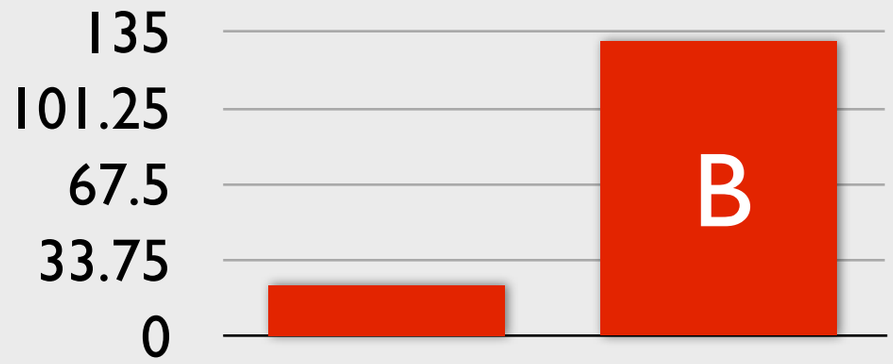
make as many

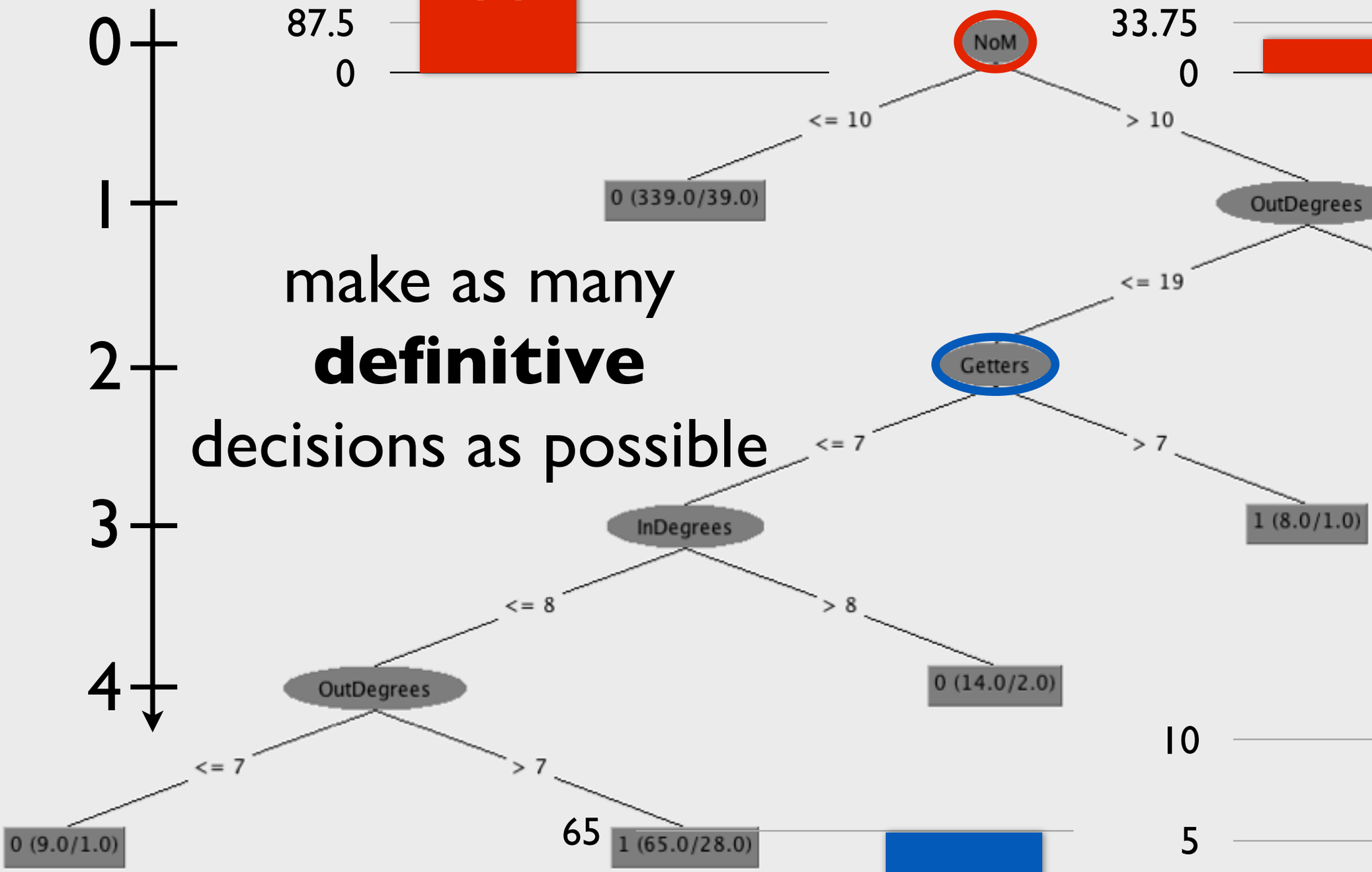
definitive

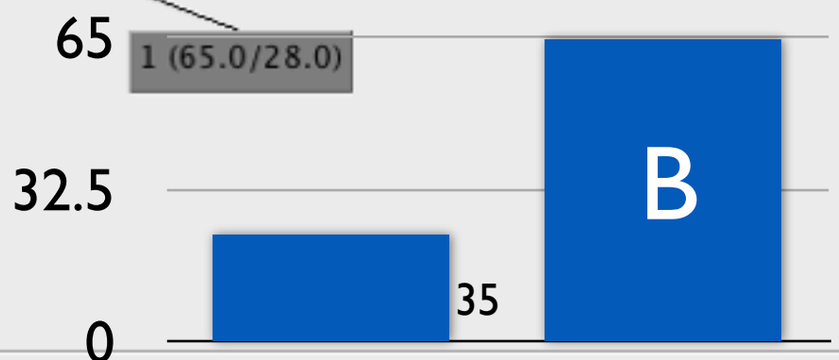
decisions as possible

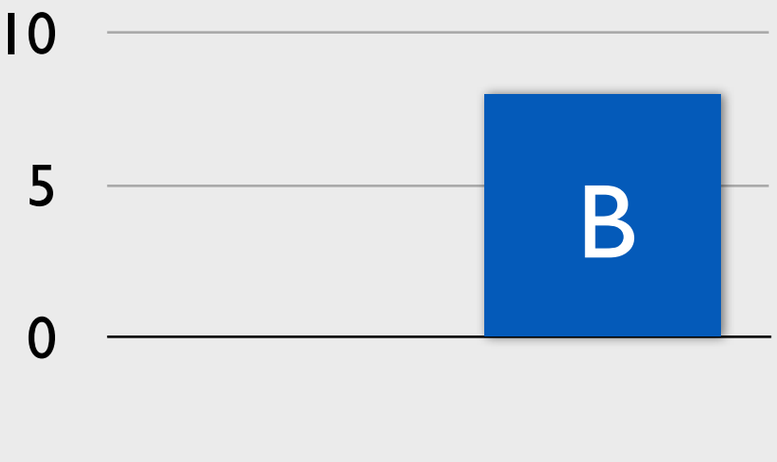
entropy/infogain



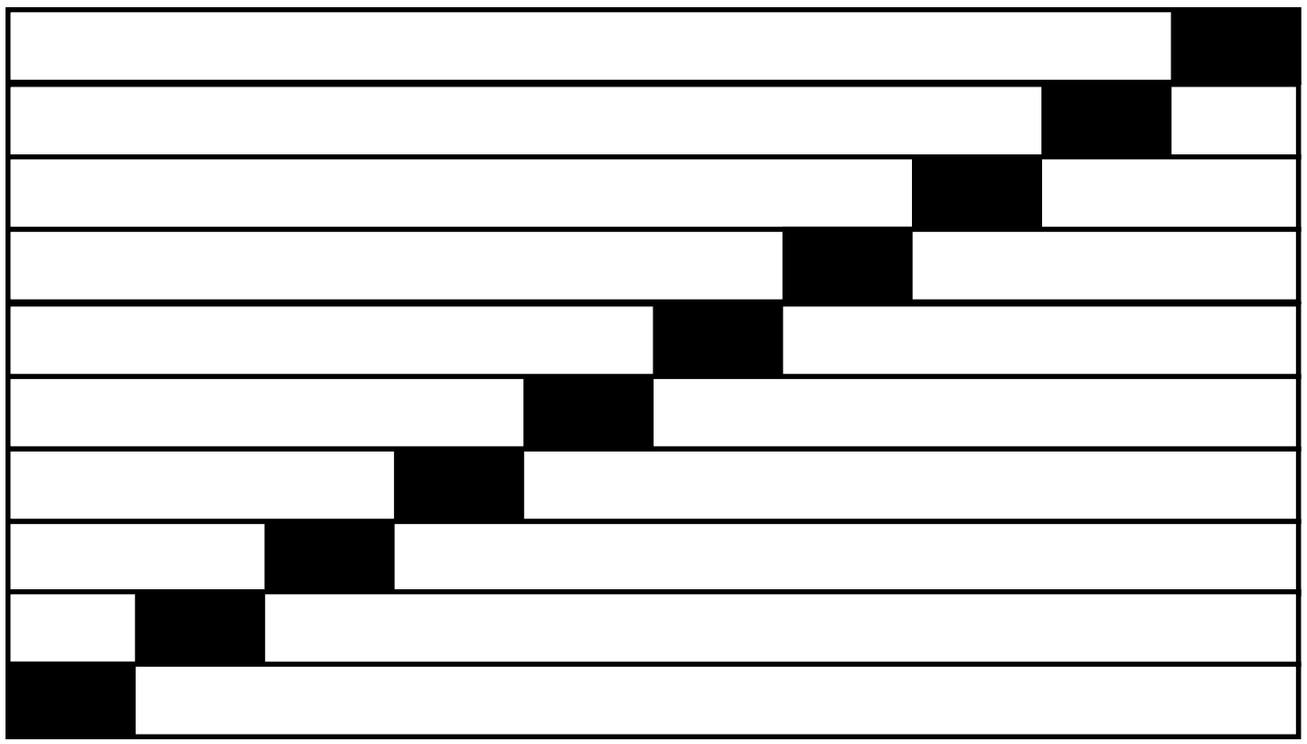








Top Node Analysis



Level 0:
- 7 NoM
- 3 OutDegrees

Level 1:
- 6 OutDegrees
- 1 Setters
- 1 NoM

Level 2:
- 5 Getters
- 2 OutDegrees
- 1 ClusteringCoefficient

NoM and OutDegrees are most influential!

Weka Classifier Visualize: ThresholdCurve. (Class value 0)

X: False Positive Rate (Num)

Y: True Positive Rate (Num)

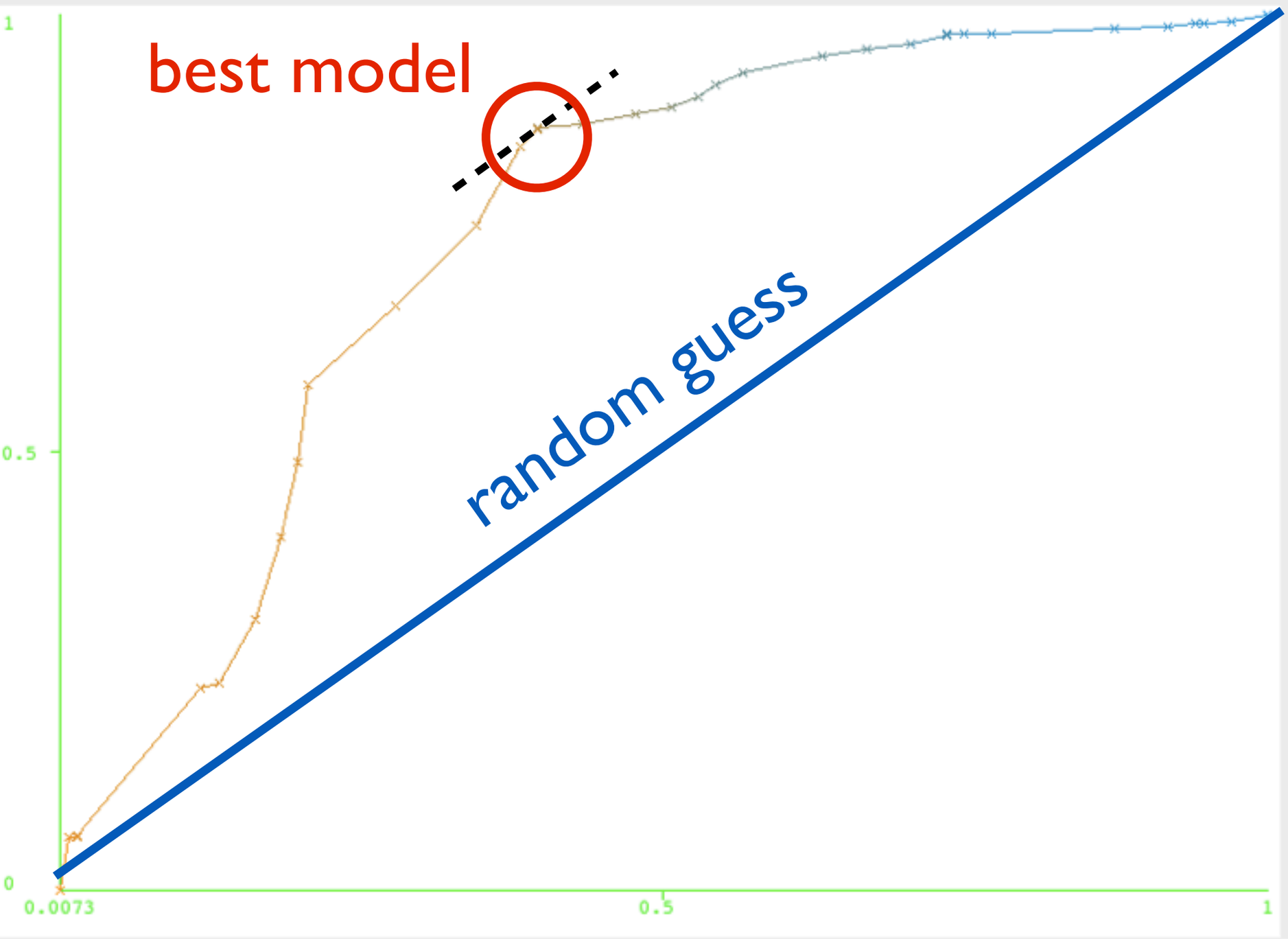
Colour: Threshold (Num)

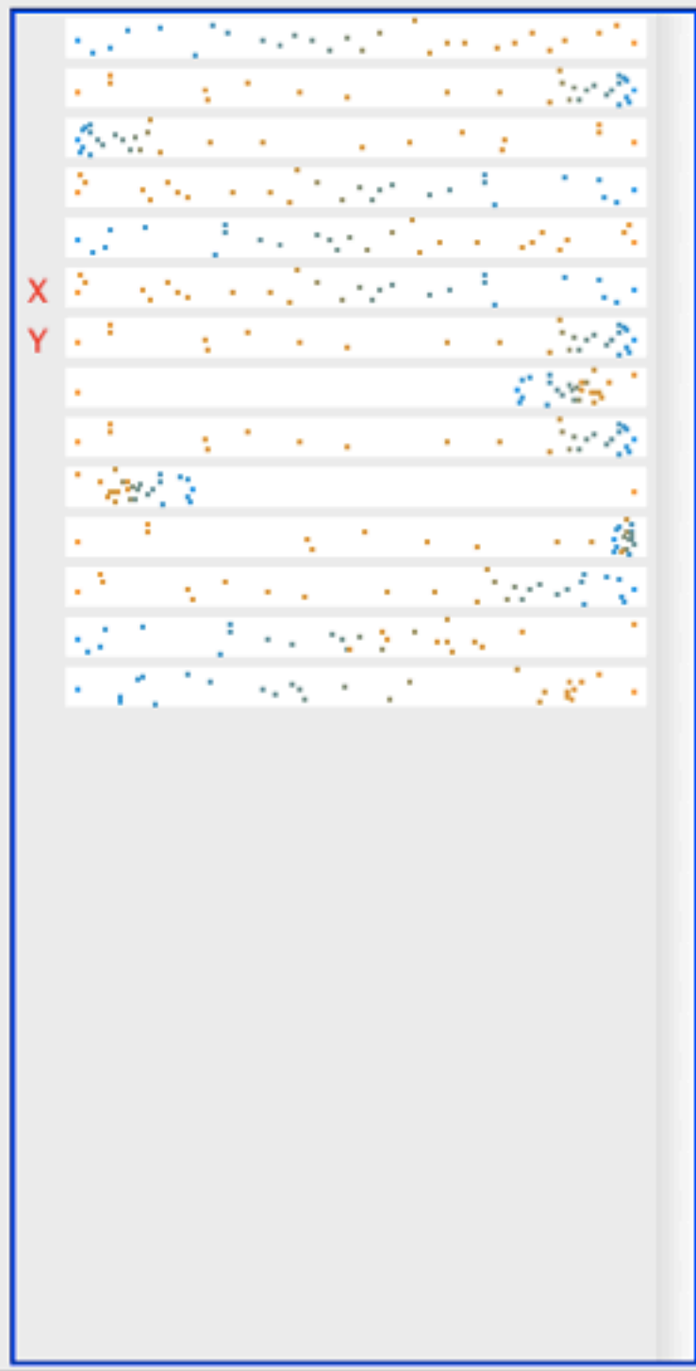
Select Instance

Reset | Clear | Open | Save

Jitter

Plot (Area under ROC = 0.7475)





Class colour

0 0.5 1

37

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
1	1	0.722	1	0.839	0.489	0
0	0	0	0	0	0.489	1
0.722	0.722	0.521	0.722	0.606	0.489	

OR

```
  a   b   <-- classified as
356   0 |  a = 0
137   0 |  b = 1
```

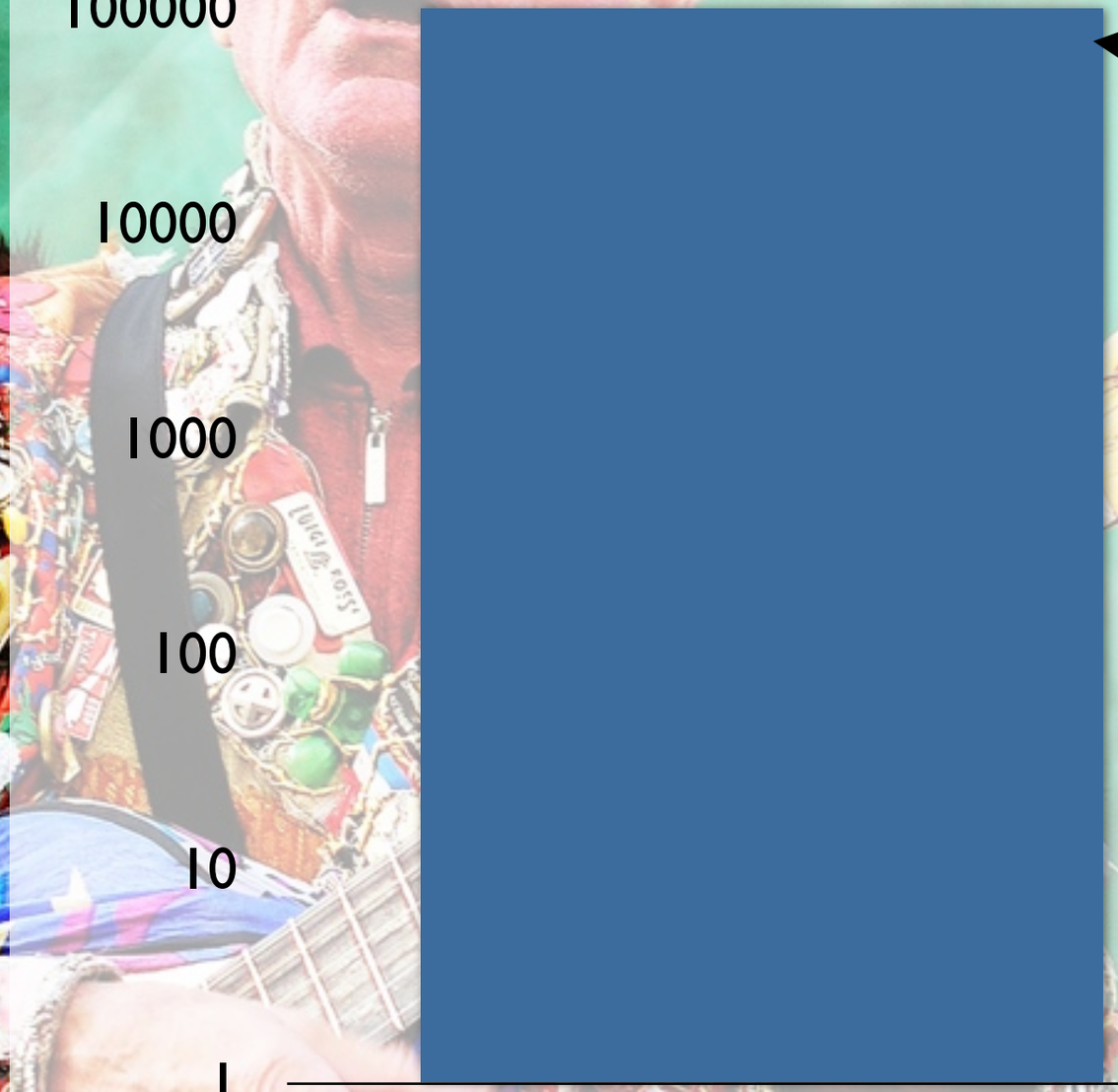
decision trees provide a **richer** model

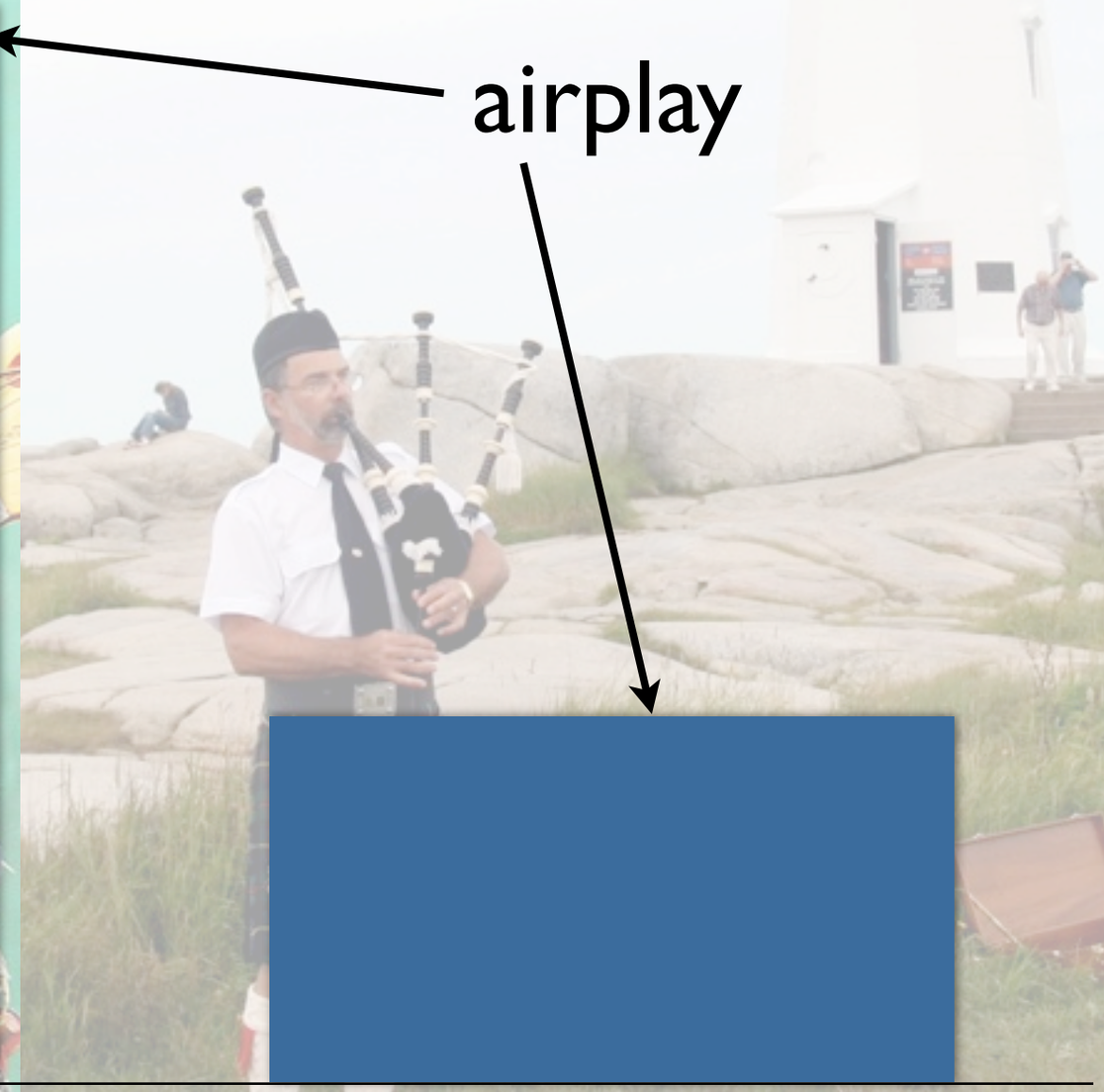
```
  a   b   <-- classified as
315  41 |  a = 0
 66  71 |  b = 1
```

Decision Tree

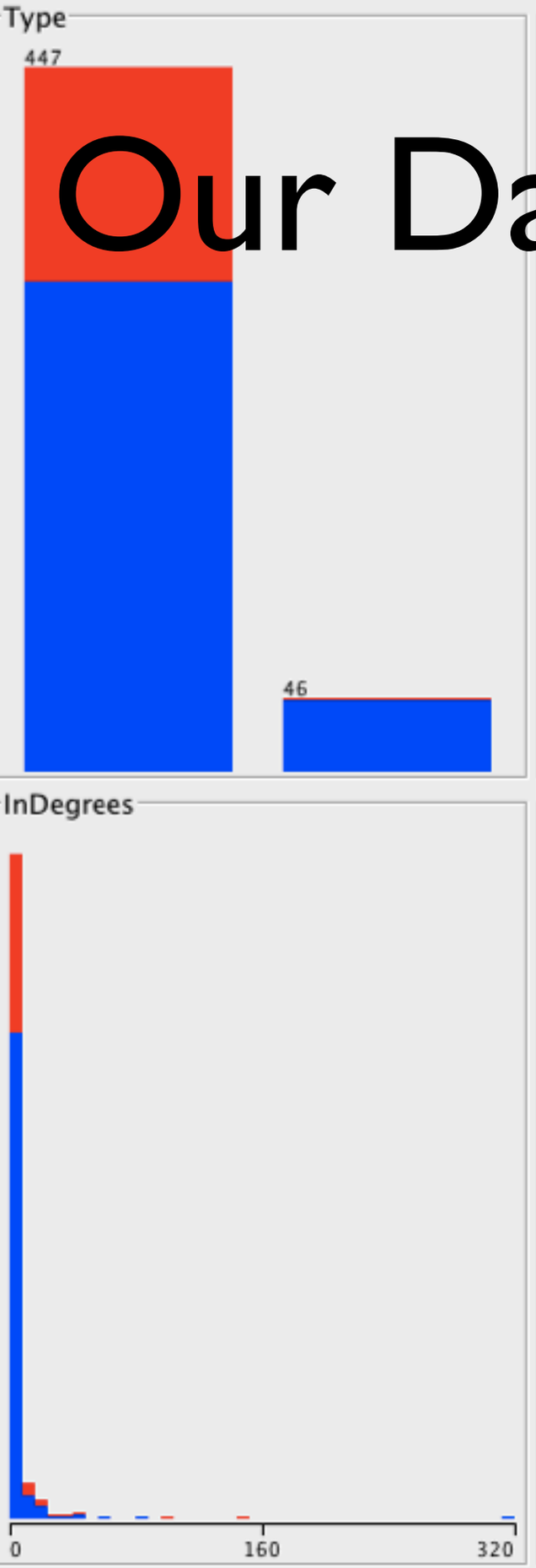
TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.885	0.482	0.827	0.885	0.855	0.747	0
0.518	0.115	0.634	0.518	0.57	0.747	1
0.783	0.38	0.773	0.783	0.776	0.747	

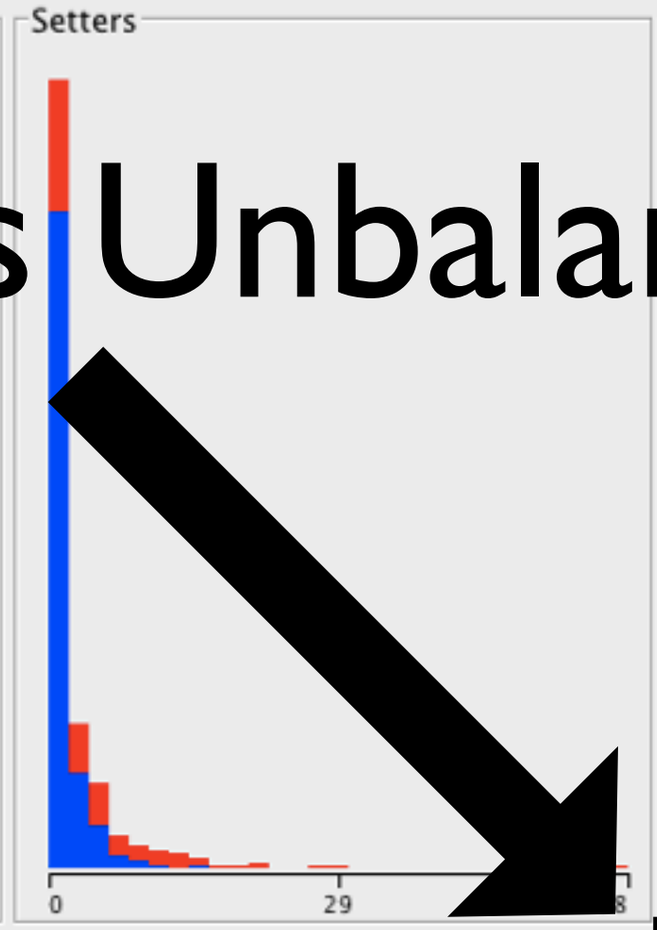
Unbalanced Data Biases the Model!

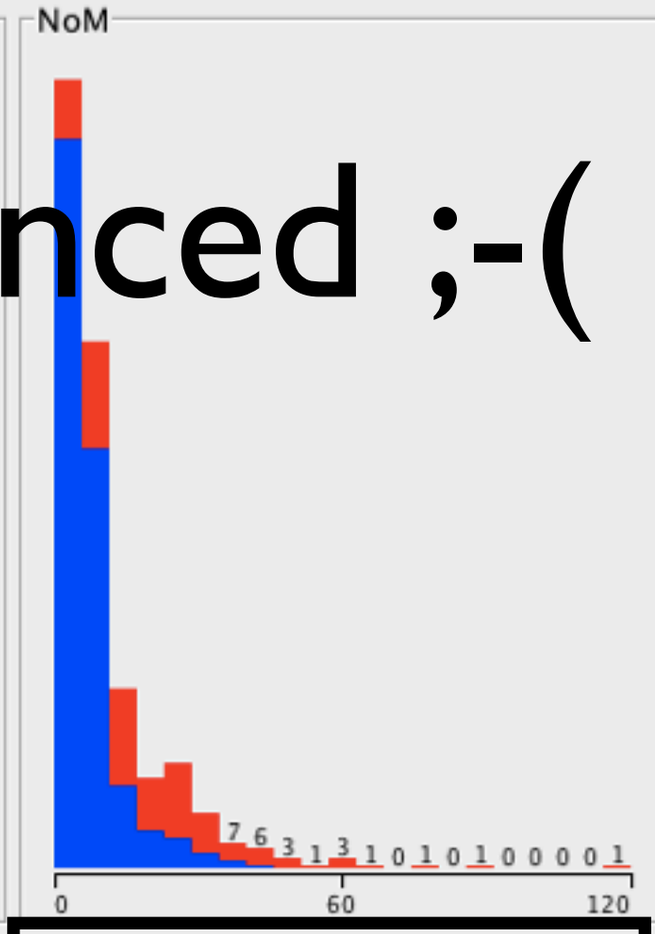


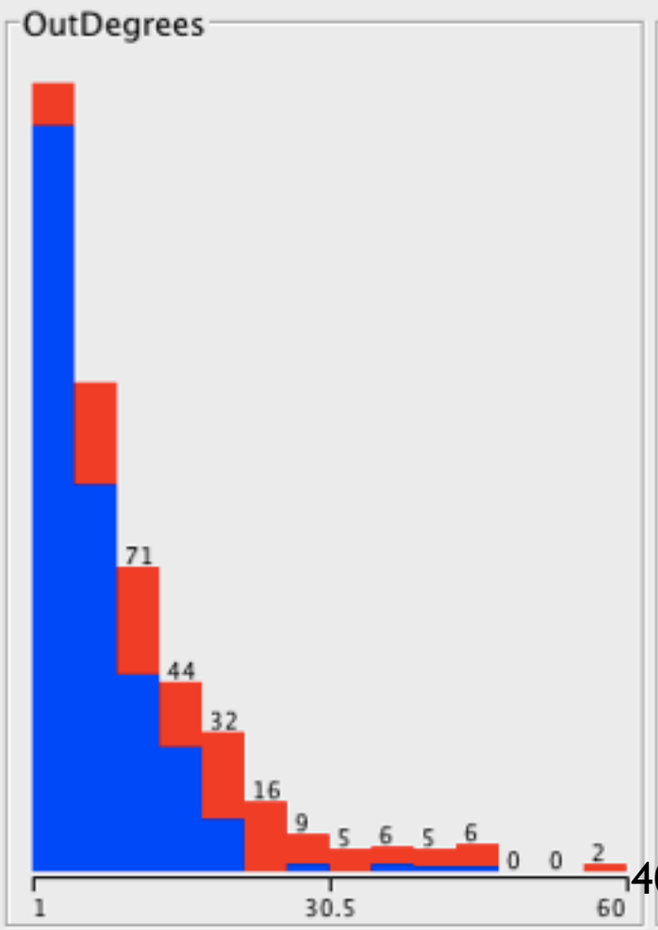


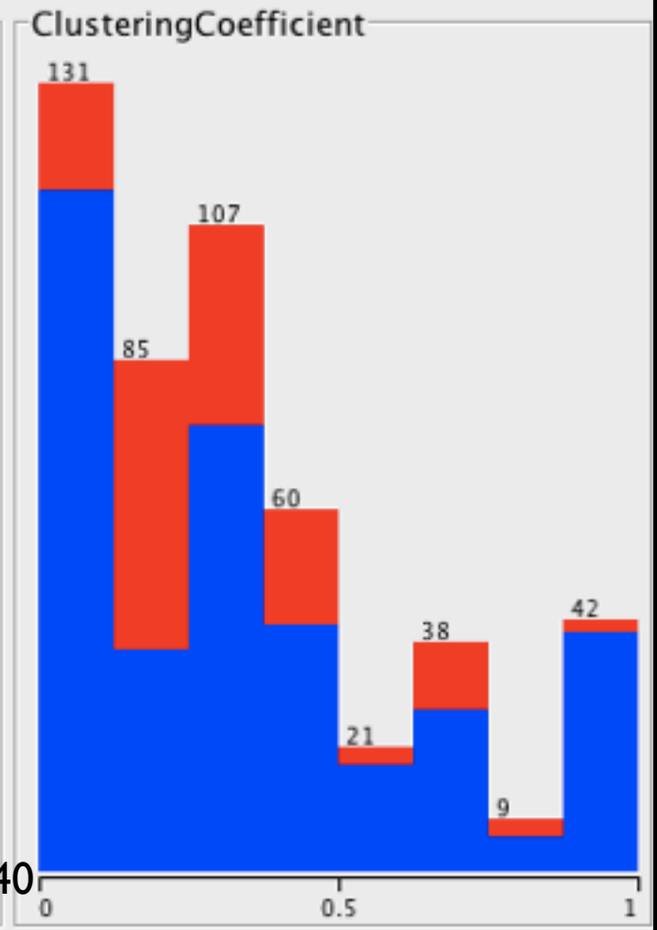
Our Data Set is Unbalanced ;-(

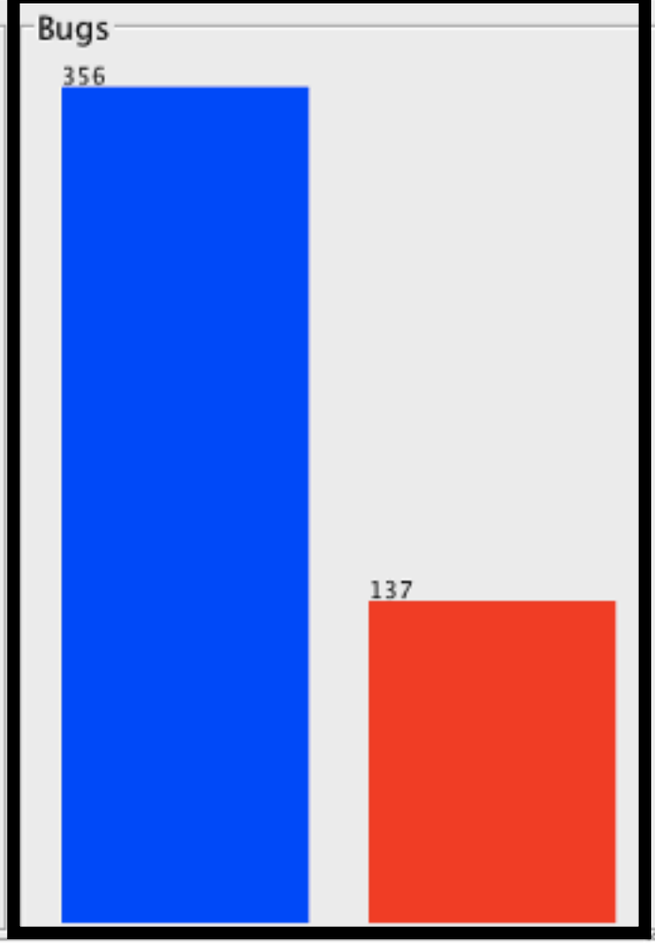




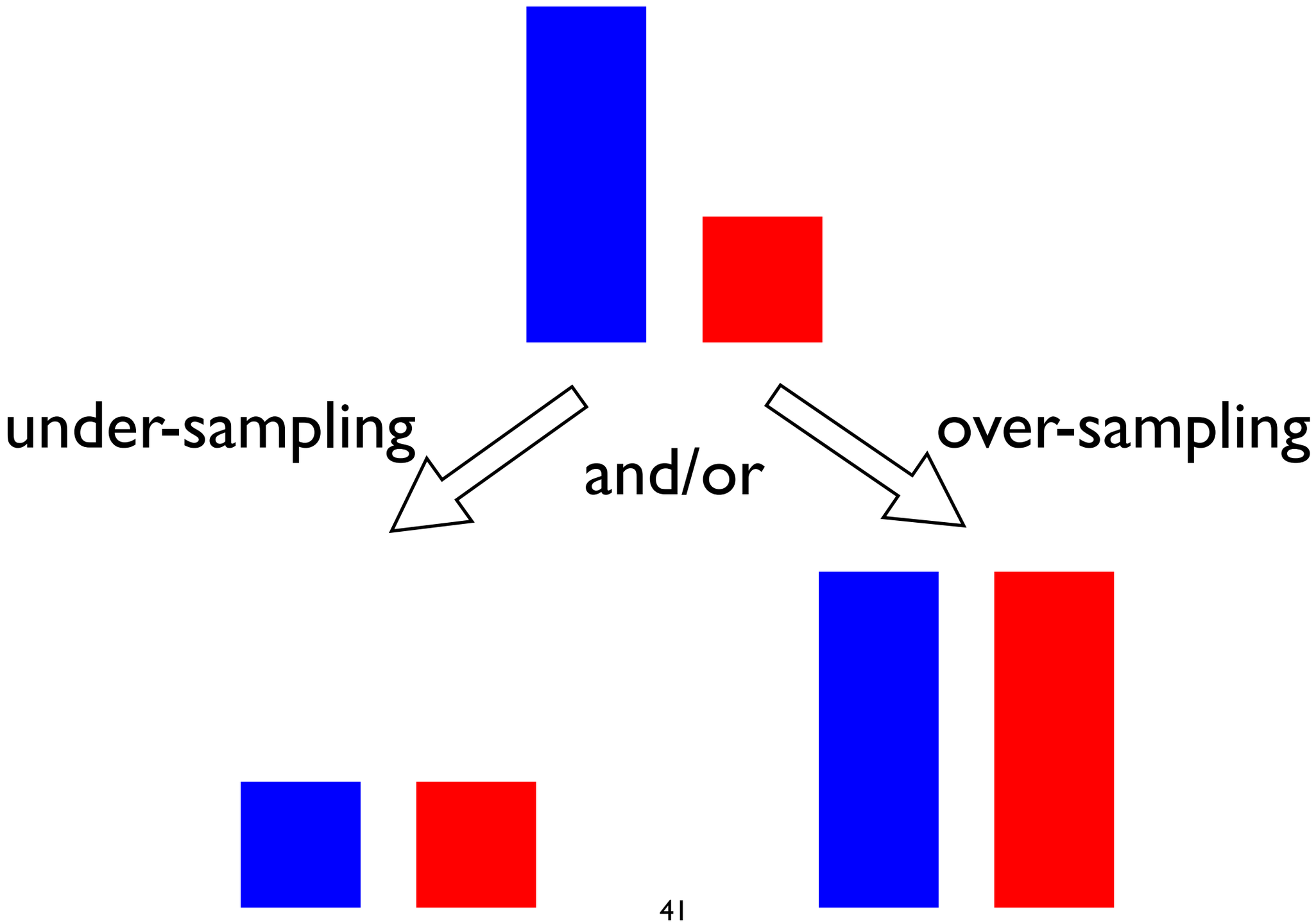








Re-sampling Fixes Unbalance



do **NOT re-sample test set**

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.885	0.482	0.827	0.885	0.855	0.747	0
0.518	0.115	0.634	0.518	0.57	0.747	1
0.783	0.38	0.773	0.783	0.776	0.747	

Decision Tree

```
   a   b   <-- classified as
 315  41 |  a = 0
  66  71 |  b = 1
```

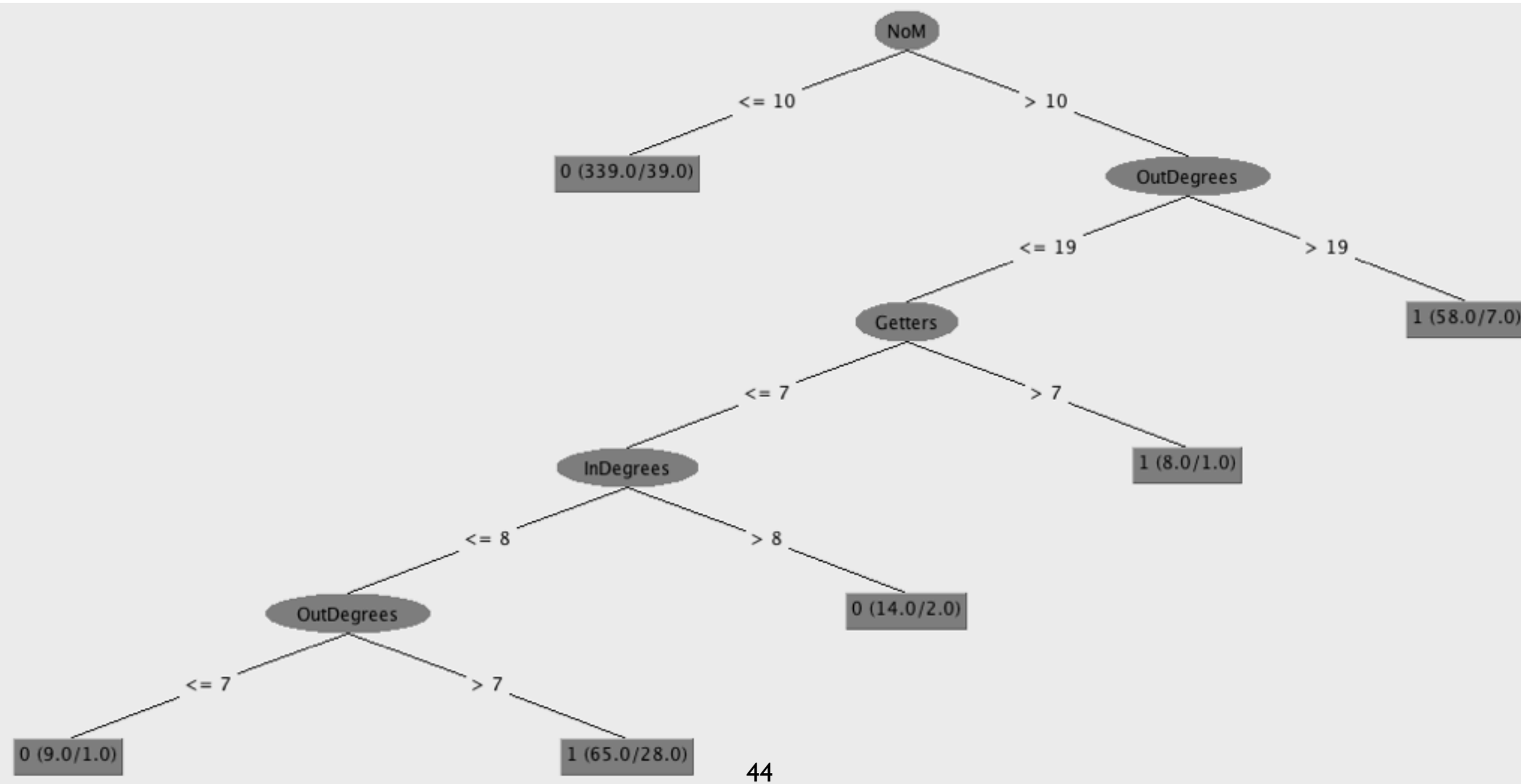
balanced data **improves** minority performance

```
   a   b   <-- classified as
 260  96 |  a = 0
  38  99 |  b = 1
```

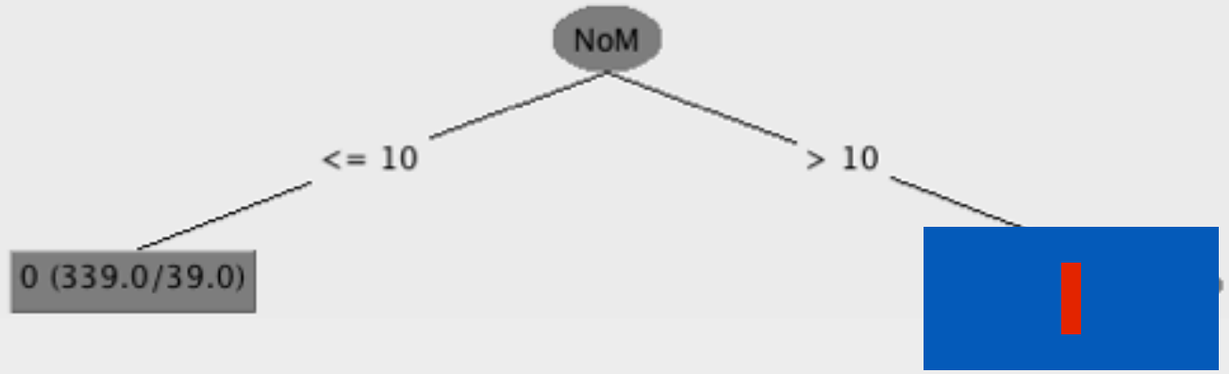
Balanced

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.73	0.277	0.872	0.73	0.795	0.725	0
0.723	0.27	0.508	0.723	0.596	0.725	1
0.728	0.275	0.771	0.728	0.74	0.725	

But, do we Really Need a Complex Model?



But, do we Really Need a Complex Model?



oneR is surprisingly accurate => good baseline

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Classifier

Choose **OneR -B 6**

oneR

Test options

- Use training set
- Supplied test set — Set...
- Cross-validation — Folds 10
- Percentage split — % 66

More options...

(Nom) Bugs

Start | Stop

Result list (right-click for options)

- 17:17:17 - rules.ZeroR
- 17:18:06 - trees.J48
- 17:36:00 - trees.J48
- 17:36:16 - trees.J48
- 17:57:27 - meta.AdaBoostM1
- 17:58:23 - meta.AdaBoostM1
- 19:08:42 - trees.J48
- 19:11:07 - trees.J48
- 19:12:02 - trees.J48
- 20:27:20 - rules.OneR
- 20:29:03 - rules.OneR
- 20:29:23 - rules.OneR

Classifier output

```
               NoM
               InDegrees
               OutDegrees
               ClusteringCoefficient
               Bugs
Test mode:     10-fold cross-validation

=== Classifier model (full training set) ===

NoM:
        < 12.5  -> 0
        < 19.5  -> 1
        < 22.5  -> 0
        >= 22.5 -> 1
(404/493 instances correct)


Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         391               79.3103 %
Incorrectly Classified Instances       102               20.6897 %
Kappa statistic                          0.4499
Mean absolute error                      0.2069
Root mean squared error                  0.4549
Relative absolute error                 51.4933 %
Root relative squared error            101.5357 %
Total Number of Instances              493

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure   ROC Area  Class
                 0.896     0.474      0.831     0.896     0.862      0.711    0
                 0.526     0.104      0.661     0.526     0.585      0.711    1
Weighted Avg.    0.793     0.371      0.783     0.793     0.785      0.711

=== Confusion Matrix ===

   a   b   <-- classified as
 319  37 |   a = 0
  65  72 |   b = 1
```

NoM < 12.5 => 0
NoM < 19.5 => 1
NoM < 22.5 => 0
NoM >= 22.5 => 1

Status

OK

Log x 0

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.885	0.482	0.827	0.885	0.855	0.747	0
0.518	0.115	0.634	0.518	0.57	0.747	1
0.783	0.38	0.773	0.783	0.776	0.747	

Decision Tree

```
a   b   <-- classified as
315  41 |  a = 0
 66  71 |  b = 1
```

a **simpler** model performs at least as good

oneR

```
a   b   <-- classified as
319  37 |  a = 0
 65  72 |  b = 1
```

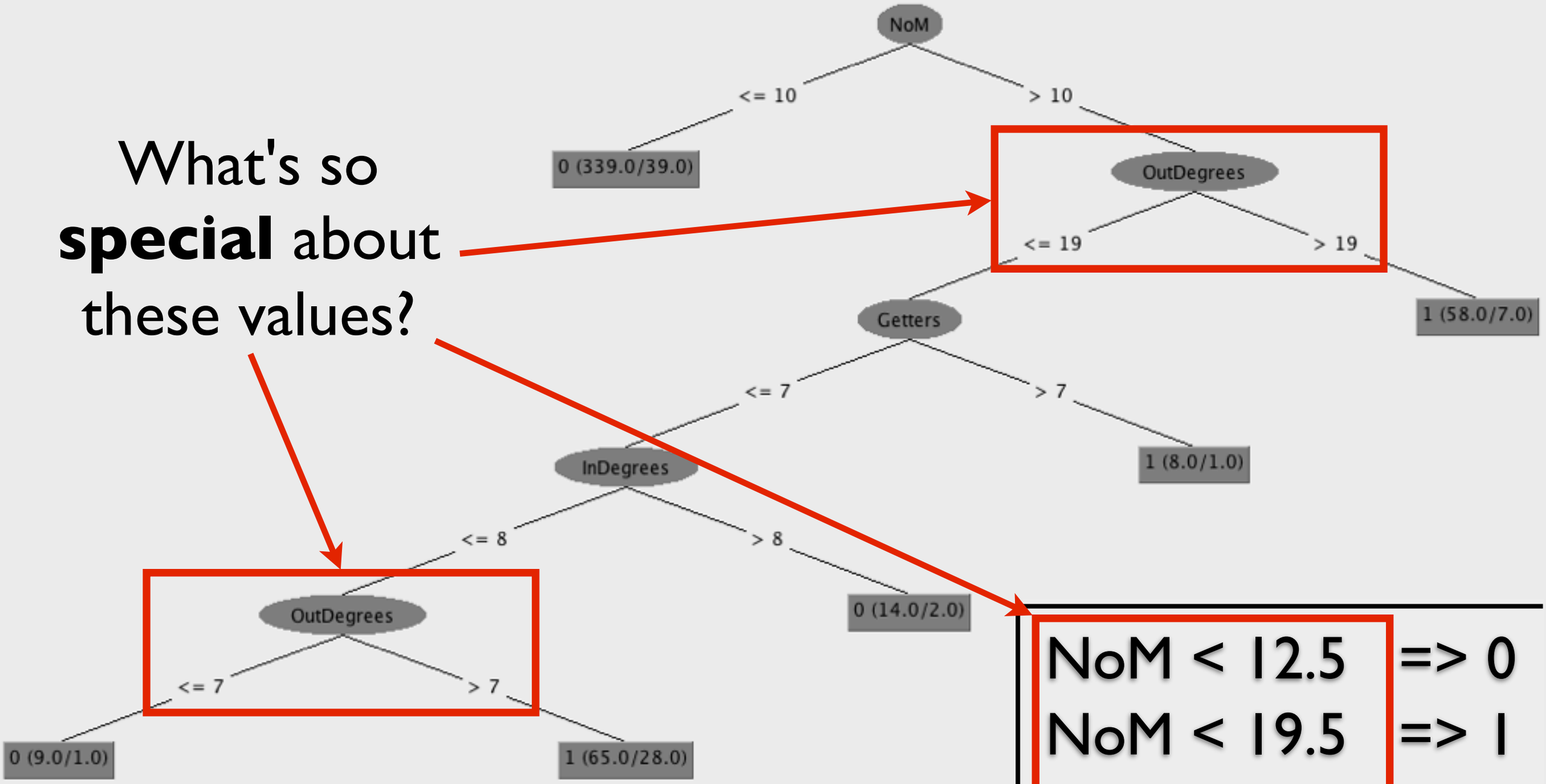
TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.896	0.474	0.831	0.896	0.862	0.711	0
0.526	0.104	0.661	0.526	0.585	0.711	1
0.793	0.371	0.783	0.793	0.785	0.711	

Weka Classifier Tree Visualizer: 22:38:34 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.N...

Tree View

Complex Models (2)

What's so **special** about these values?



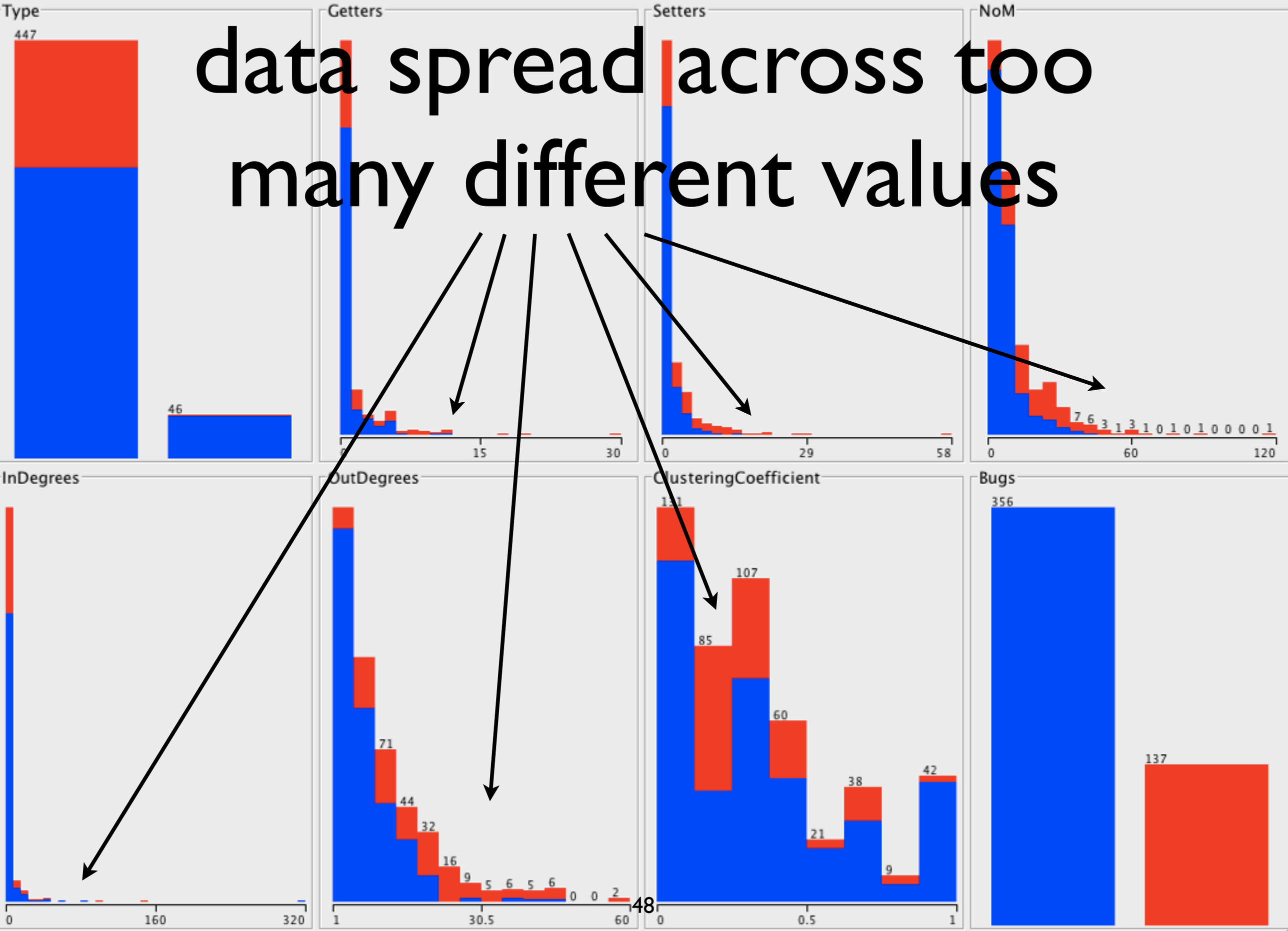
NoM < 12.5 => 0

NoM < 19.5 => 1

NoM < 22.5 => 0

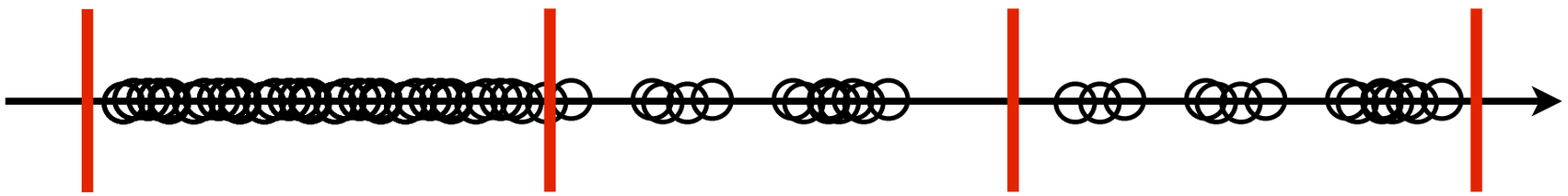
NoM >= 22.5 => 1

data spread across too many different values

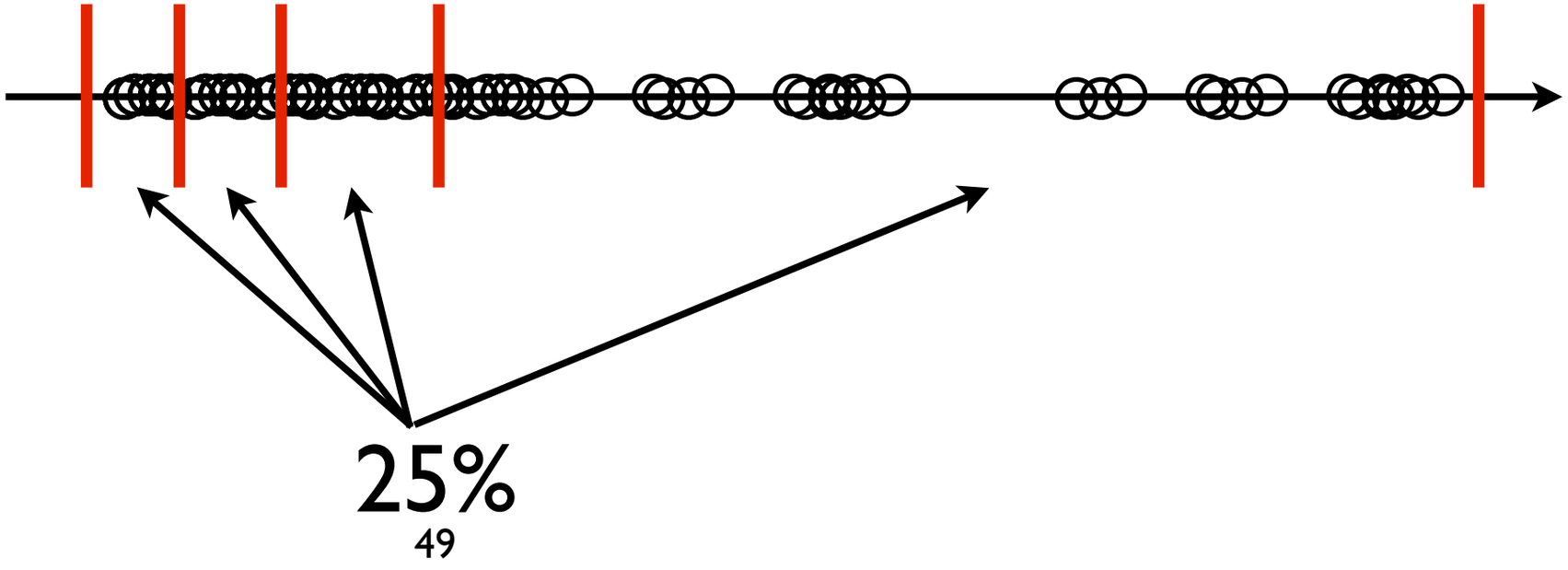


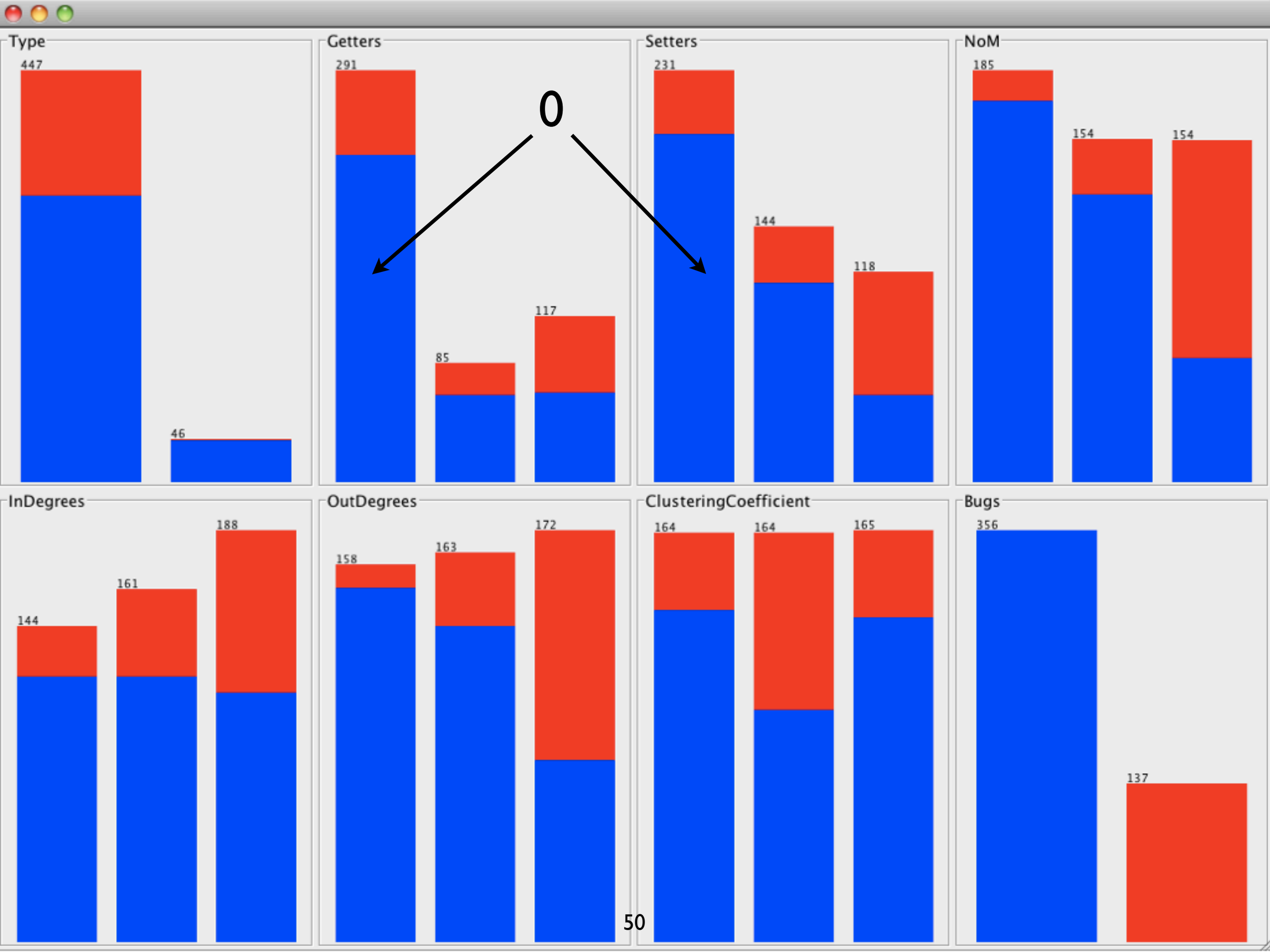
Unsupervised Discretization

equal-interval

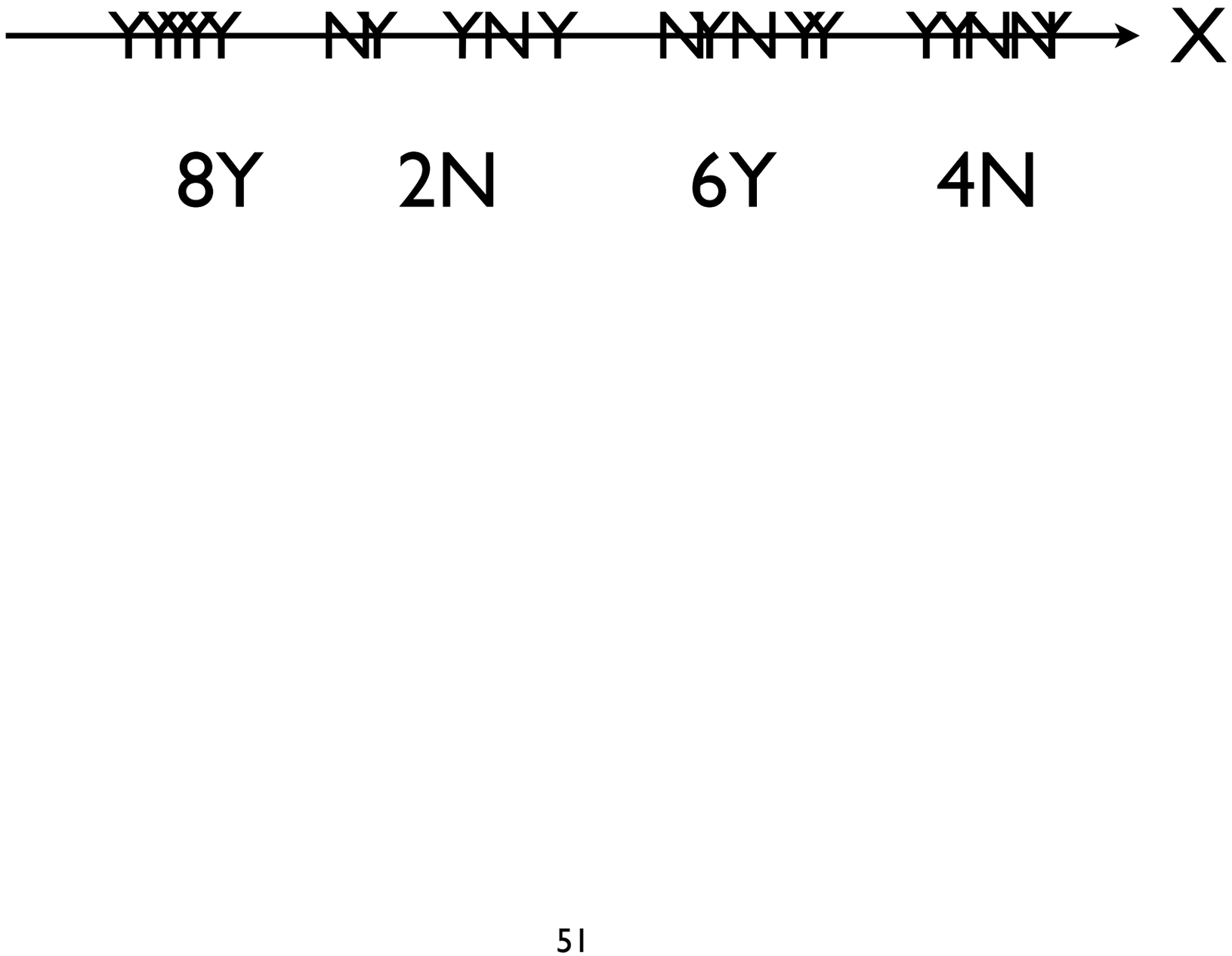


equal-frequency

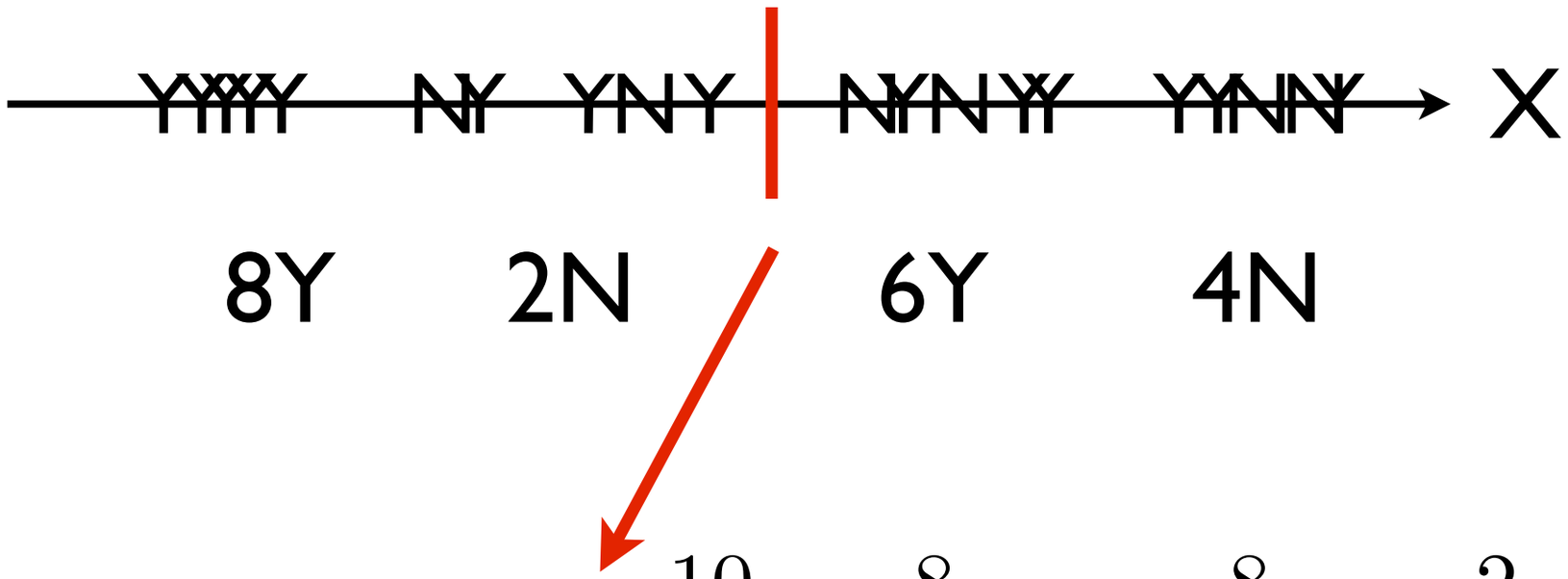



Type
447
46
Getters
291
85
117
Setters
231
144
118
NoM
185
154
154
0
InDegrees
144
161
188
OutDegrees
158
163
172
ClusteringCoefficient
164
164
165
Bugs
356
137

Supervised Discretization

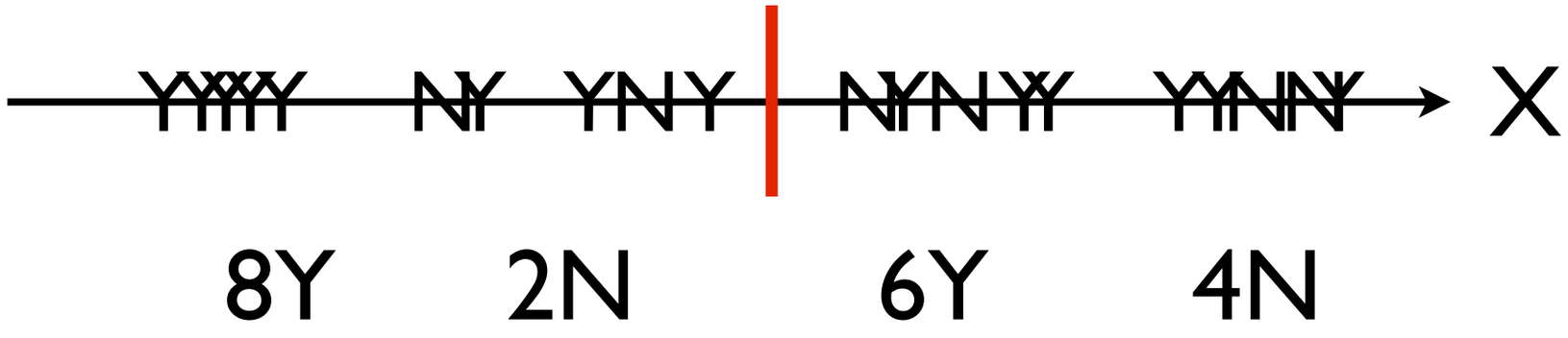


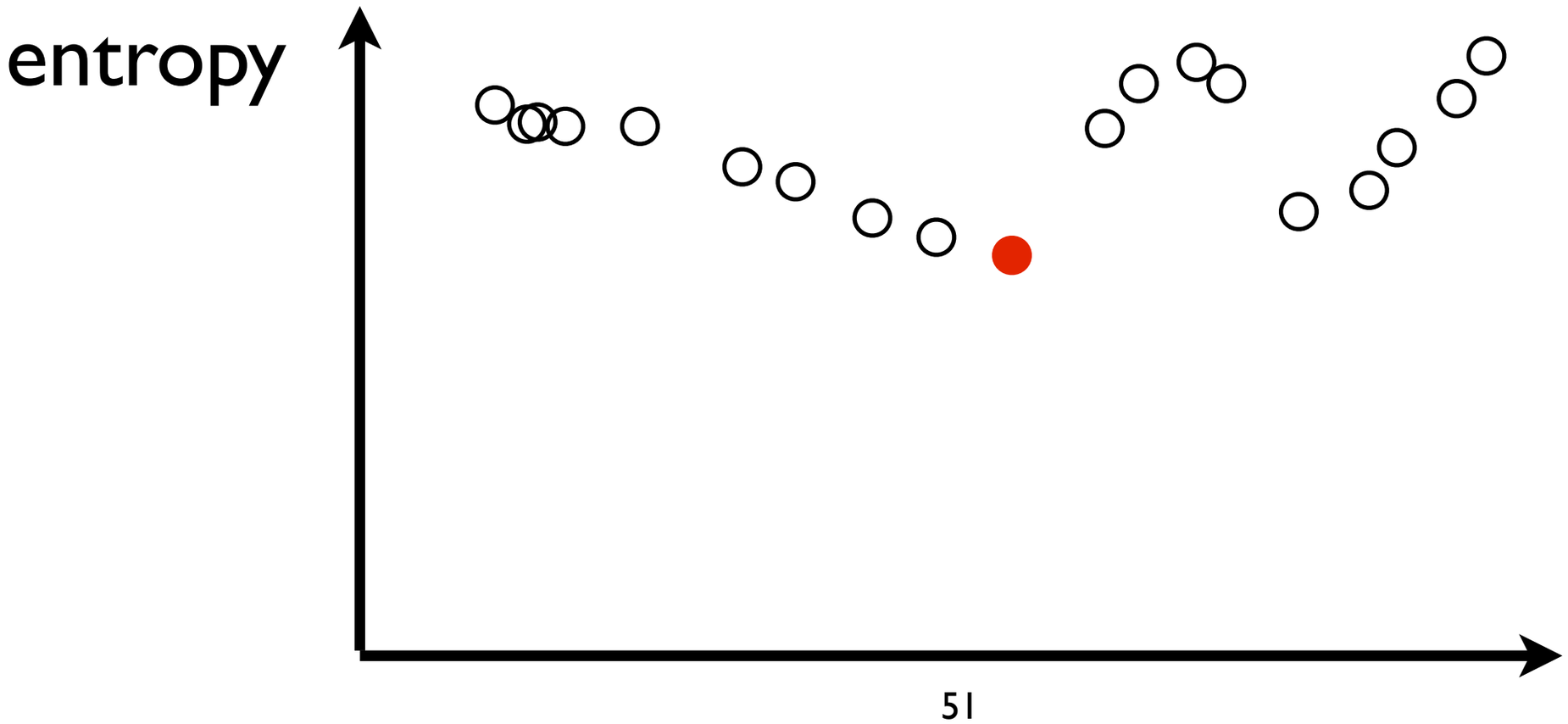
Supervised Discretization



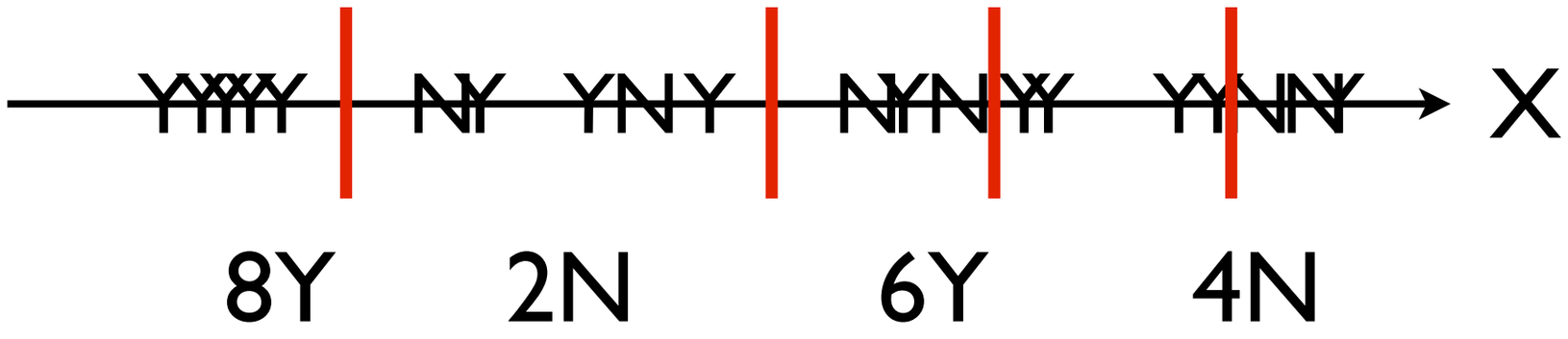
$$
\begin{aligned}
entropy &= \frac{10}{20}.[-\frac{8}{10}.log_2(\frac{8}{10}) - \frac{2}{10}.log_2(\frac{2}{10})] \\
&+ \frac{10}{20}.[-\frac{6}{10}.log_2(\frac{6}{10}) - \frac{4}{10}.log_2(\frac{4}{10})] \\
&= 0.846
\end{aligned}
$$

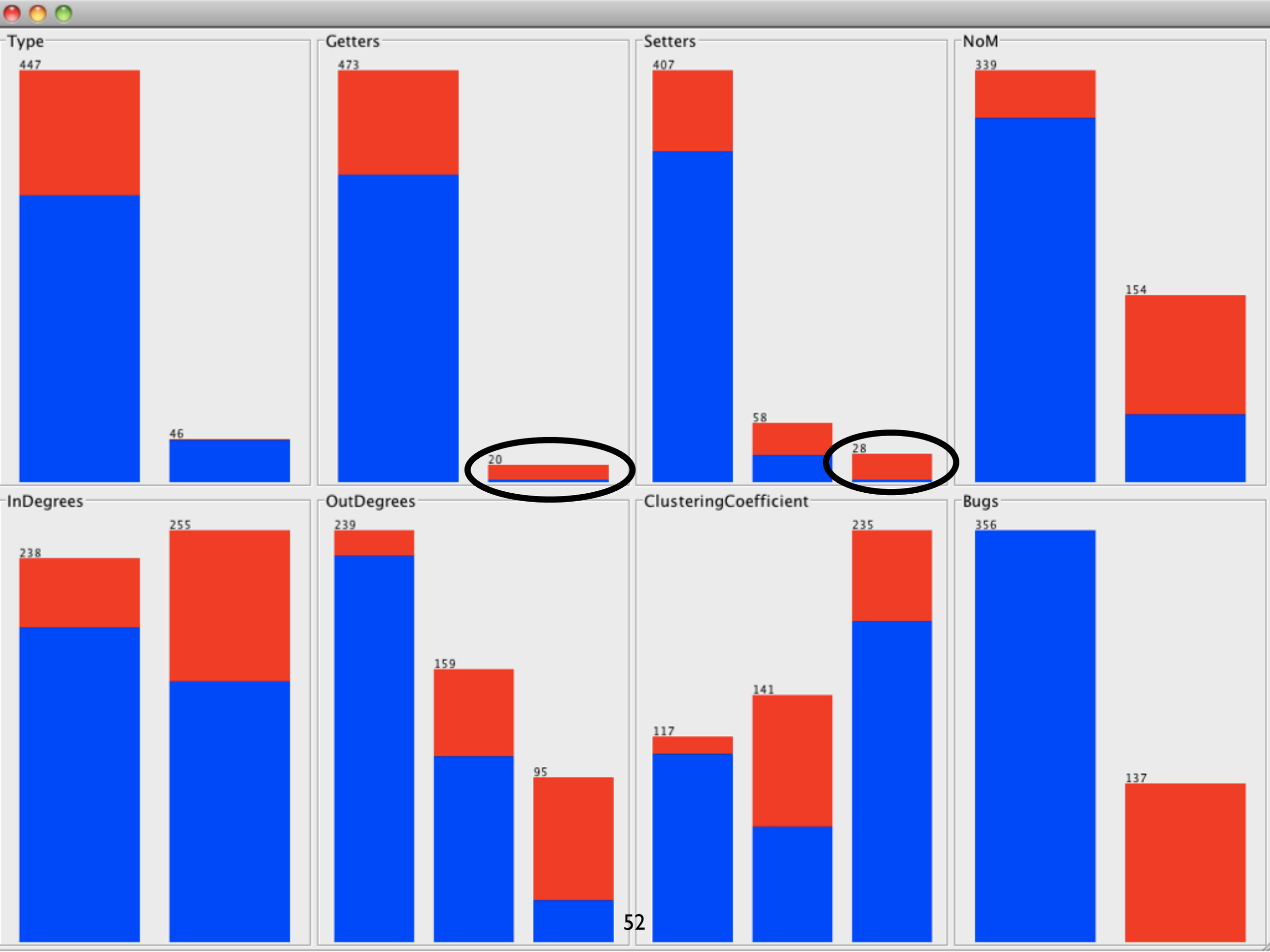
Supervised Discretization





Supervised Discretization




Type
447
46
Getters
473
20
Setters
407
58
28
NoM
339
154
InDegrees
238
255
OutDegrees
239
159
95
ClusteringCoefficient
117
141
235
Bugs
356
137

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.885	0.482	0.827	0.885	0.855	0.747	0
0.518	0.115	0.634	0.518	0.57	0.747	1
0.783	0.38	0.773	0.783	0.776	0.747	

Decision Tree

```
a  b   <-- classified as
315 41 |  a = 0
 66 71 |  b = 1
```

discretization **improves** performance

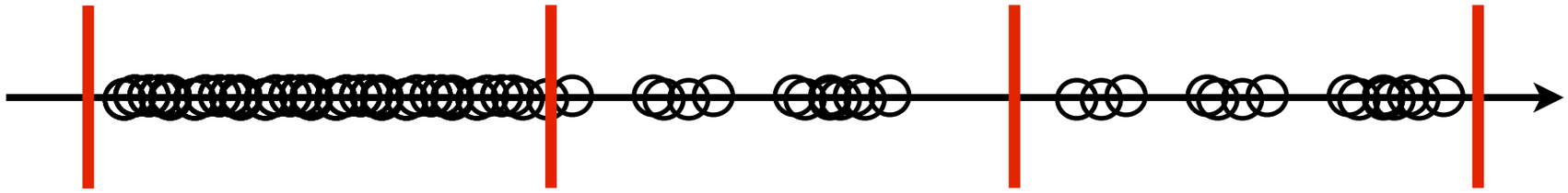
```
a  b   <-- classified as
312 44 |  a = 0
 49 88 |  b = 1
```

Equal-frequency

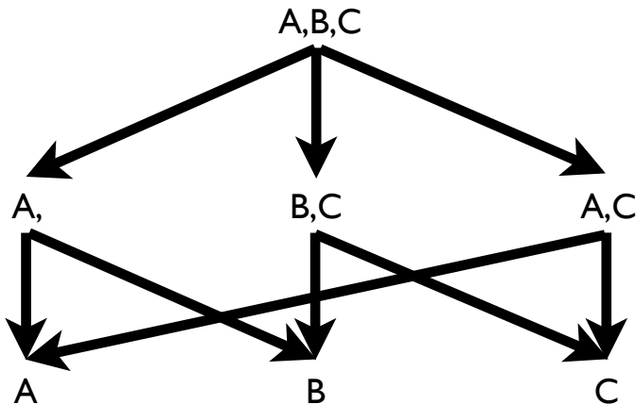
TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.876	0.358	0.864	0.876	0.87	0.774	0
0.642	0.124	0.667	0.642	0.654	0.774	1
0.811	0.293	0.809	0.811	0.81	0.774	

Data Carving Operators

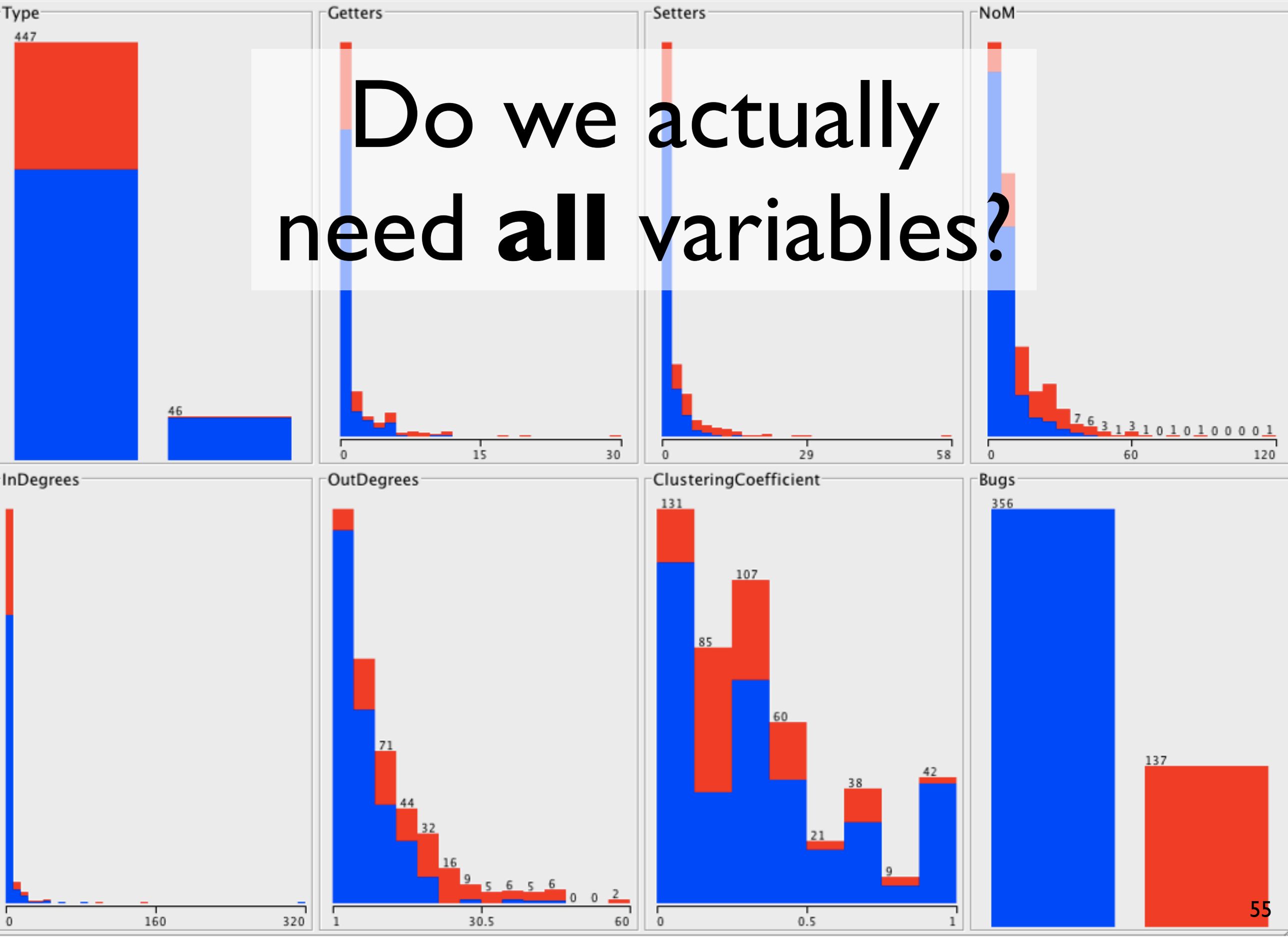
discretization



attribute selection



Do we actually need **all** variables?



Weka Explorer: Visualizing ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.NumericTransform-R9-C...

X: ClusteringCoefficient (Num)

Y: Bugs (Nom)

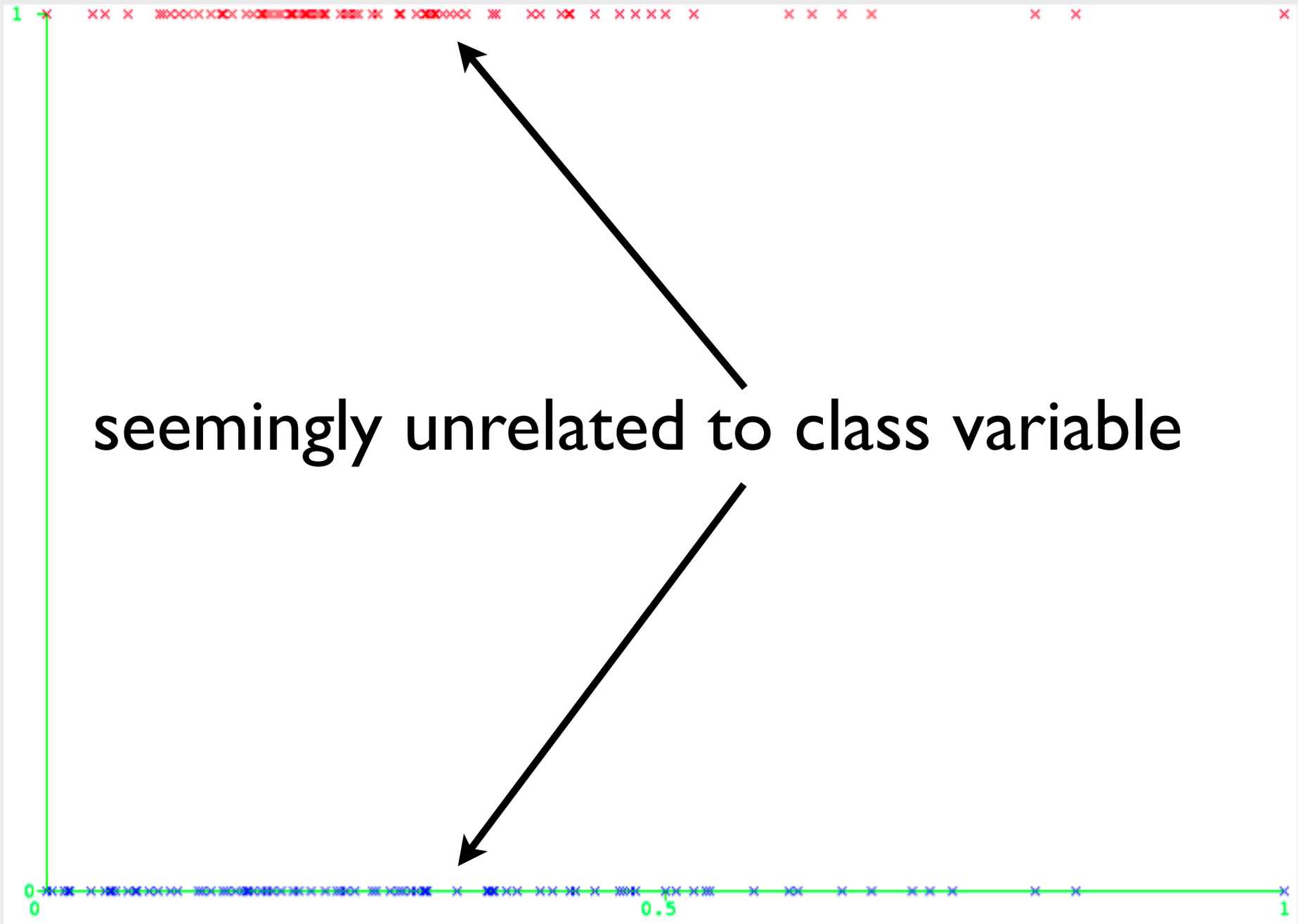
Colour: Bugs (Nom)

Select Instance

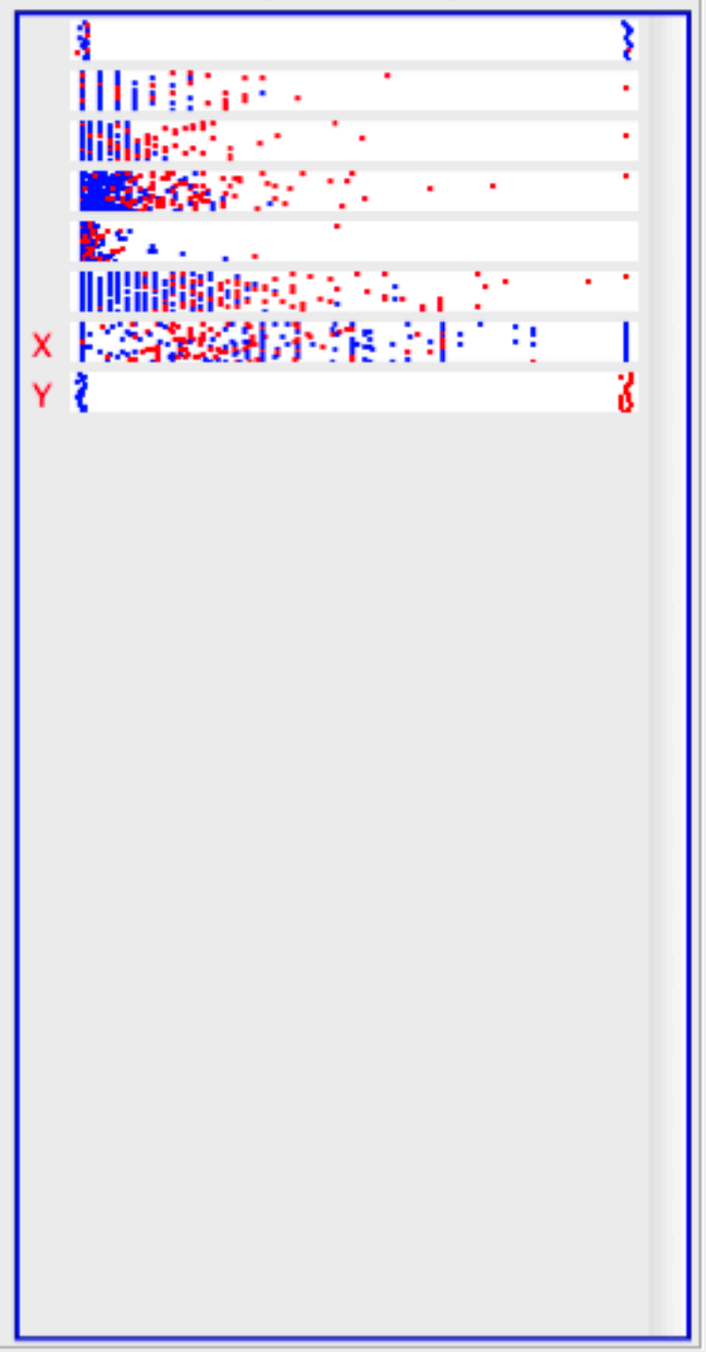
Reset | Clear | Open | Save

Jitter

Plot: ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute.NumericTransform-R9-Cjava.lang.Math-Msignum-weka.filte



seemingly unrelated to class variable



Class colour

0 1

Weka Explorer: Visualizing eclipse_files_2.0-weka.filters.unsupervised.attribute.NumericTransform-R4-Cjava.lang.Math-Msignum-weka.filters.uns...

X: SimpleName (Num)

Y: MLOC_sum (Num)

Colour: post (Nom)

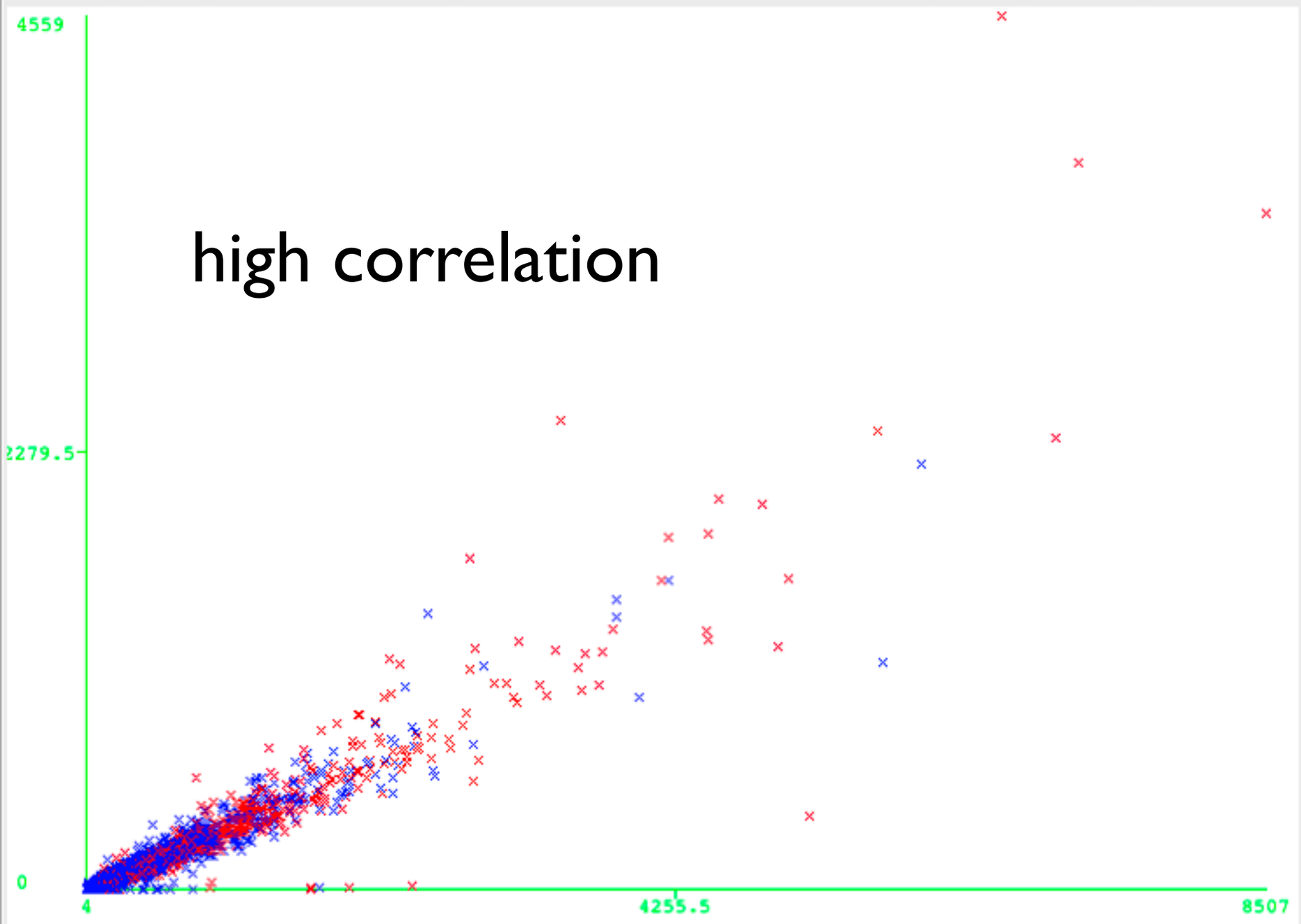
Select Instance

Select the shape to use for data selection

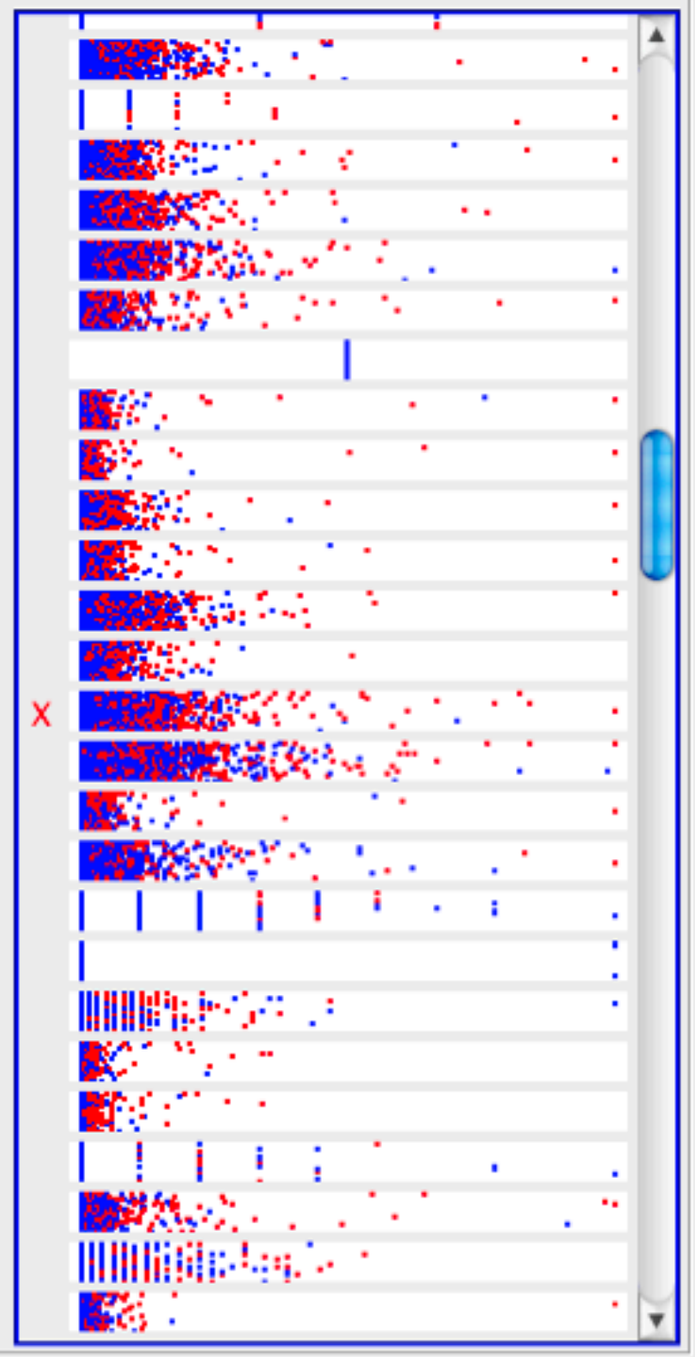
Reset | Clear | Open | Save

Jitter

Plot: eclipse_files_2.0-weka.filters.unsupervised.attribute.NumericTransform-R4-Cjava.lang.Math-Msignum-weka.filters.unsupervised.attribute.NumericToNomi



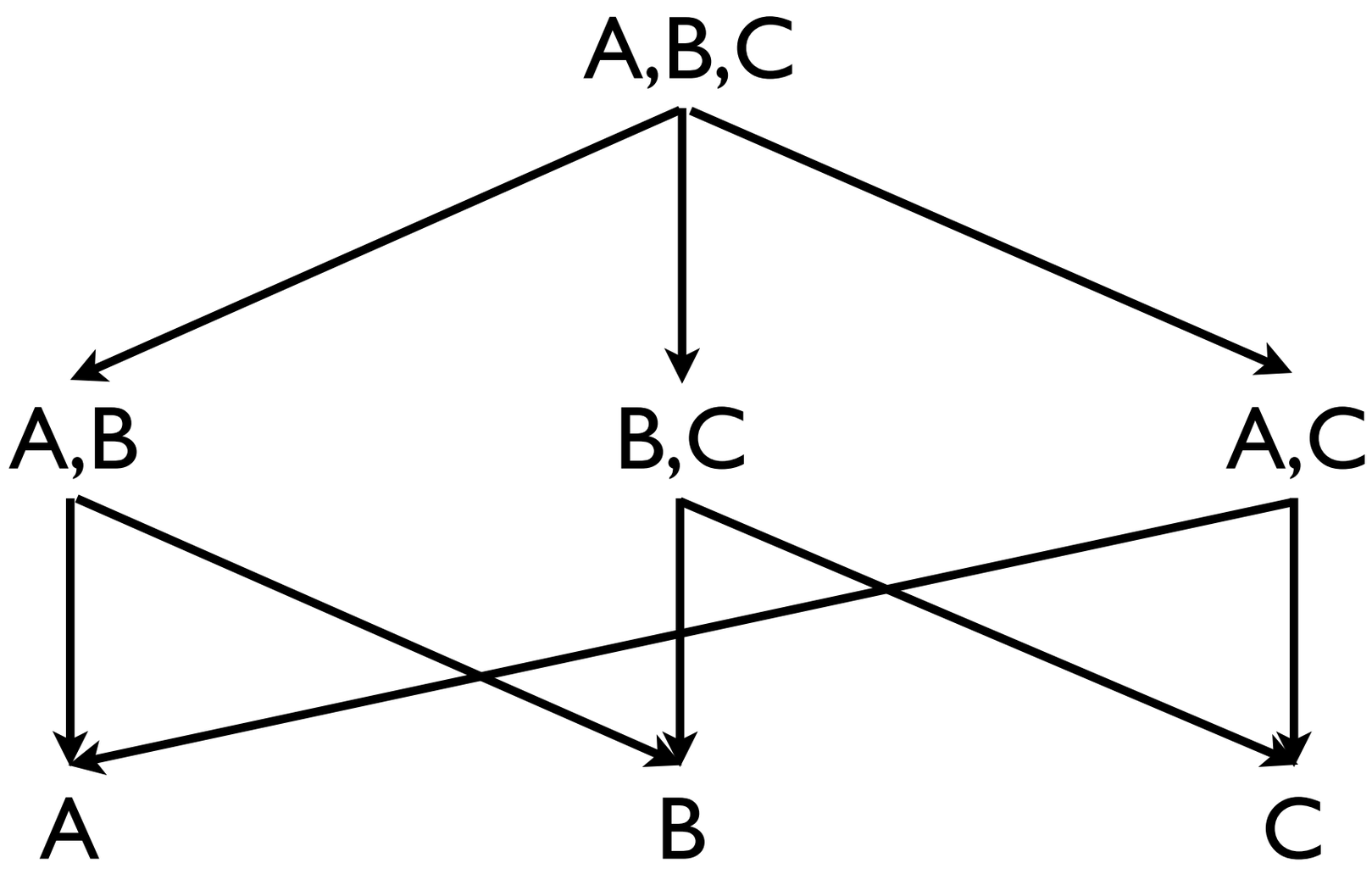
high correlation



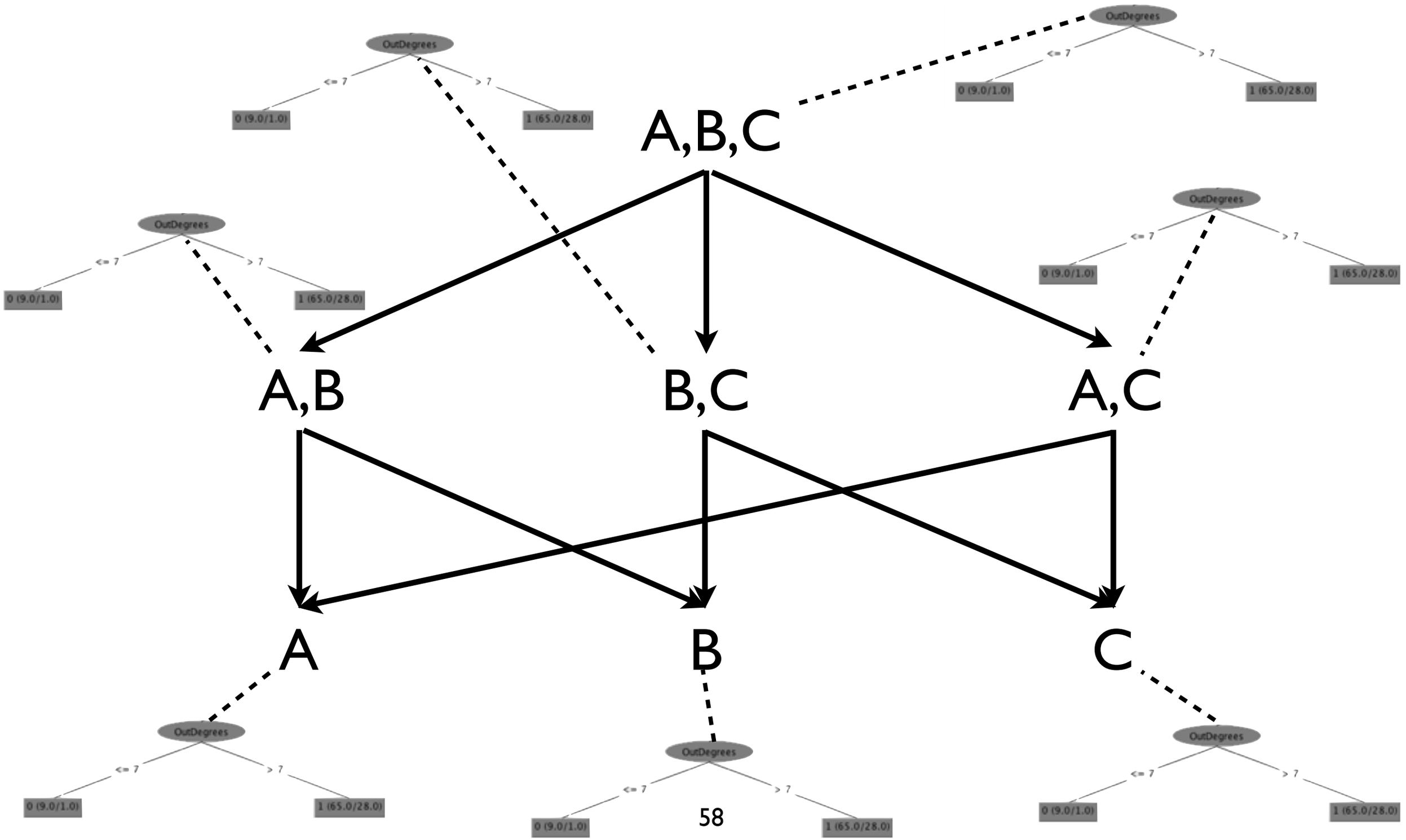
Class colour

0 1

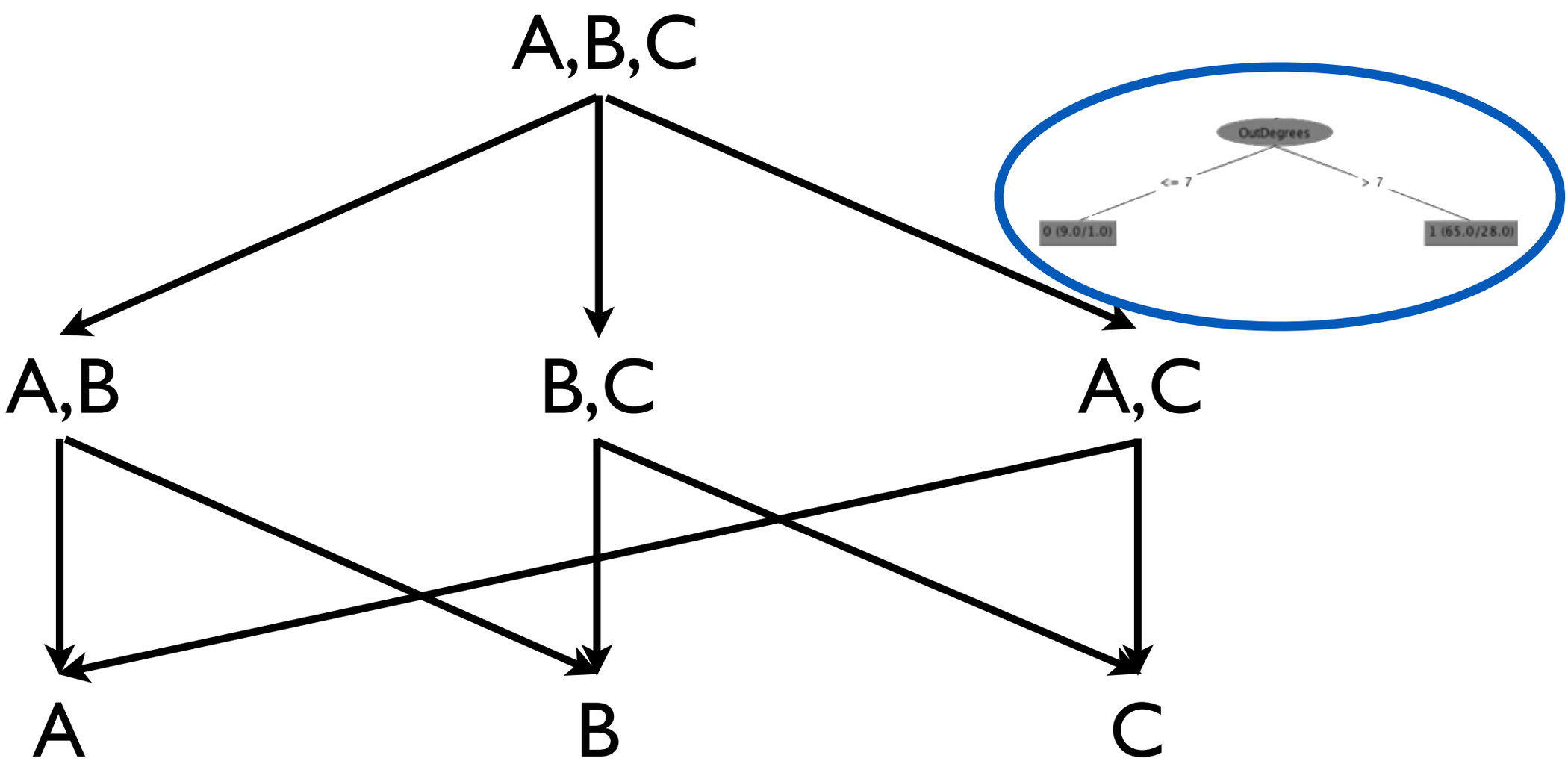
Attribute Selection



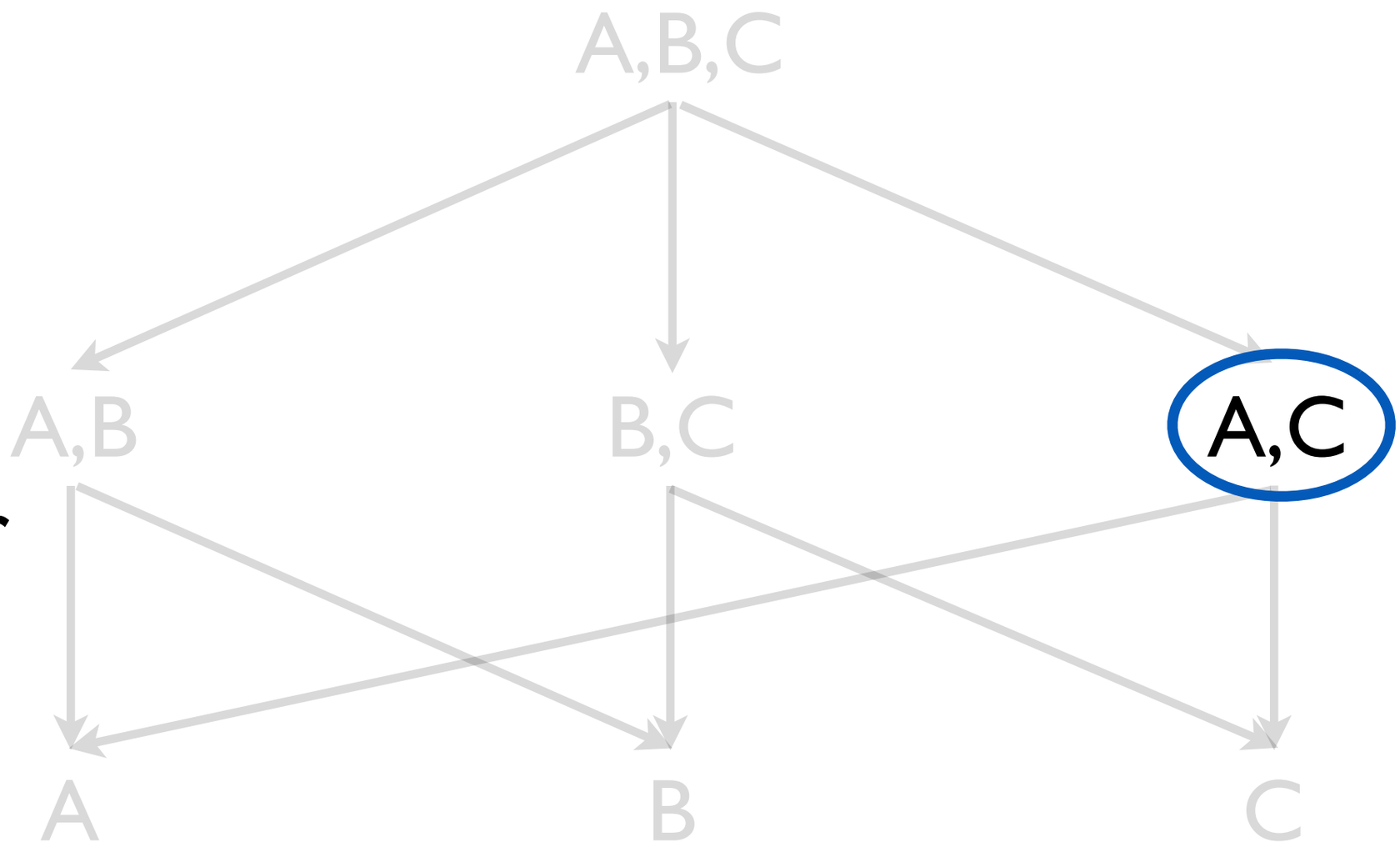
Attribute Selection



Attribute Selection



Attribute Selection



- wrapper
- infogain
- oneR
- CFS

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Attribute Evaluator

Choose **CfsSubsetEval**

attribute evaluator

Search Method

Choose **GreedyStepwise** -T -1.7976931348623157E308 -N -1

search method

Attribute Selection Mode

- Use full training set
- Cross-validation Folds 10
  Seed 1

(Nom) Bugs

Start | Stop

Result list (right-click for options)

22:57:56 - BestFirst + WrapperSubsetEval
22:58:22 - GreedyStepwise + WrapperSubse
22:58:45 - Ranker + InfoGainAttributeEval
22:59:23 - Ranker + OneRAttributeEval
23:00:39 - GreedyStepwise + CfsSubsetEval

Attribute selection output

```
=== Run information ===

Evaluator:     weka.attributeSelection.InfoGainAttributeEval
Search:        weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation:      ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribu
Instances:     493
Attributes:    8
               Type
               Getters
               Setters
               NoM
               InDegrees
               OutDegrees
               ClusteringCoefficient
               Bugs
Evaluation mode:    10-fold cross-validation


=== Attribute selection 10 fold cross-validation (stratified), seed: 1 ===

average merit      average rank  attribute
 0.242 +- 0.011     1.1 +- 0.3     6 OutDegrees
 0.225 +- 0.01      1.9 +- 0.3     4 NoM
 0.132 +- 0.011     3.1 +- 0.3     3 Setters
 0.109 +- 0.011     3.9 +- 0.3     7 ClusteringCoefficient
 0.051 +- 0.008     5   +- 0       2 Getters
 0.035 +- 0.004     6.3 +- 0.46    1 Type
 0.033 +- 0.004     6.7 +- 0.46    5 InDegrees
```

Status

OK

Log x 0

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.885	0.482	0.827	0.885	0.855	0.747	0
0.518	0.115	0.634	0.518	0.57	0.747	1
0.783	0.38	0.773	0.783	0.776	0.747	

Decision Tree

```
   a  b   <-- classified as
 315  41 |  a = 0
  66  71 |  b = 1
```

simpler, better model

```
   a  b   <-- classified as
 309  47 |  a = 0
  53  84 |  b = 1
```

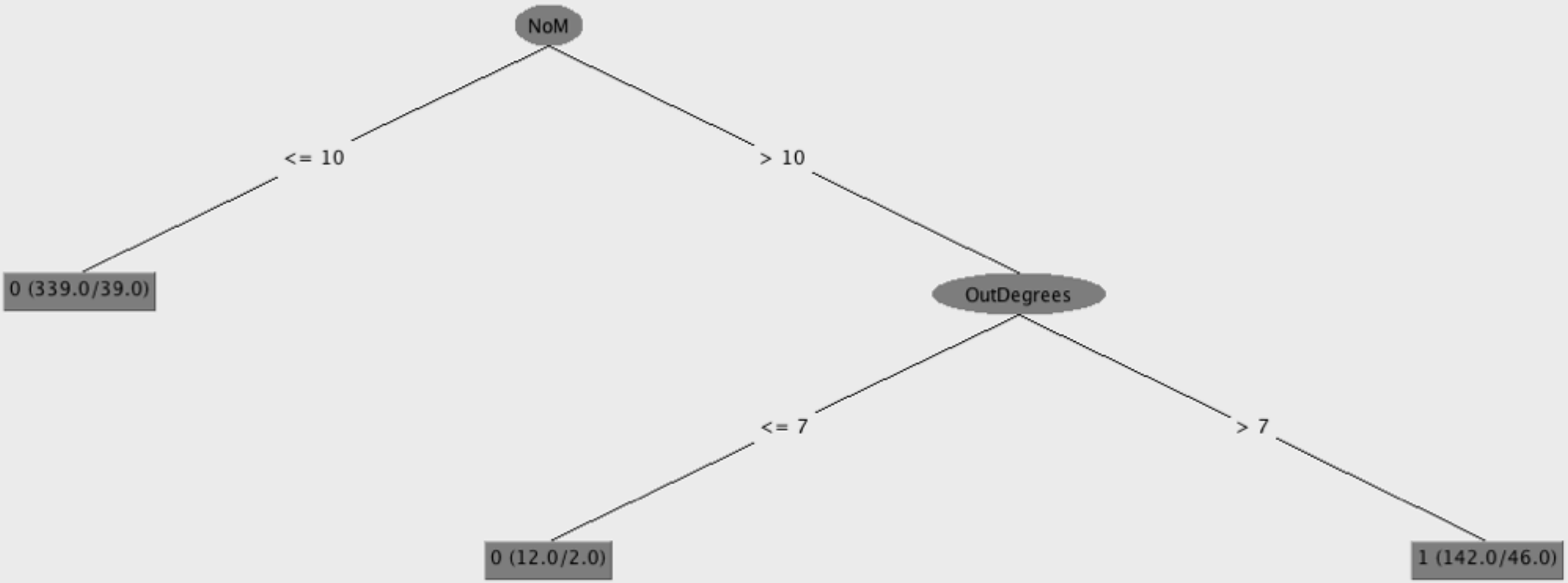
Attribute Selection

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.868	0.387	0.854	0.868	0.861	0.737	0
0.613	0.132	0.641	0.613	0.627	0.737	1
0.797	0.316	0.795	0.797	0.796	0.737	

Weka Classifier Tree Visualizer: 15:36:38 - trees.J48 (ant-1.7-weka.filters.unsupervised.attribute.Remove-R1-3-weka.filters.unsupervised.attribute...

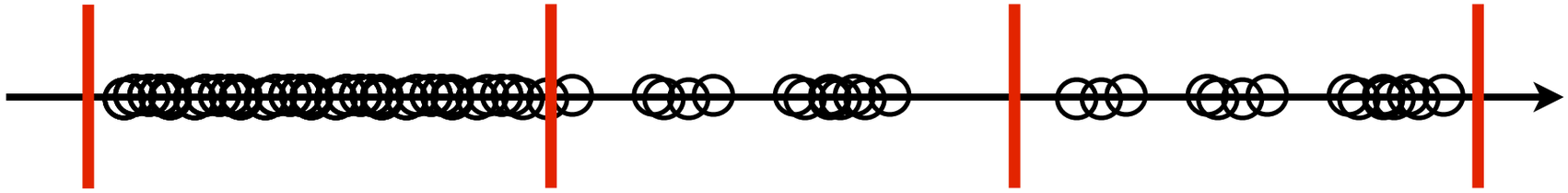
Tree View

simple model

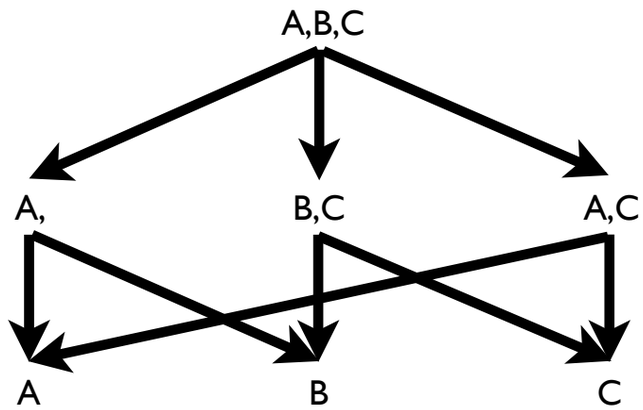


Data Carving Operators

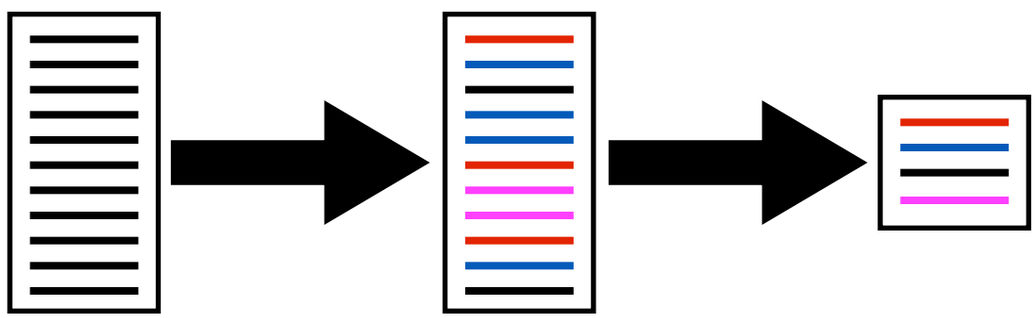
discretization



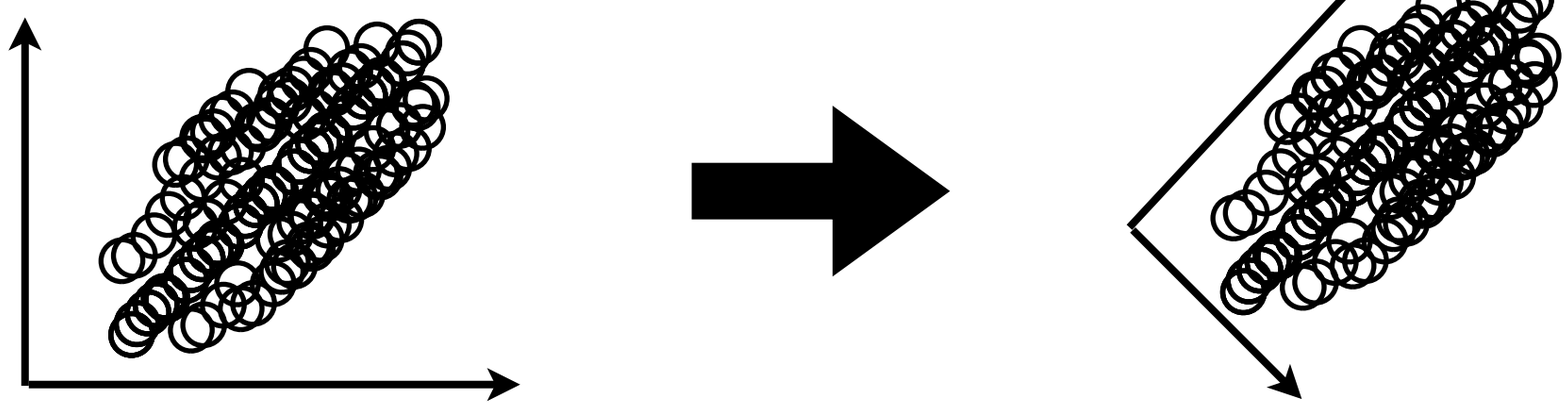
attribute selection



row selection



rotation



Points in Favour/Against

- intuitive models
- powerful

- can yield complex models
- no probabilities

Logistic Regression: What are the Odds?

Can we Predict the Popularity of a Song? (3)

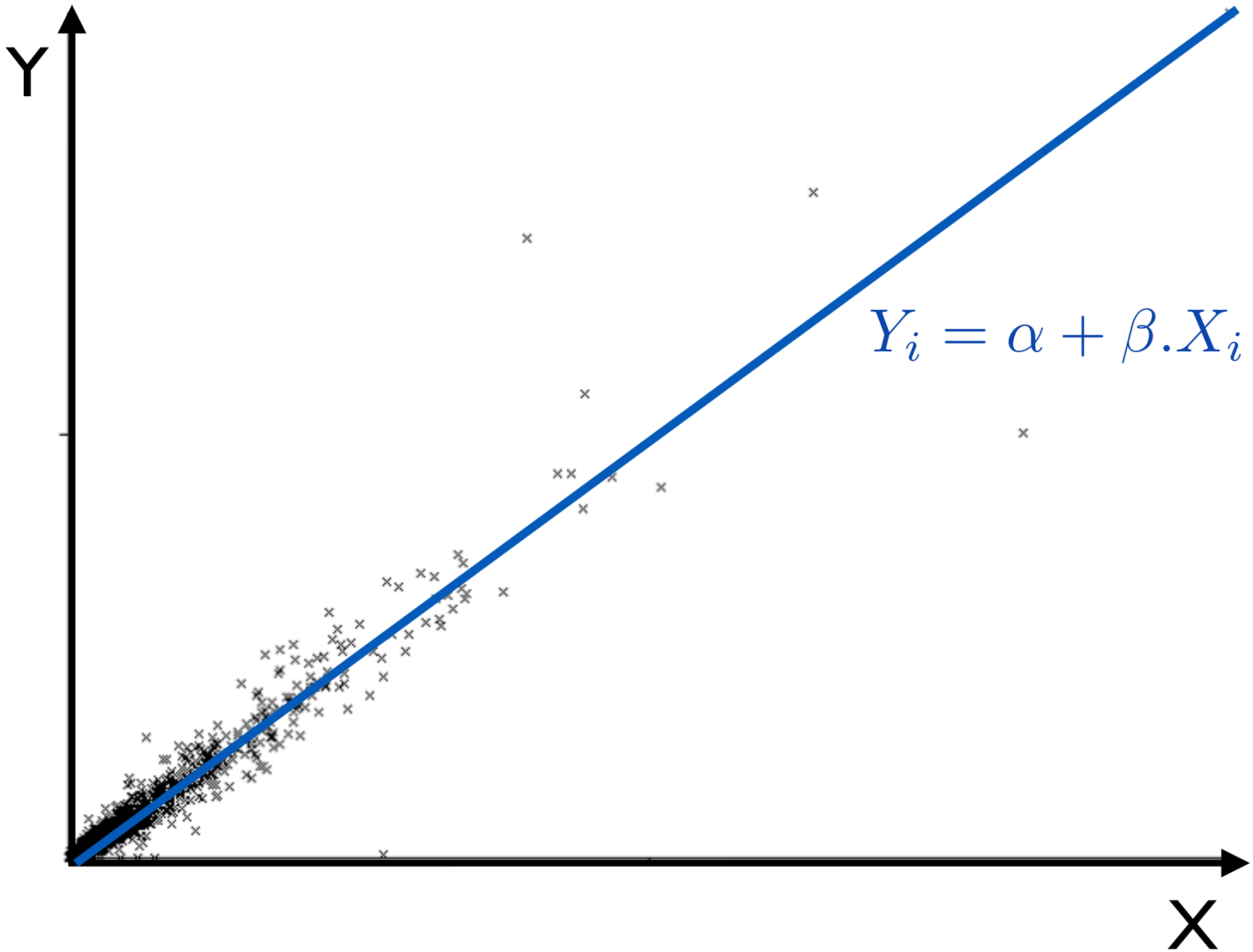
Never Sing, Never

Your World

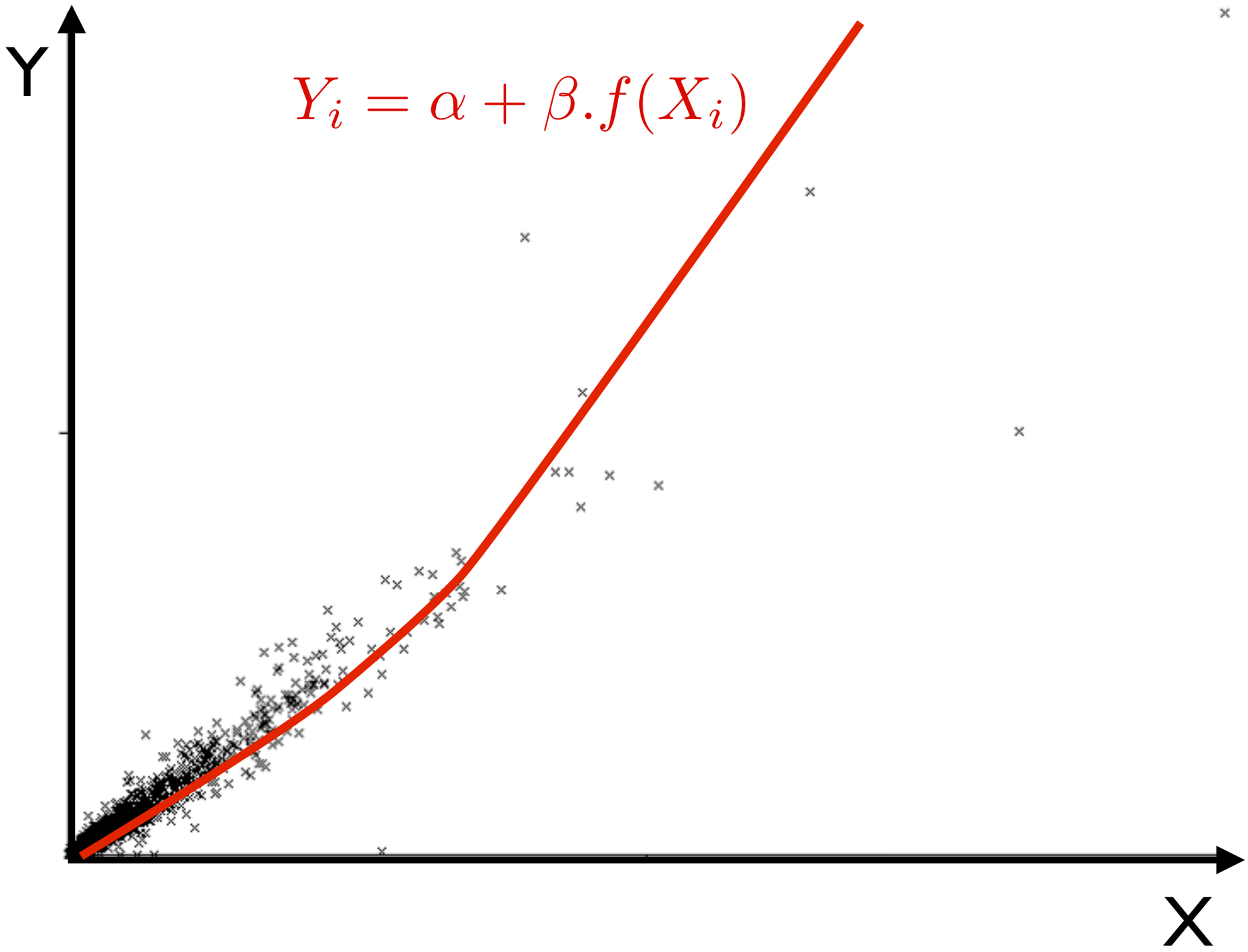
99%

51%

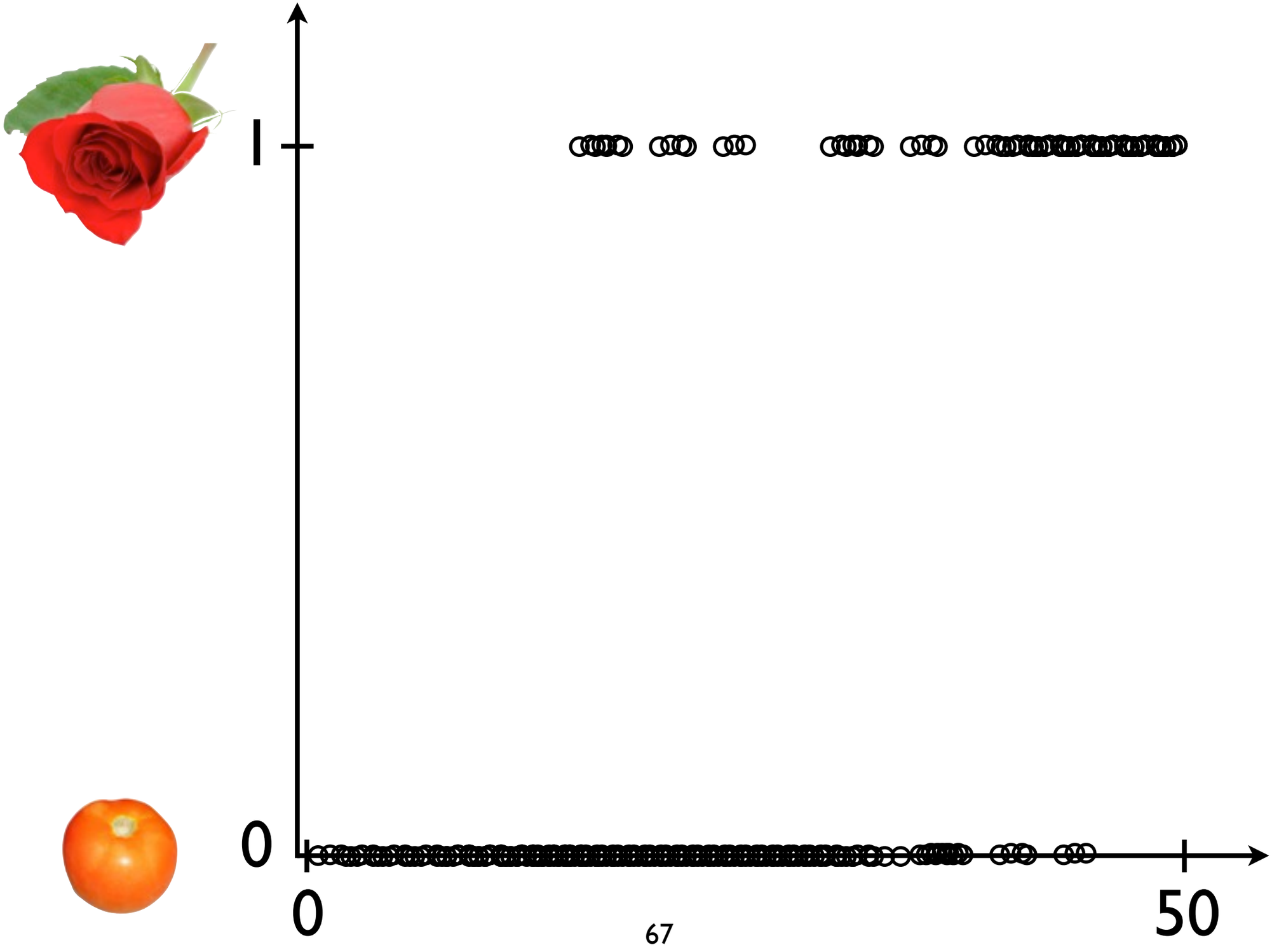
Linear Regression



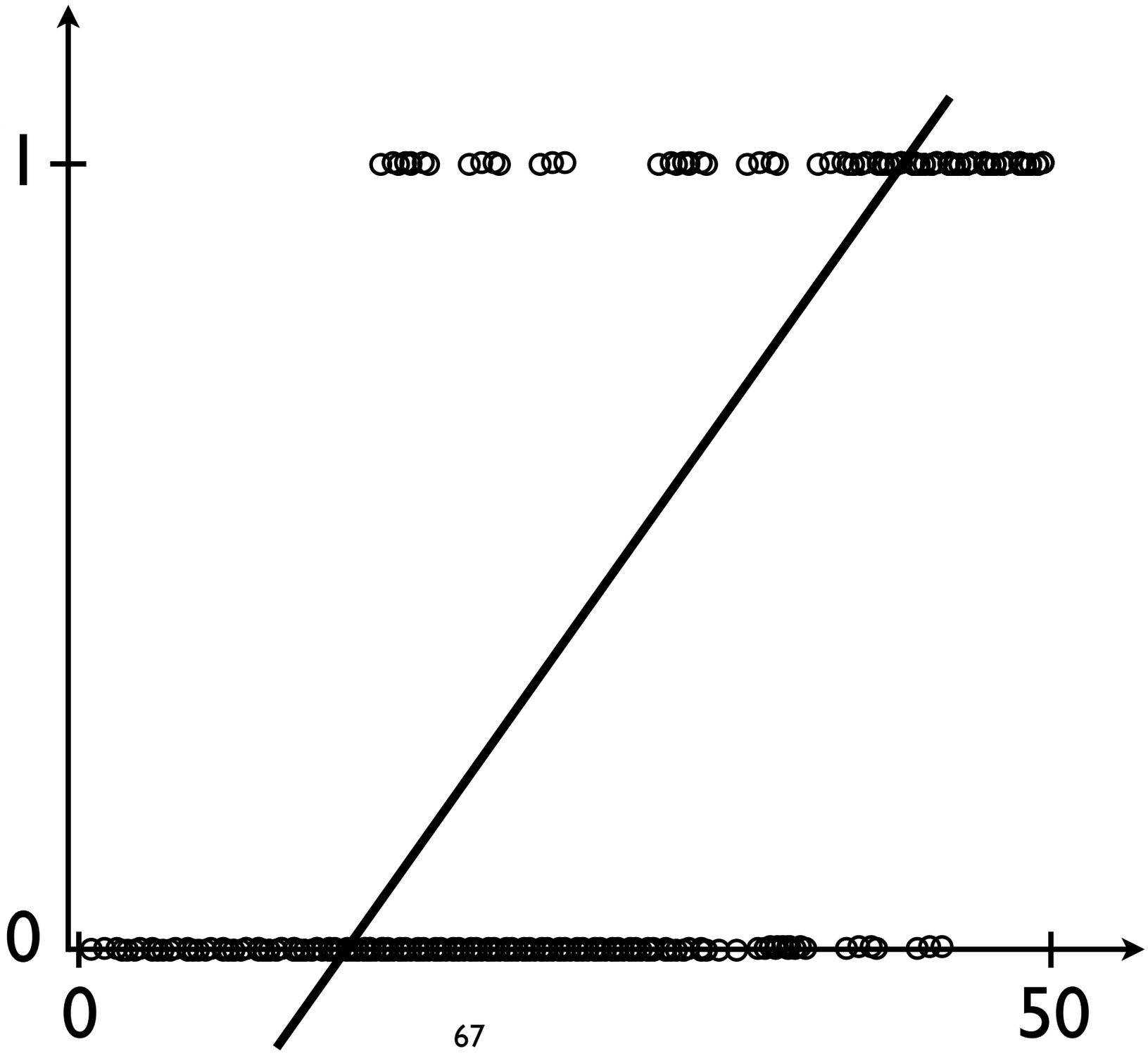
Linear Regression



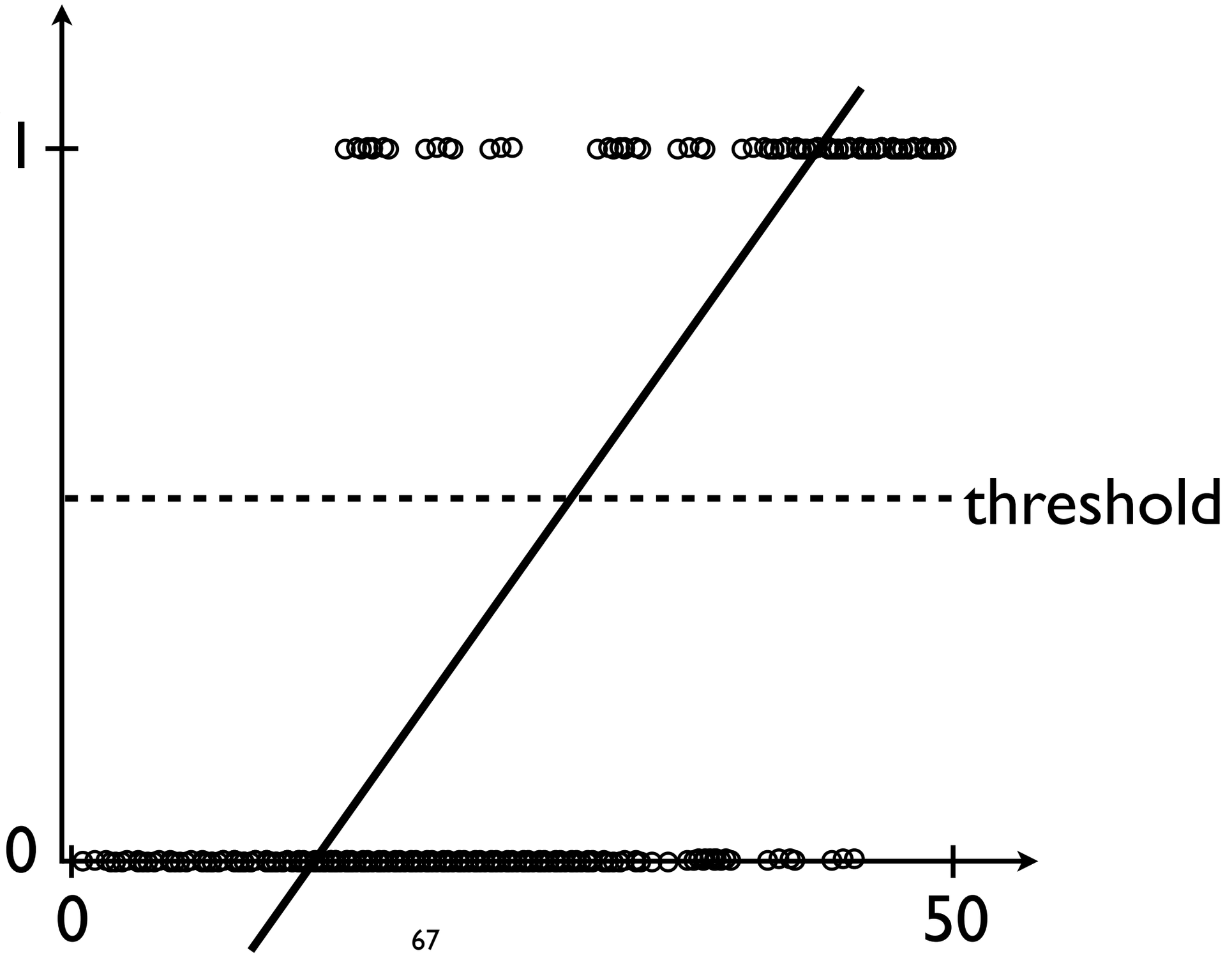
Logistic Regression



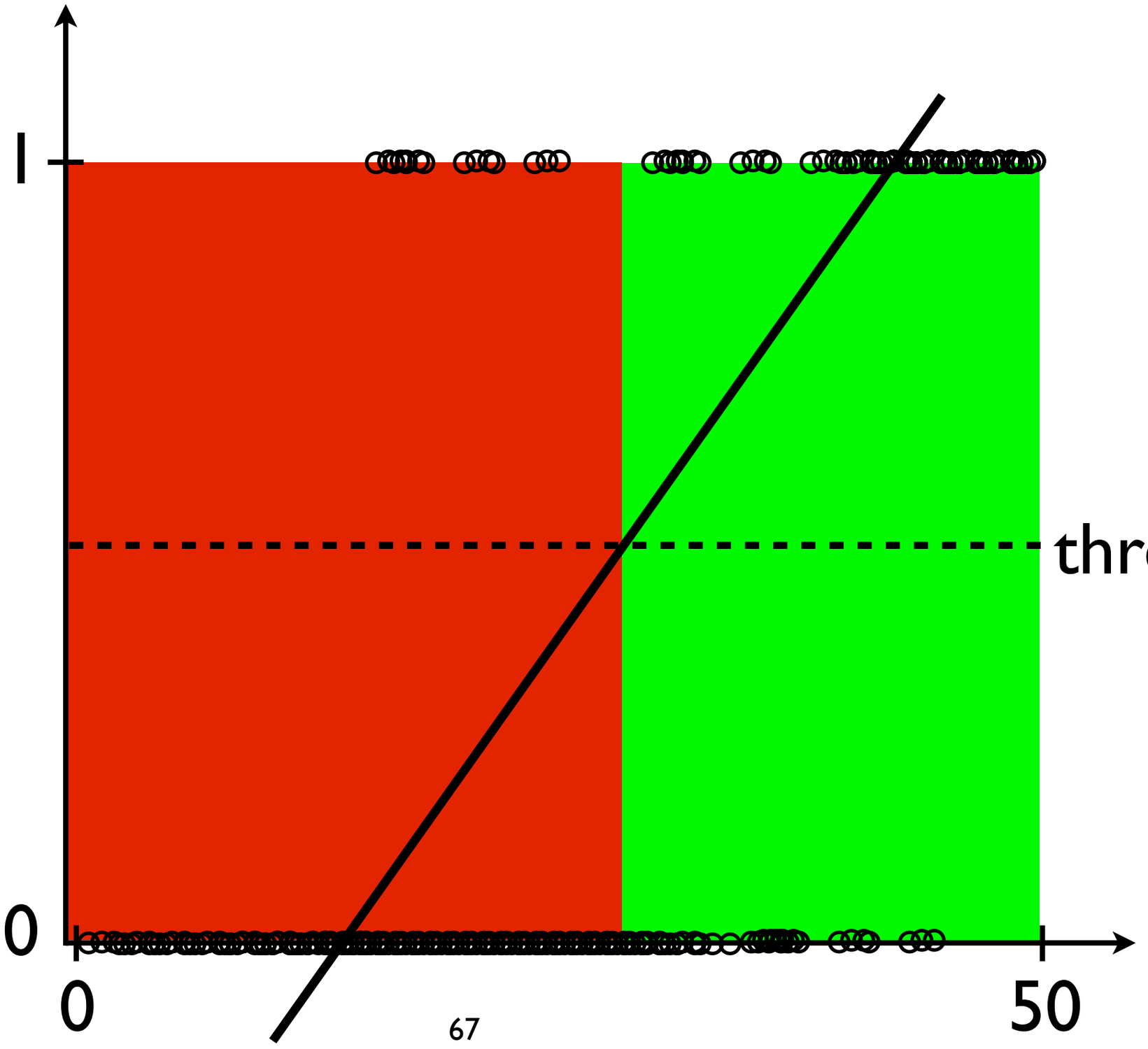
Logistic Regression



Logistic Regression



Logistic Regression



Logistic Regression

$$\pi_i = Pr(Y = 1 | X = X_i)$$

1

threshold

0

0

50

Logistic Regression

$$\pi_i = Pr(Y = 1 | X = X_i)$$

1

threshold

0

0

50

Logistic Regression

$$\pi_i = Pr(Y = 1 | X = X_i)$$

1

threshold

0

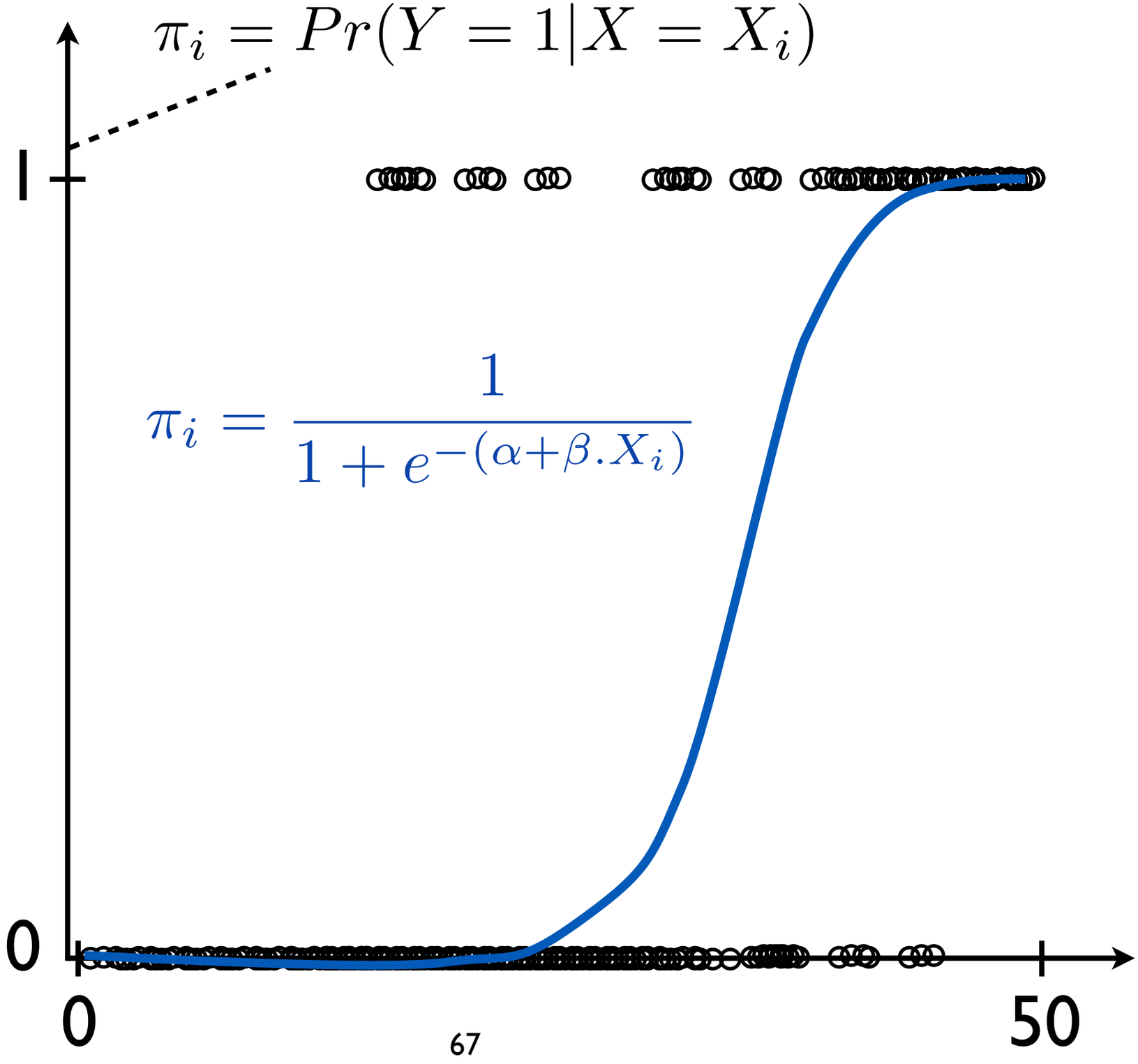
0

50

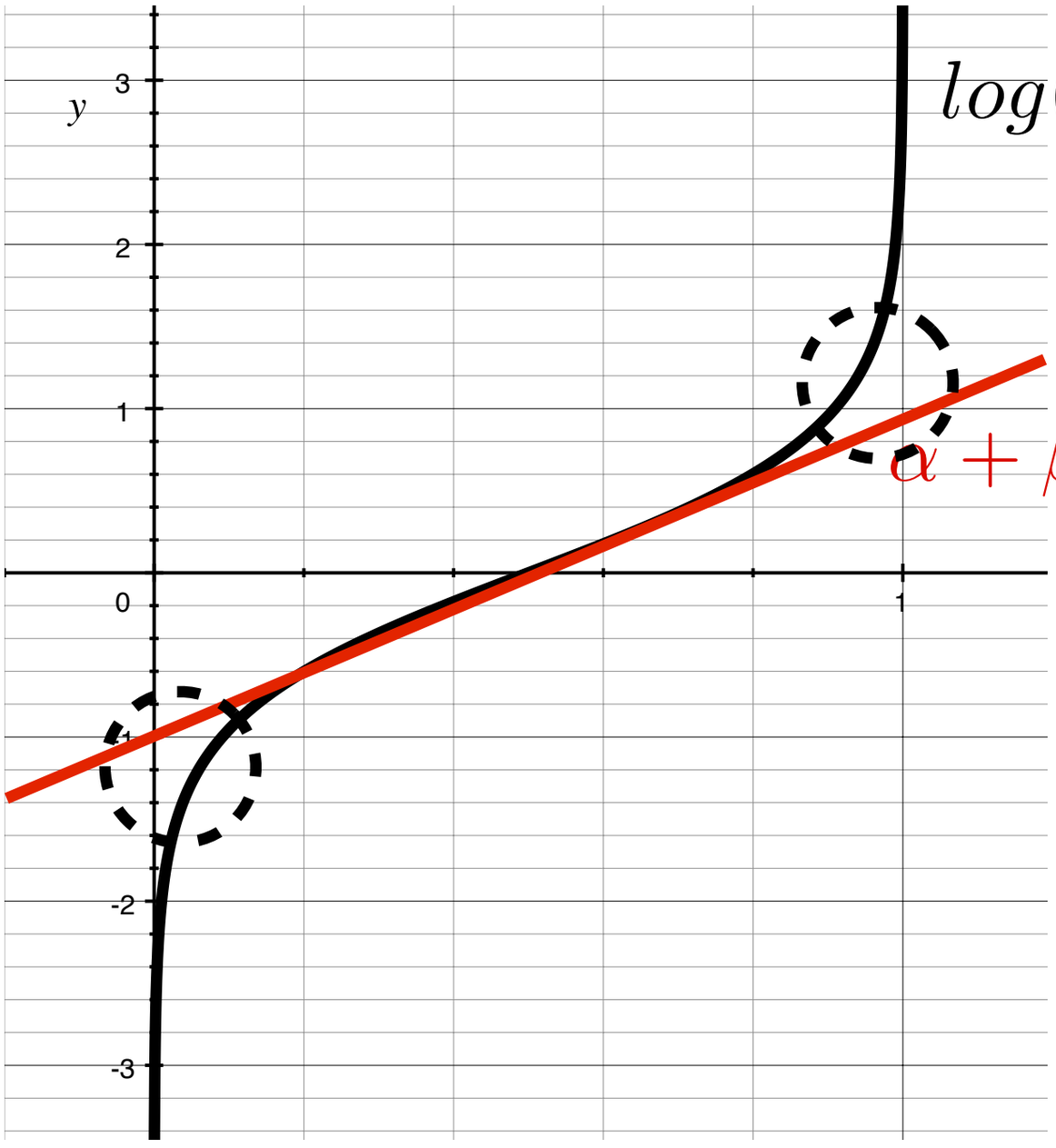
Logistic Regression

$$\pi_i = Pr(Y = 1|X = X_i)$$

$$\pi_i = \frac{1}{1 + e^{-(\alpha+\beta.X_i)}}$$



Log Odds



coefficient

intercept

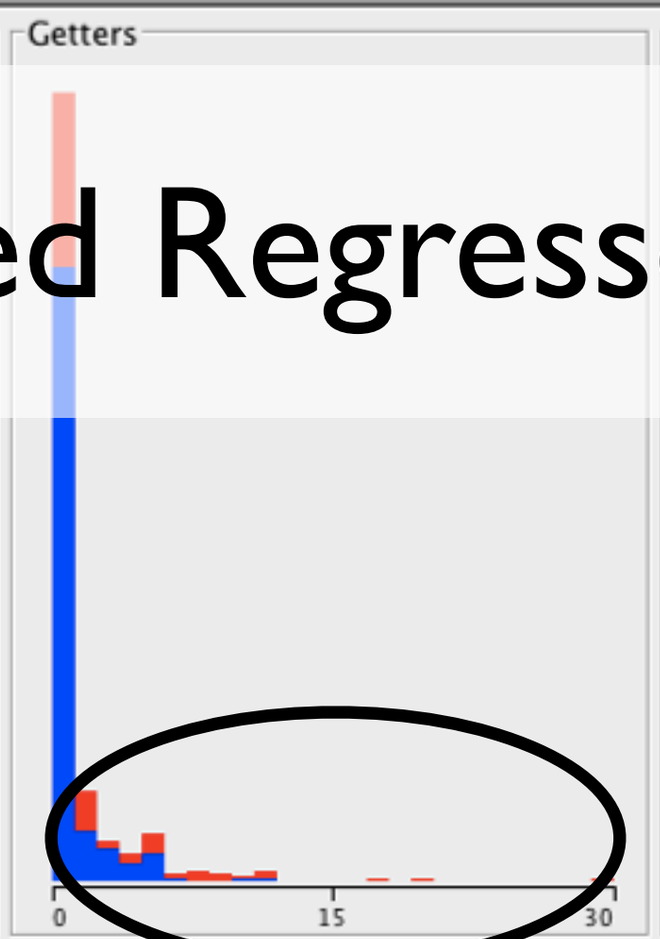
$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta.X_i$$

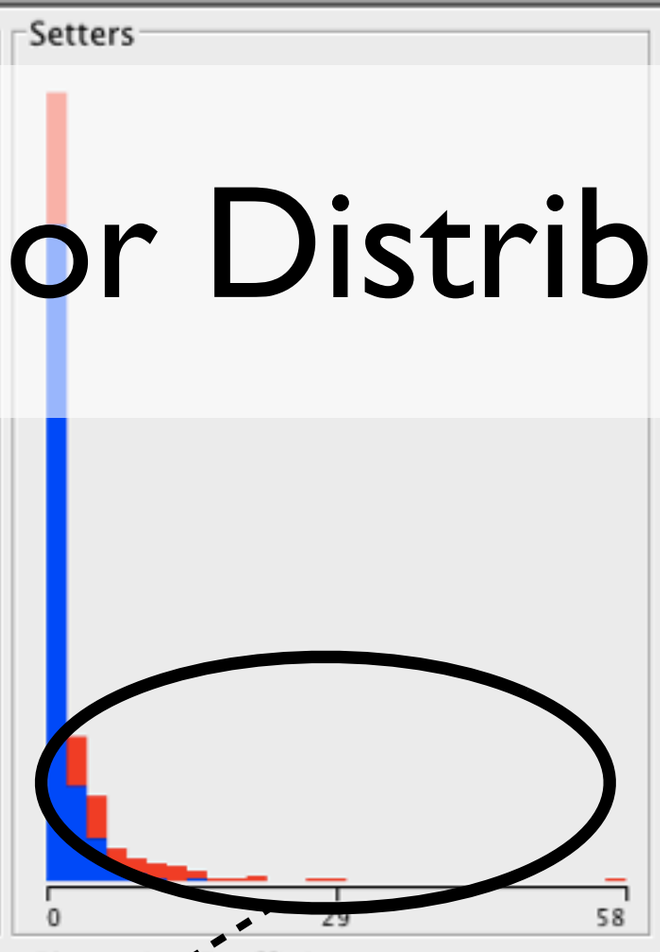
odds ratio regressor

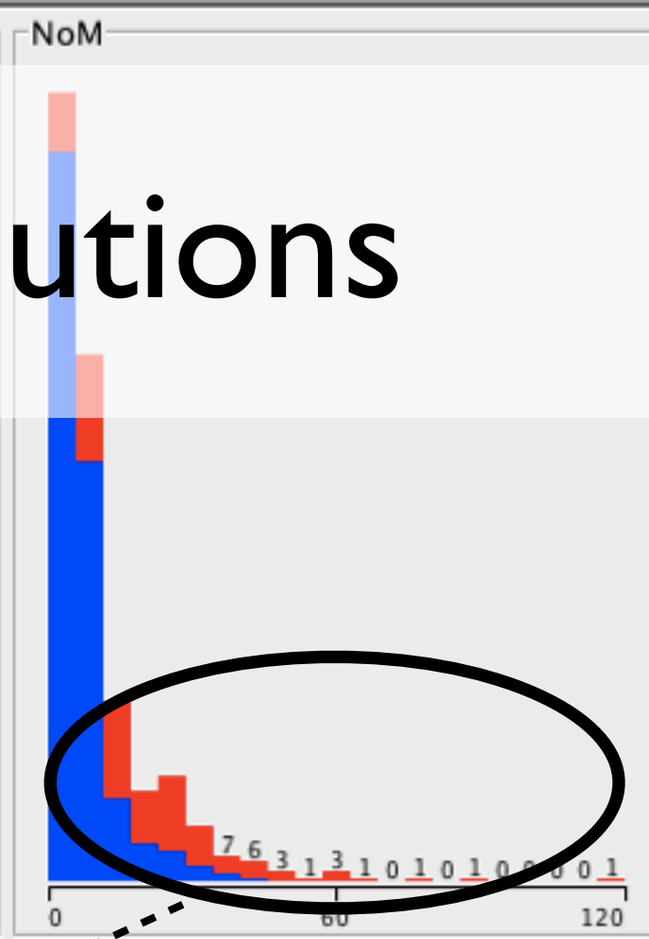
$$\Rightarrow \frac{\pi_i}{1-\pi_i} = e^{(\alpha+\beta.X_i)}$$

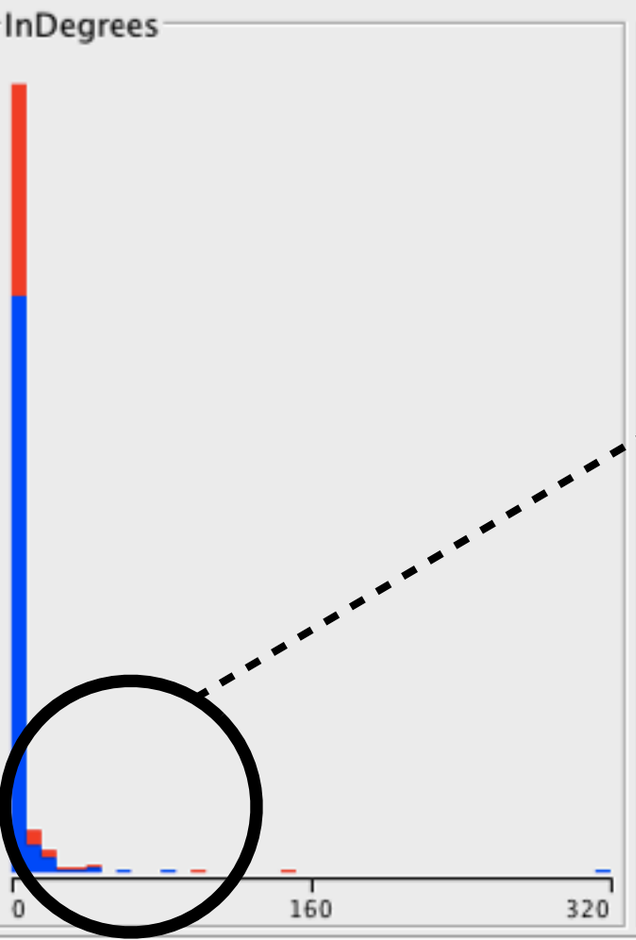
$$X_i + 1 \Rightarrow odds\ ratio \times e^{\beta}$$

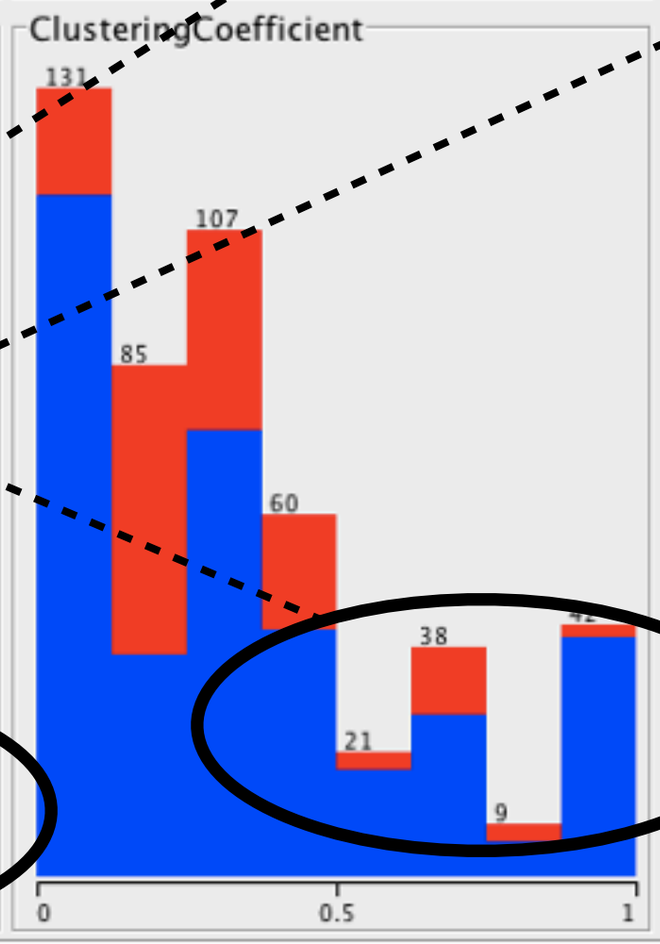
Skewed Regressor Distributions

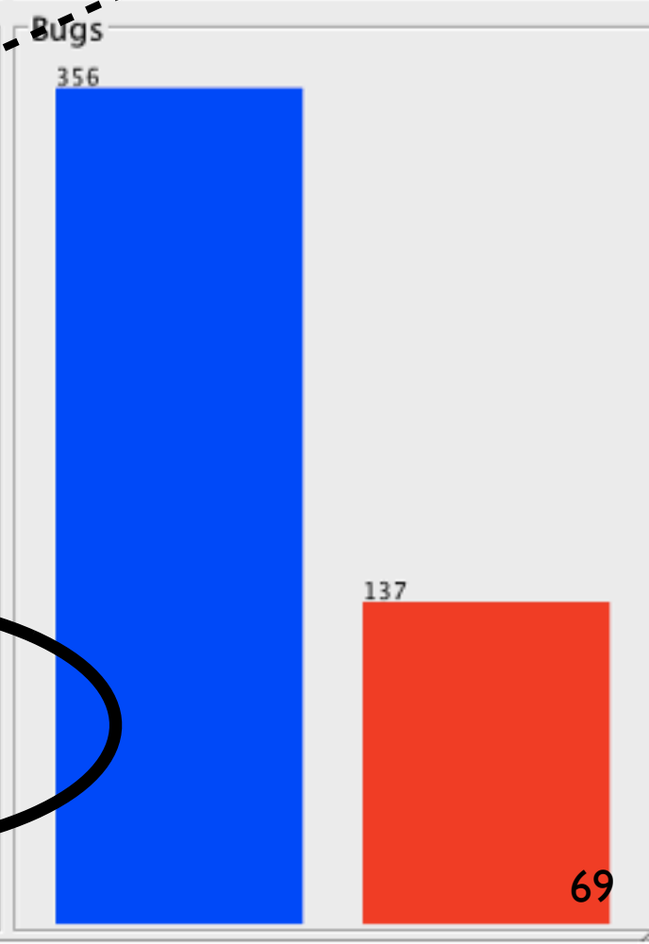








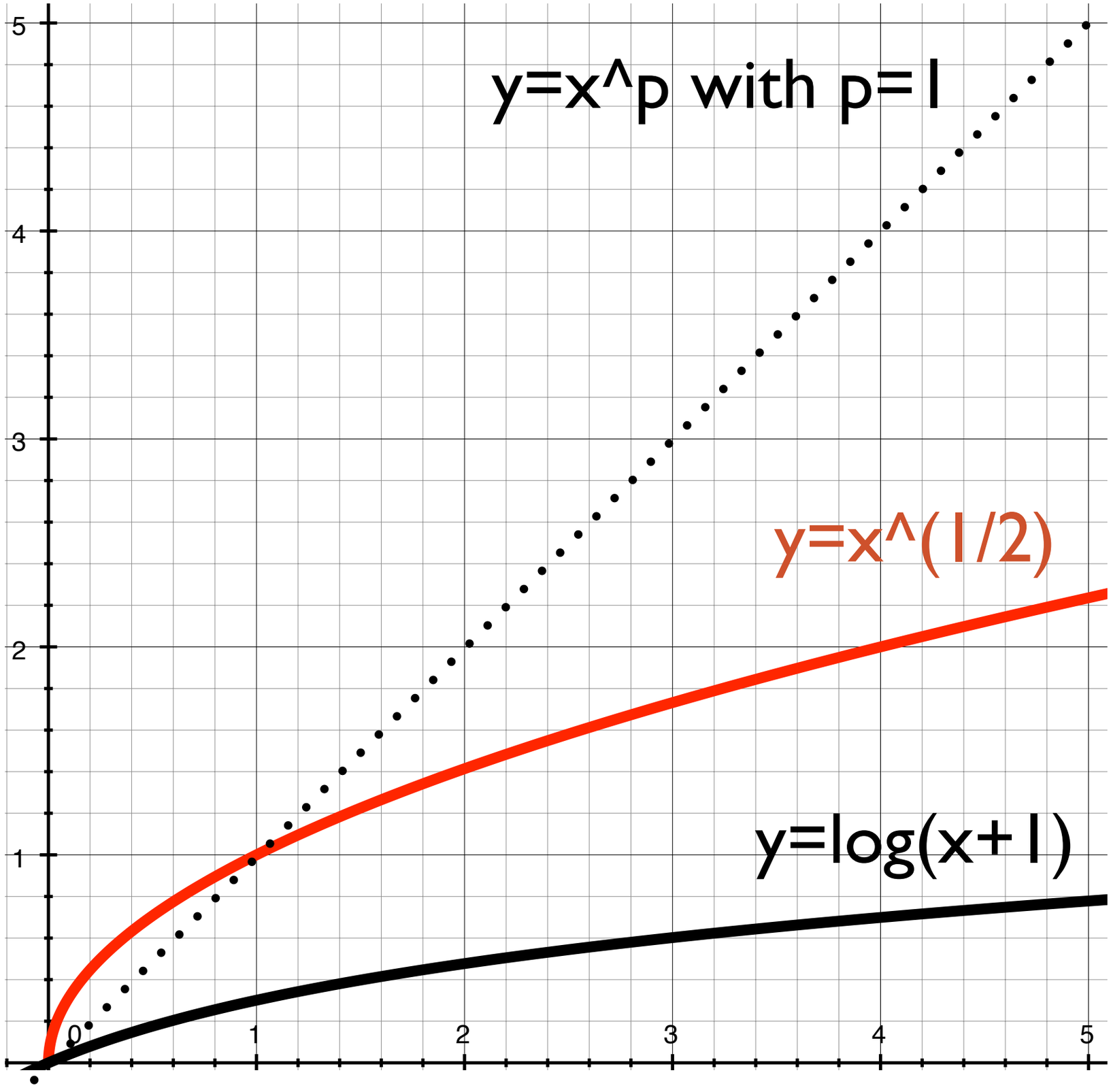




positive skew!

How Can we Fix Pos. Skew?

pull down
high values

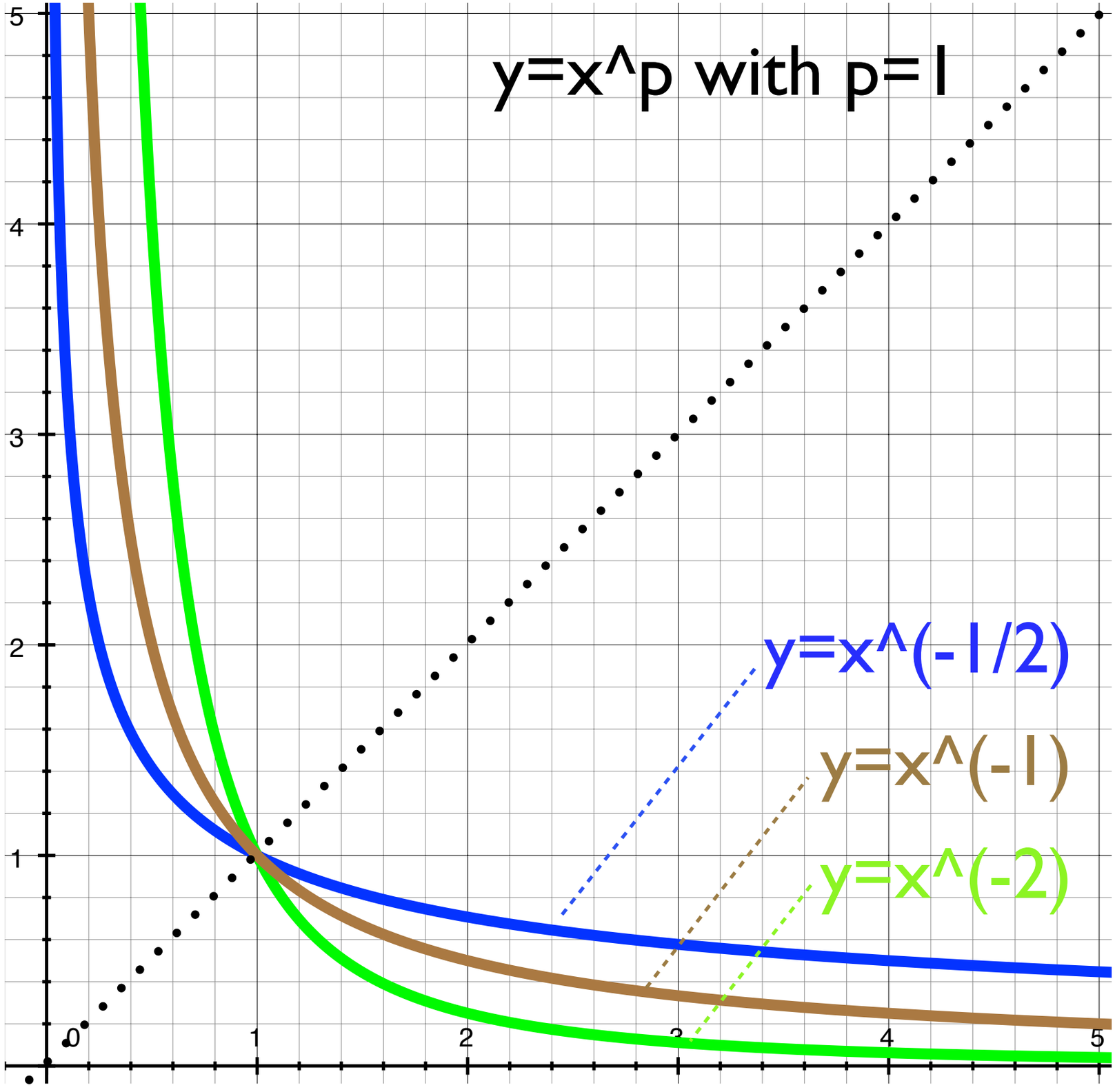


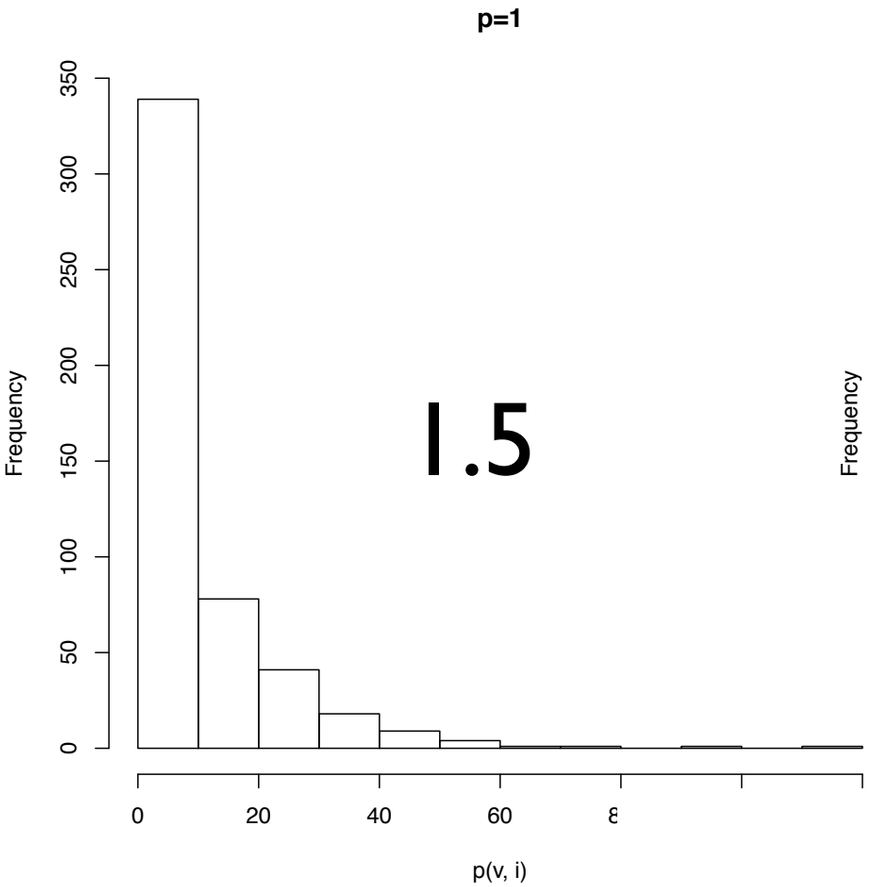
How Can we Fix Pos. Skew?

pull down
high values

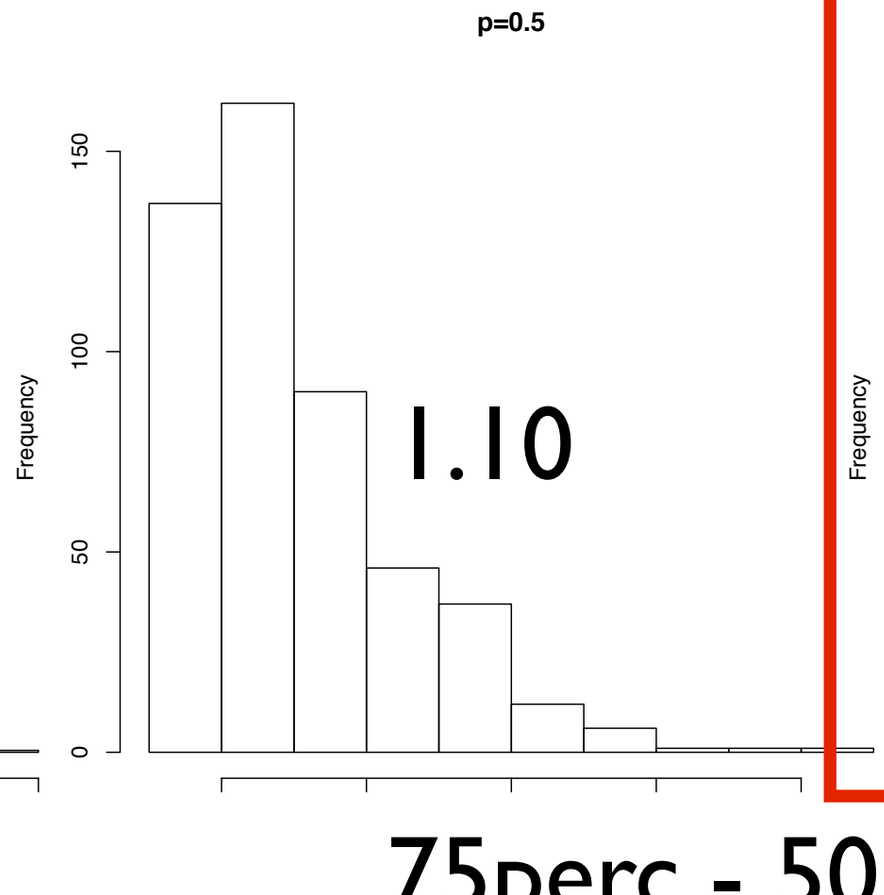
+

push up
low values

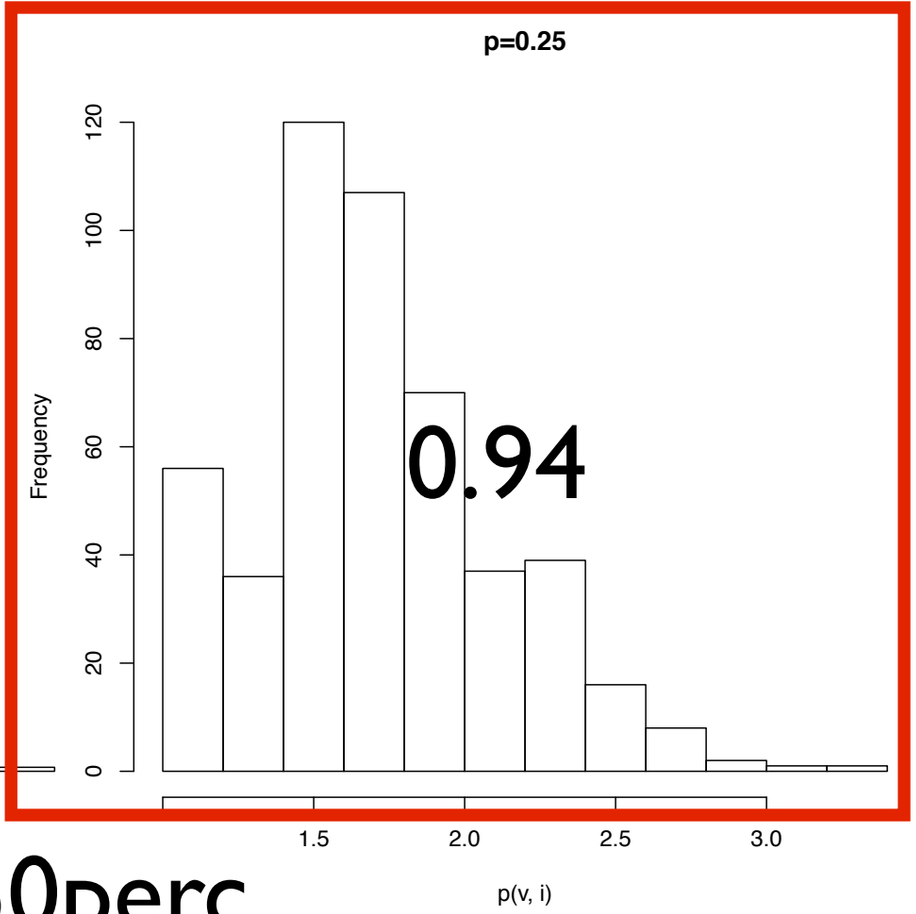




1.5

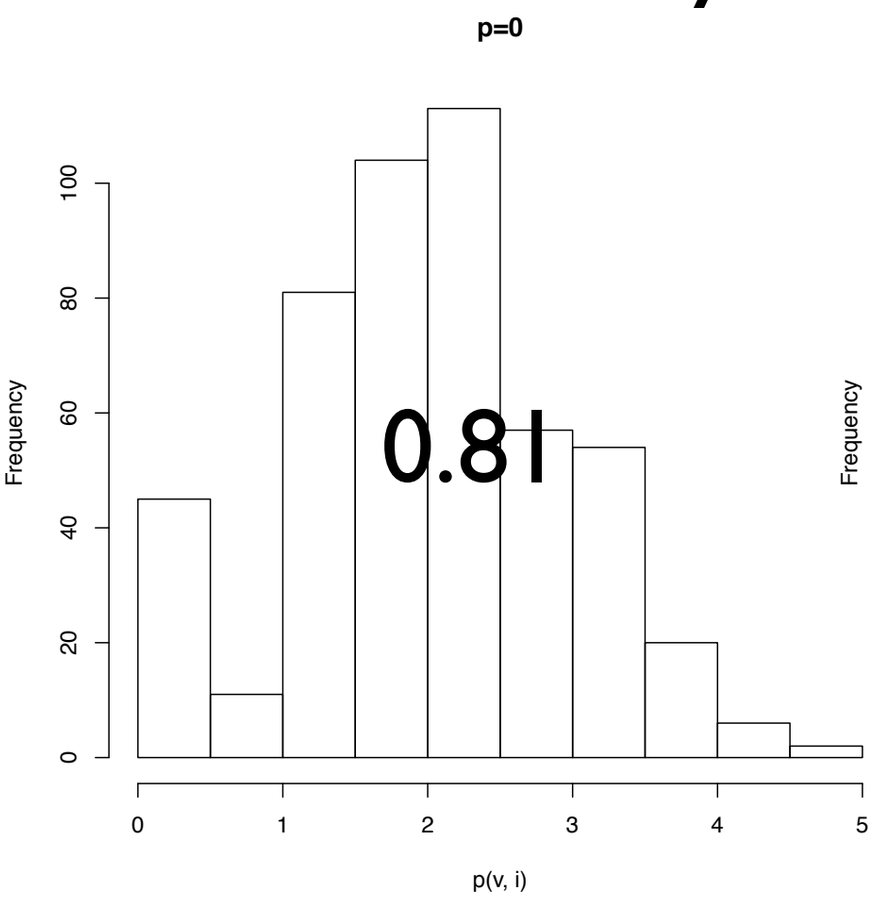


1.10

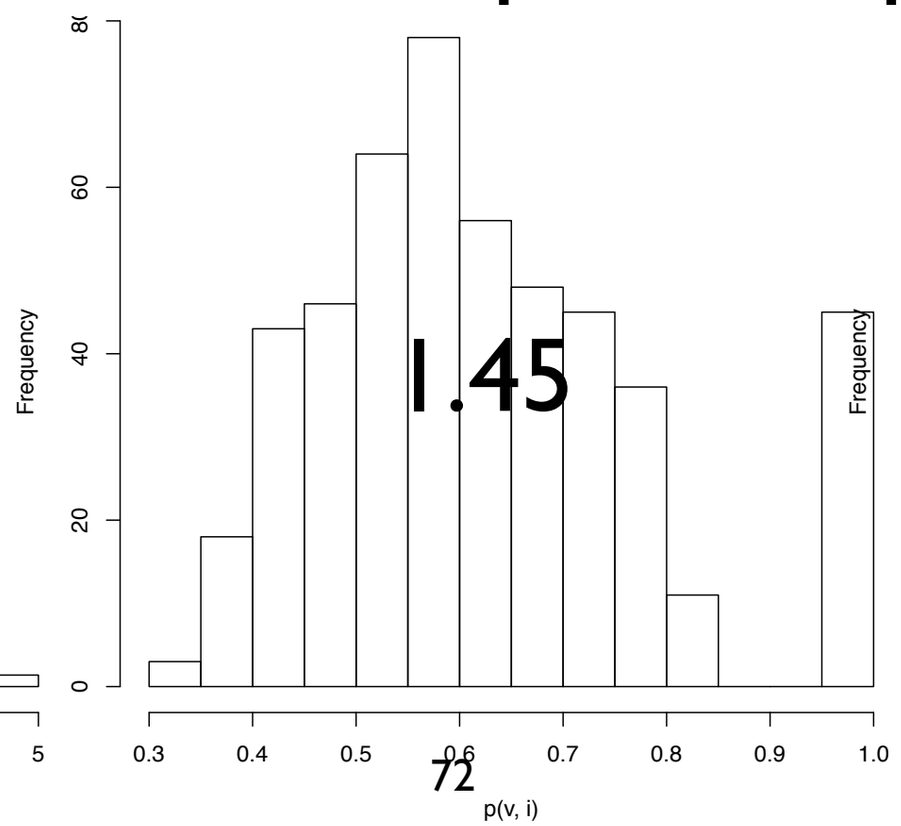


0.94

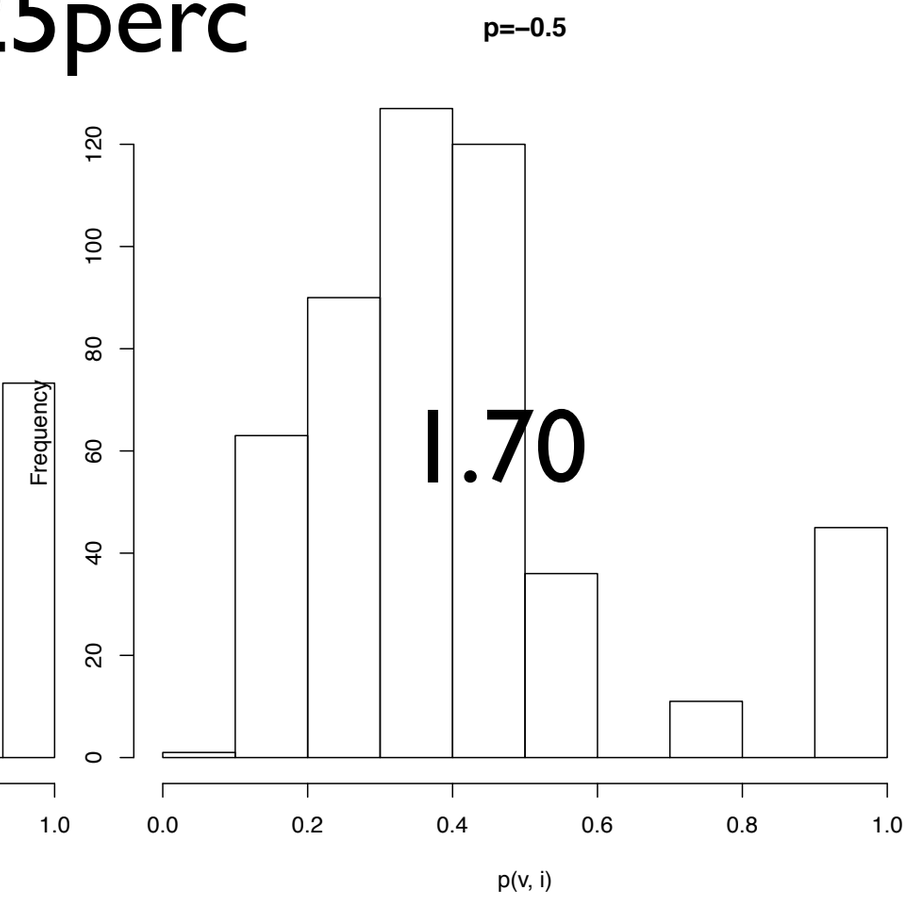
$$\text{symmetry} = \frac{\text{75perc - 50perc}}{\text{50perc - 25perc}}$$



0.81



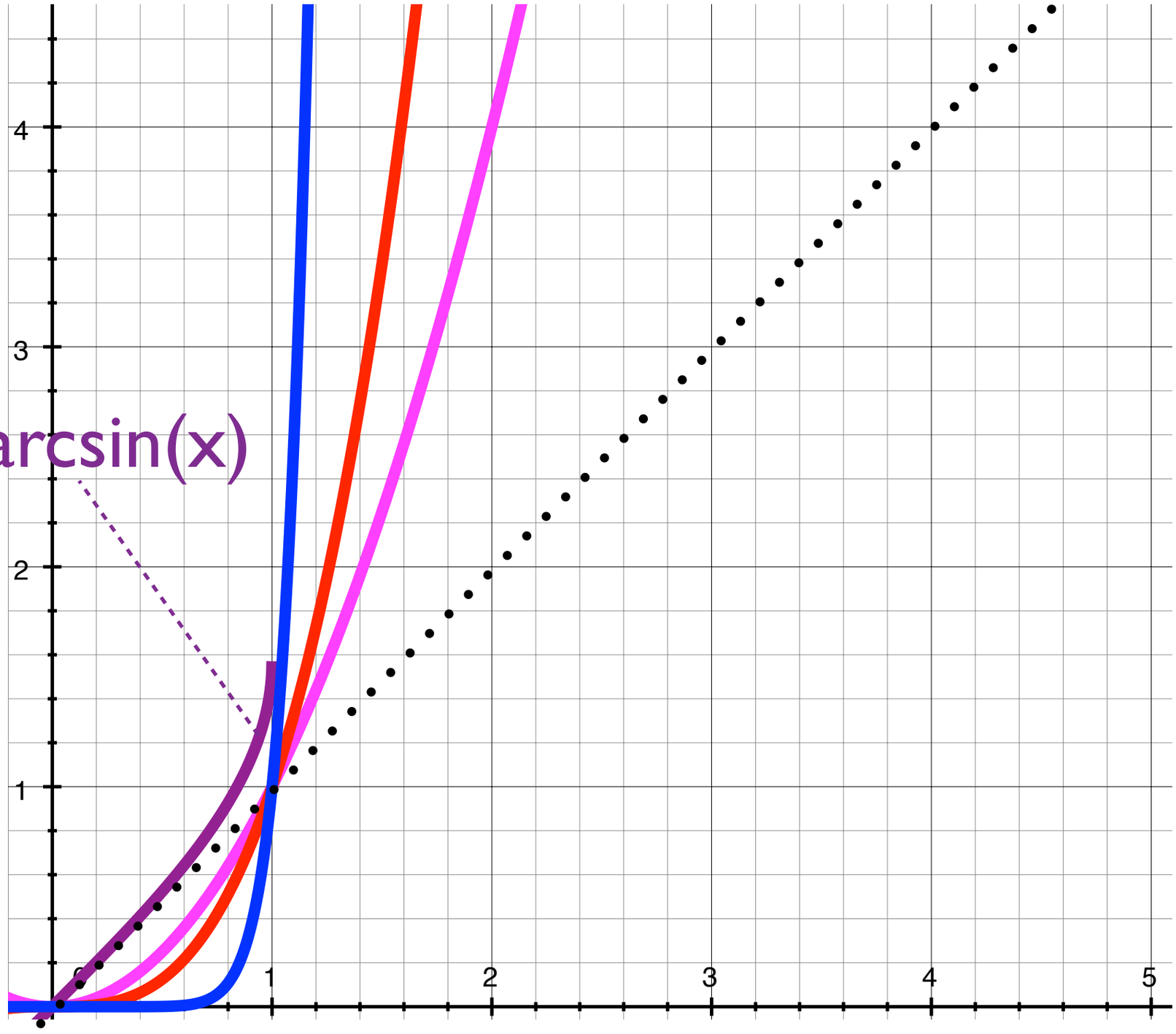
1.45



1.70

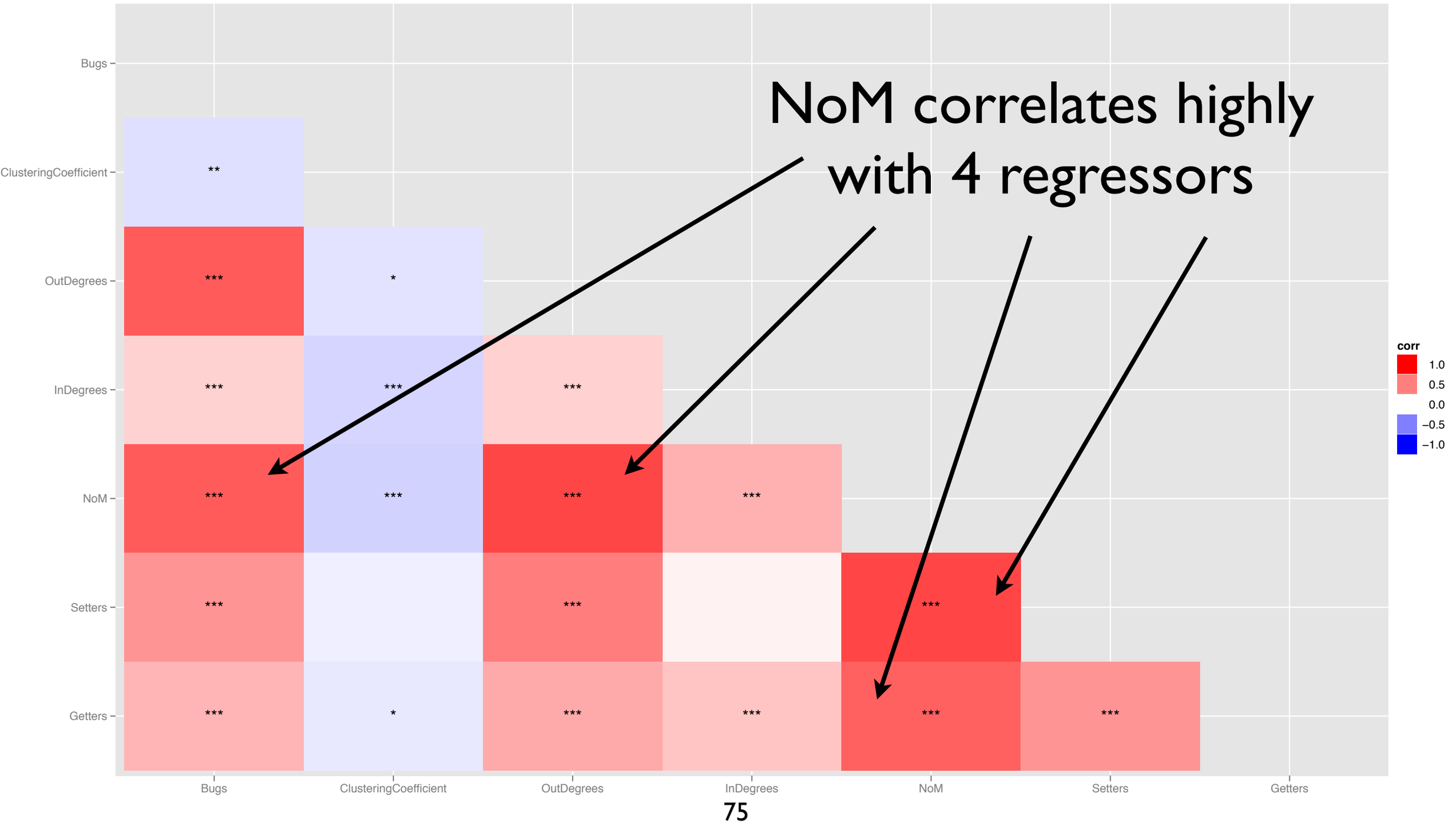
How Can we Fix Neg. Skew?

push up
high values



transformations can hamper the **interpretation** of models

Avoid Multi-collinear Regressors



R Console

~/workspace/sail/personal/presentations/2011/PASED/data

```
> summary(model)

Call:
glm(formula = Bugs > 0 ~ Type + log(Getters + 1) + log(Setters +
    1) + log(NoM + 1) + log(InDegrees + 1) + log(OutDegrees +
    1) + log(ClusteringCoefficient + 1), family = binomial(),
    data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1501  -0.6208  -0.3221   0.4882   3.2322

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                    -7.0641     0.7633  -9.255  < 2e-16 ***
TypeI                           2.0524     1.2188   1.684   0.0922 .
log(Getters + 1)               -0.2232     0.2088  -1.069   0.2852
log(Setters + 1)               -0.1021     0.2070  -0.493   0.6218
log(NoM + 1)                    1.5742     0.3479   4.525 6.03e-06 ***
log(InDegrees + 1)             -0.1110     0.1642  -0.676   0.4990
log(OutDegrees + 1)             1.0891     0.2713   4.015 5.95e-05 ***
log(ClusteringCoefficient + 1)  0.9283     0.7489   1.240   0.2151
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 582.68  on 492  degrees of freedom
Residual deviance: 396.28  on 485  degrees of freedom
AIC: 412.28

Number of Fisher Scoring iterations: 6

> vif(model)
                     TypeI               log(Getters + 1)               log(Setters + 1)                  log(NoM + 1)                        log
(InDegrees + 1)
                  2.685907                      1.691578                      2.461608                      7.779698
1.624055
          log(OutDegrees + 1) log(ClusteringCoefficient + 1)
                  2.625032                      1.109660
>
>
>
```

model containing **all** variables

Variation Inflation Factor>4

R Console

~/workspace/sail/personal/presentations/2011/PASED/data

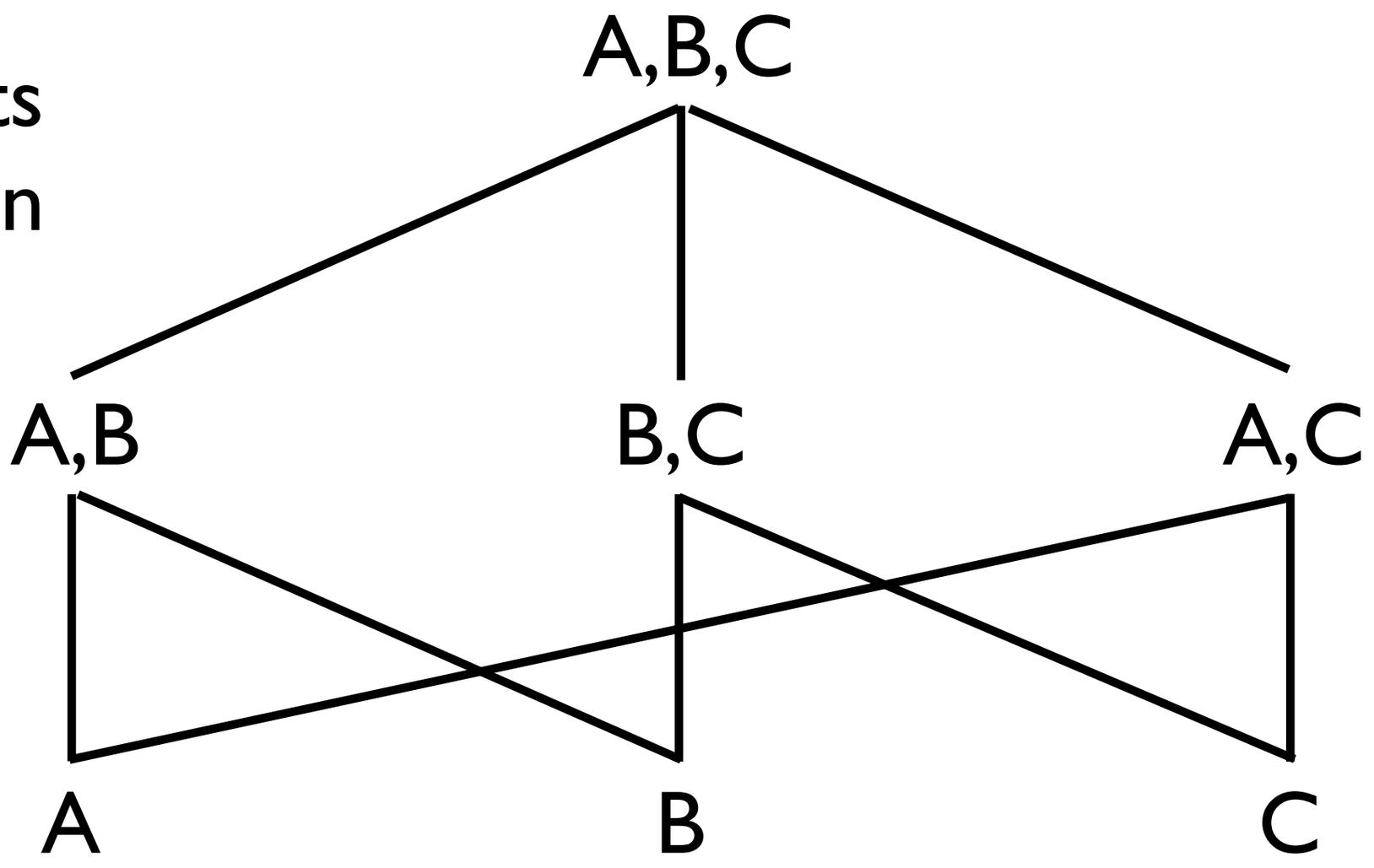
model **without** NoM

```
> summary(model)

Call:
glm(formula = Bugs > 0 ~ Type + log(Getters + 1) + log(Setters +
    1) + log(InDegrees + 1) + log(OutDegrees + 1) + log(ClusteringCoefficient +
    1), family = binomial(), data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3005  -0.6839  -0.3533   0.5721   2.9569

Coefficients:
                           Estimate Std. Error z value Pr(>|z|)
(Intercept)                 -5.4757     0.6217  -8.807  < 2e-16 ***
TypeI                       -0.3190     1.0830  -0.295  0.76835
log(Getters + 1)             0.1058     0.1924   0.550  0.58233
log(Setters + 1)             0.4552     0.1629   2.795  0.00519 **
log(InDegrees + 1)           0.1916     0.1413   1.356  0.17513
log(OutDegrees + 1)          1.6113     0.2378   6.776 1.23e-11 ***
log(ClusteringCoefficient + 1)   0.2741     0.7215   0.380  0.70405
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 582.68  on 492  degrees of freedom
Residual deviance: 418.68  on 486  degrees of freedom
AIC: 432.68

Number of Fisher Scoring iterations: 6

> vif(model)
                      TypeI           log(Getters + 1)           log(Setters + 1)          log(InDegrees + 1)                     log
(OutDegrees + 1)
                   1.450576                   1.485940                   1.655179                   1.387790
1.837712
log(ClusteringCoefficient + 1)
                   1.069934
```

all VIF values < 4

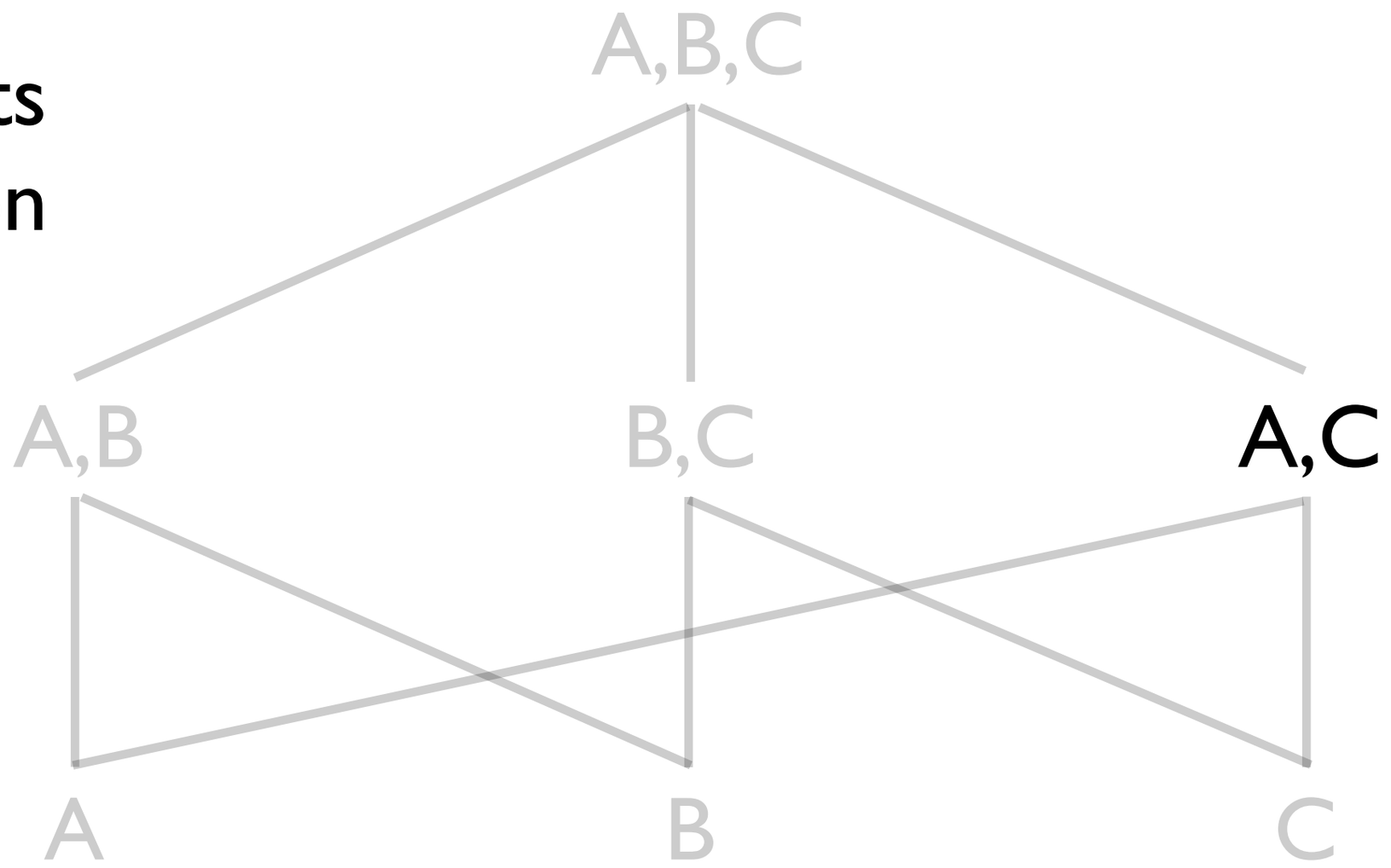
Variable Selection (2)

all-subsets regression



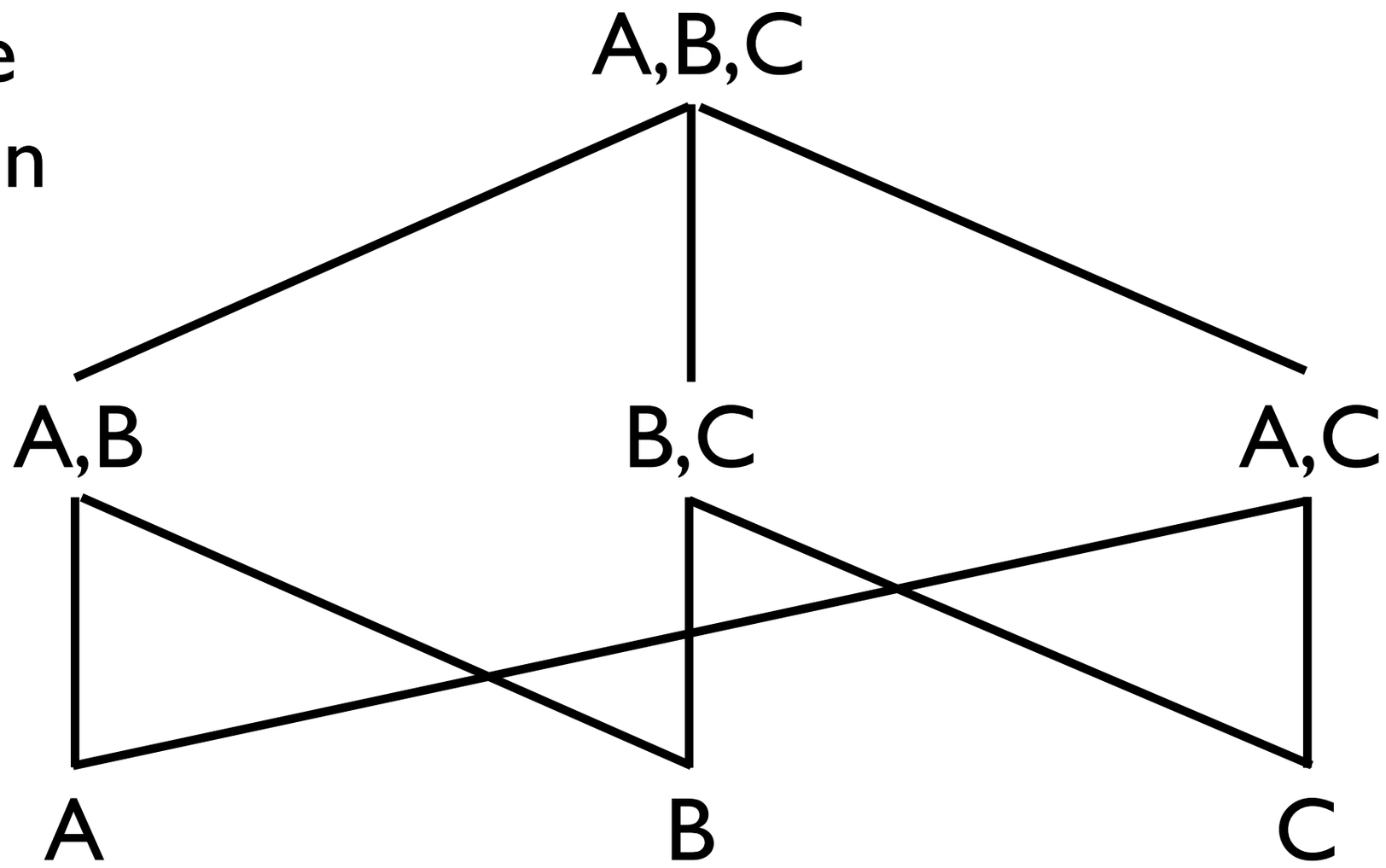
Variable Selection (2)

all-subsets regression



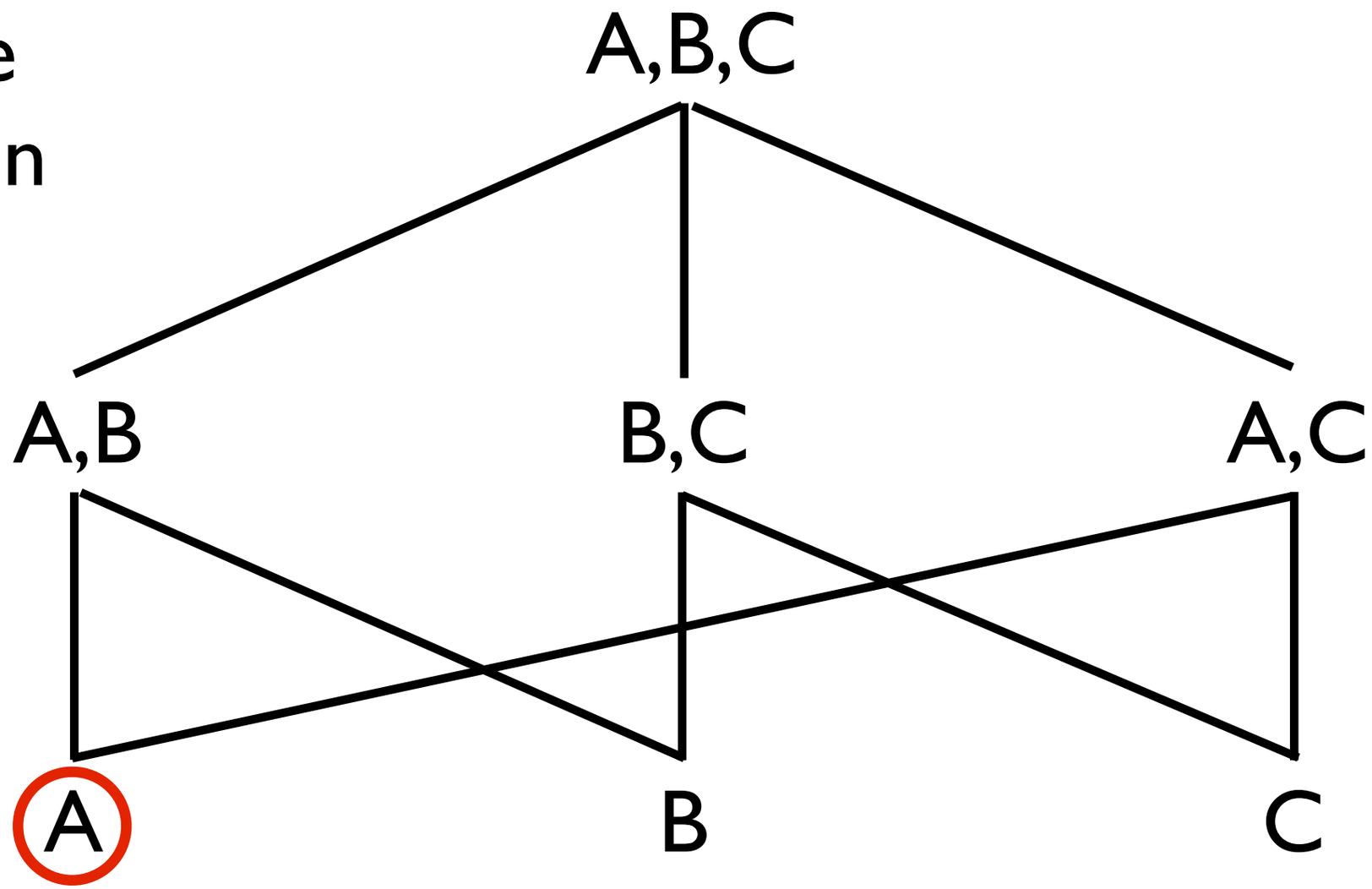
Variable Selection (2)

stepwise regression



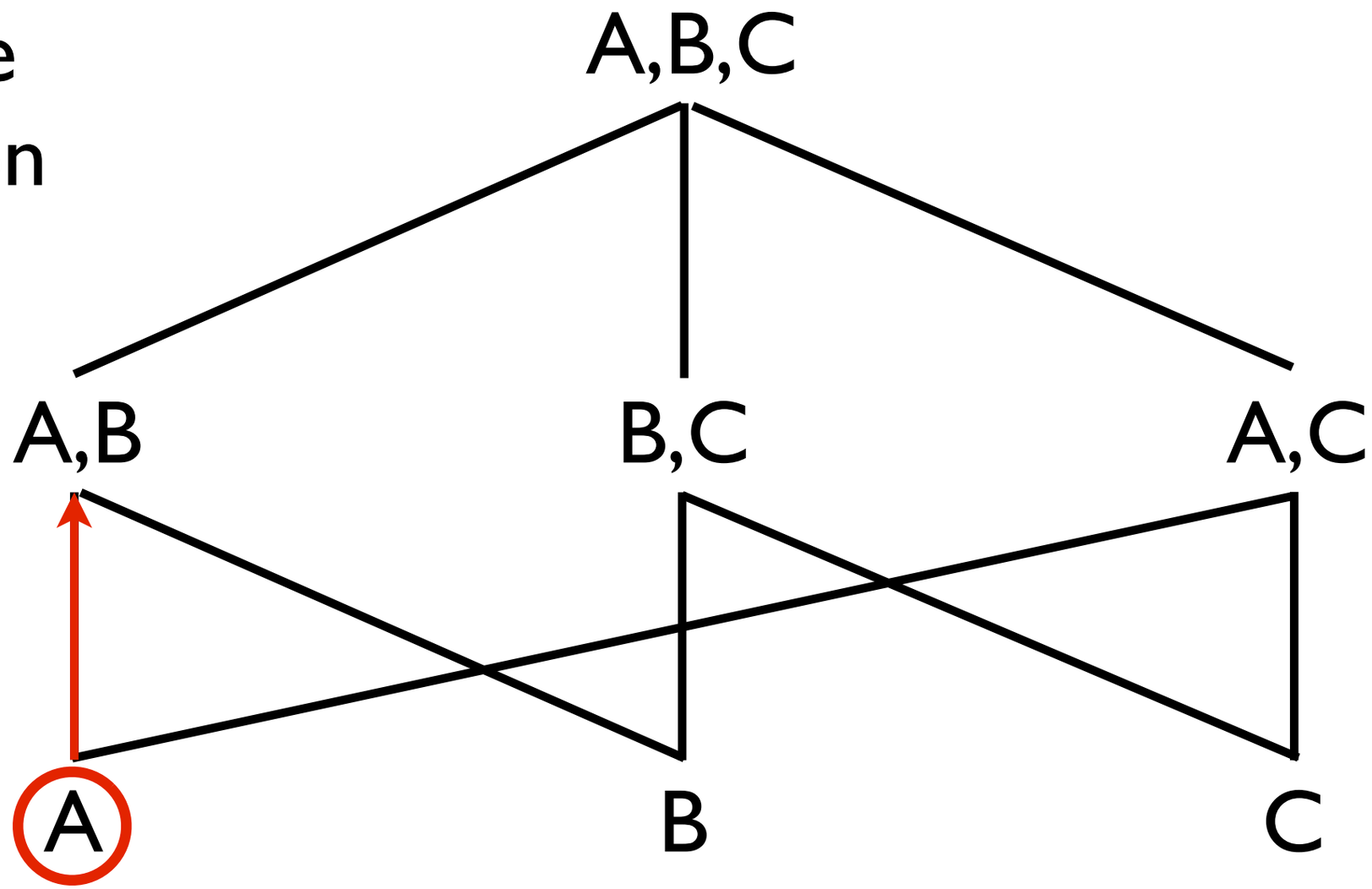
Variable Selection (2)

stepwise regression



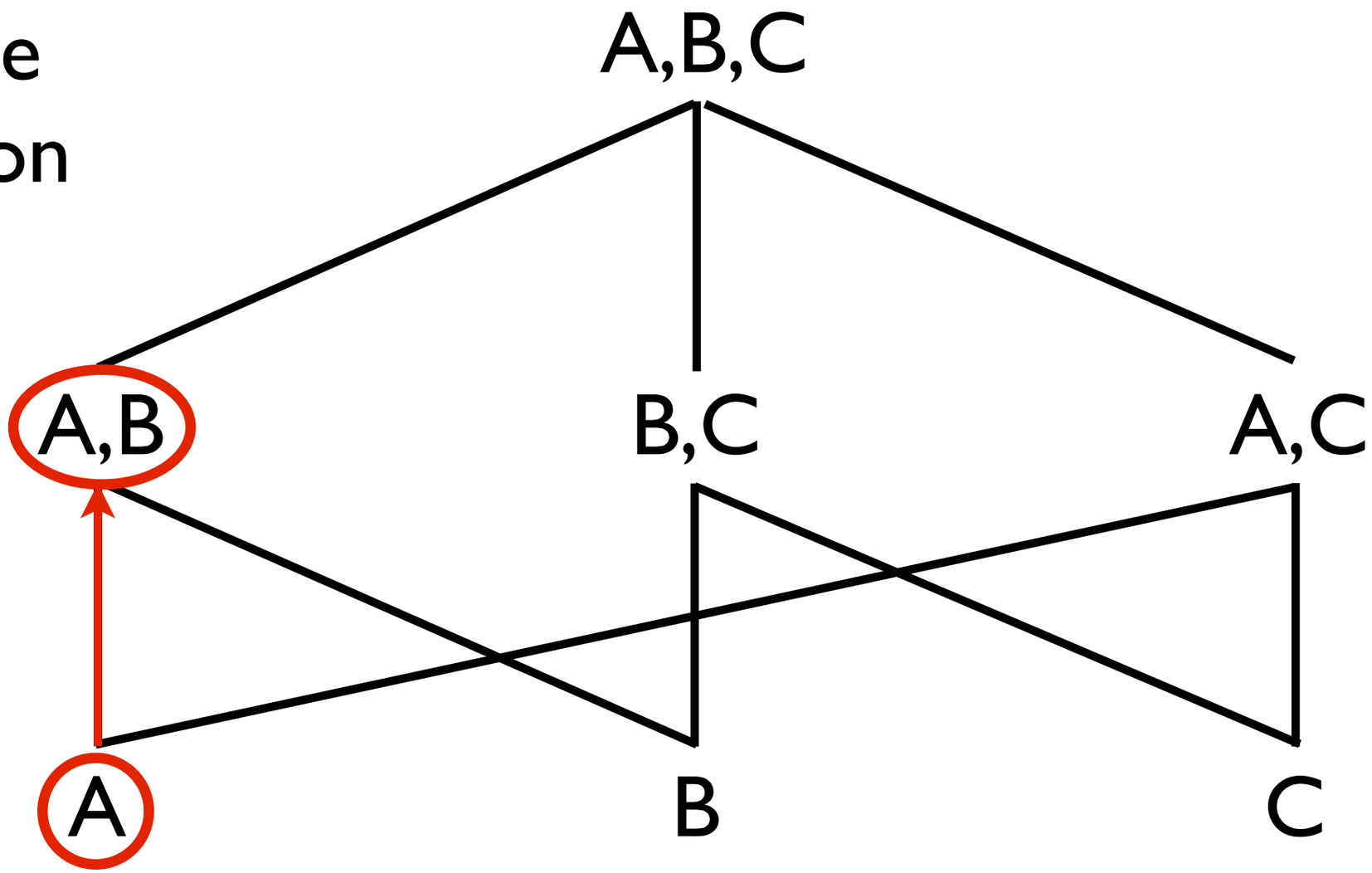
Variable Selection (2)

stepwise regression



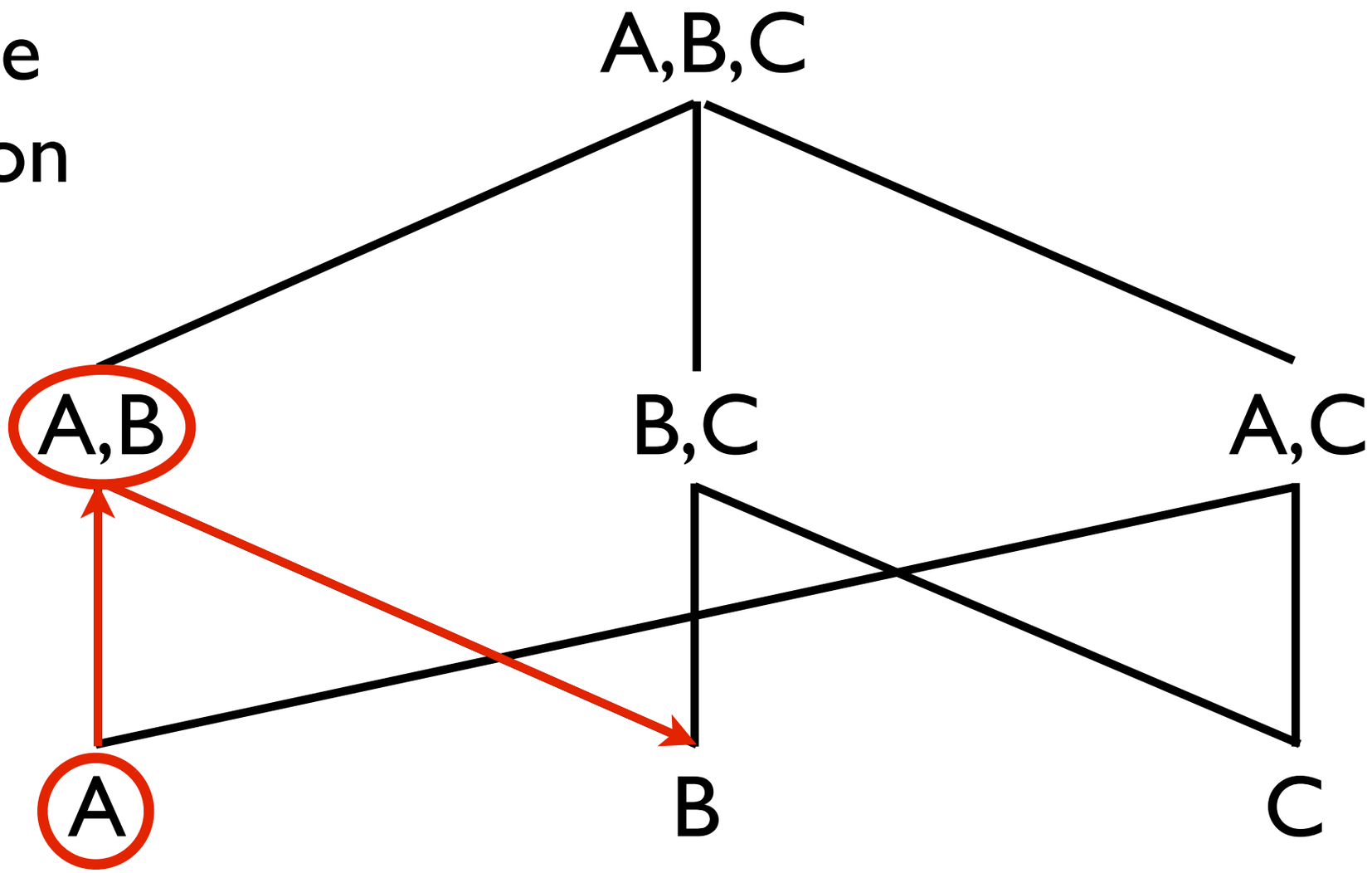
Variable Selection (2)

stepwise regression



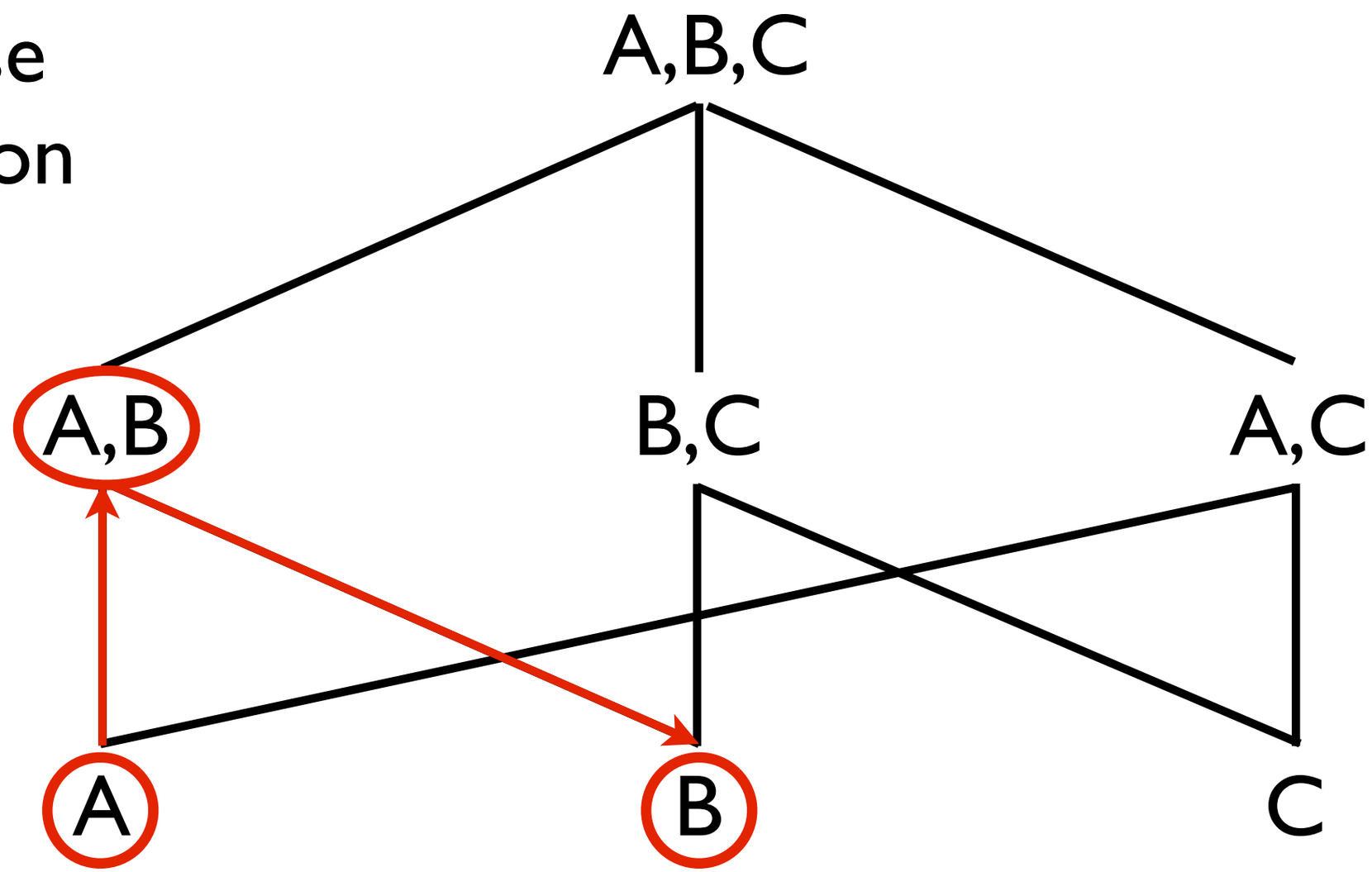
Variable Selection (2)

stepwise regression



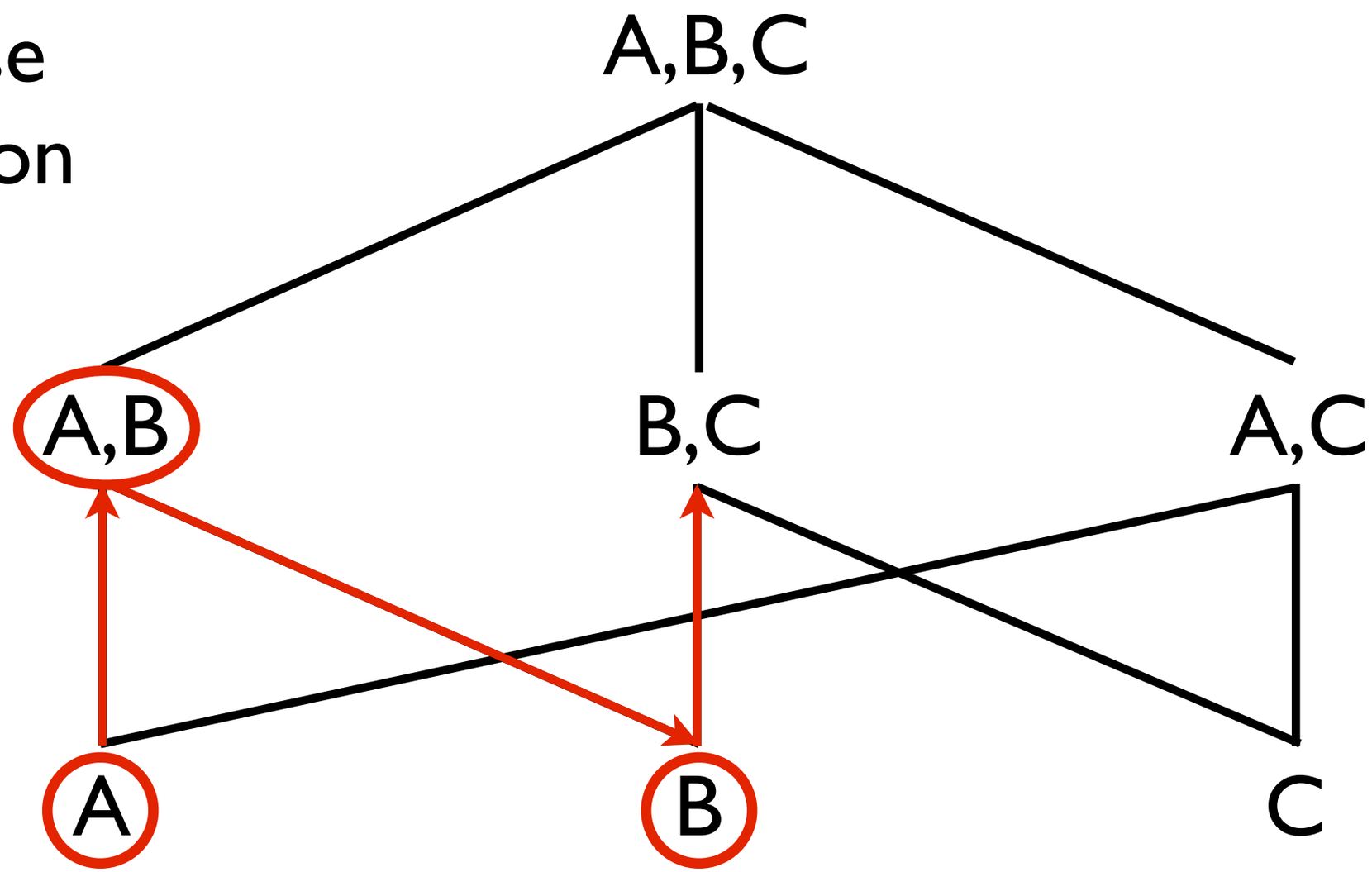
Variable Selection (2)

stepwise regression



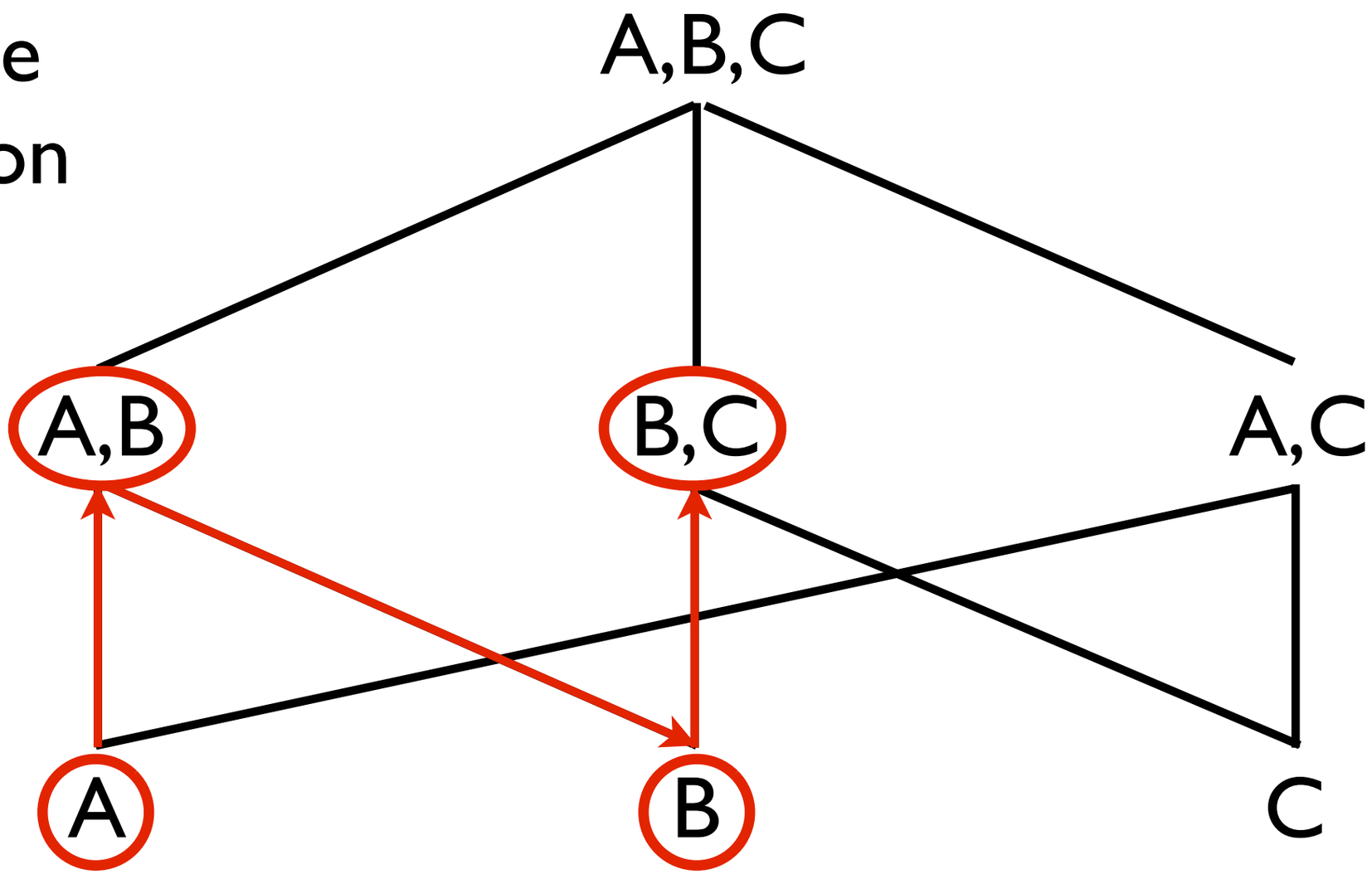
Variable Selection (2)

stepwise regression



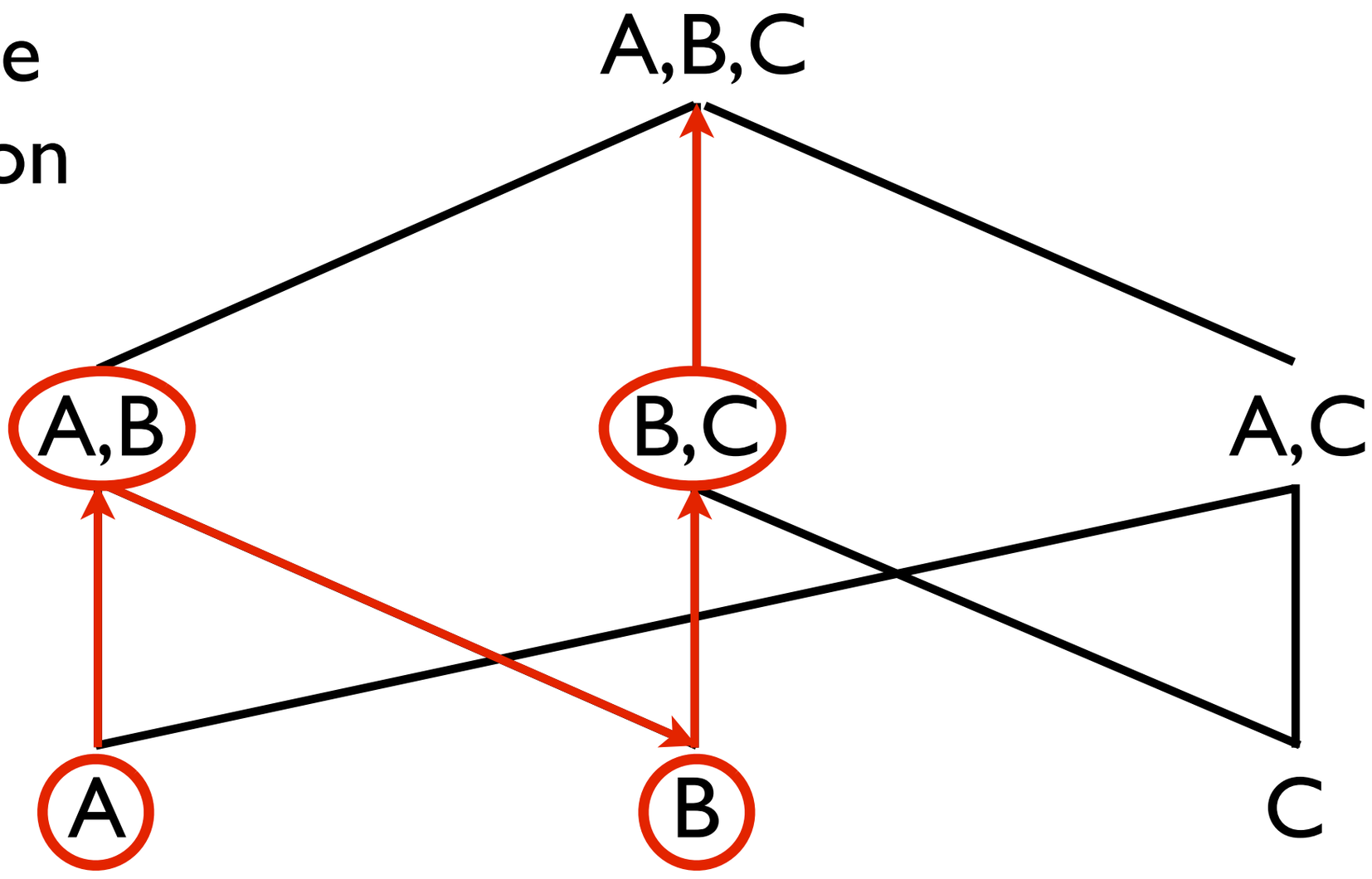
Variable Selection (2)

stepwise regression



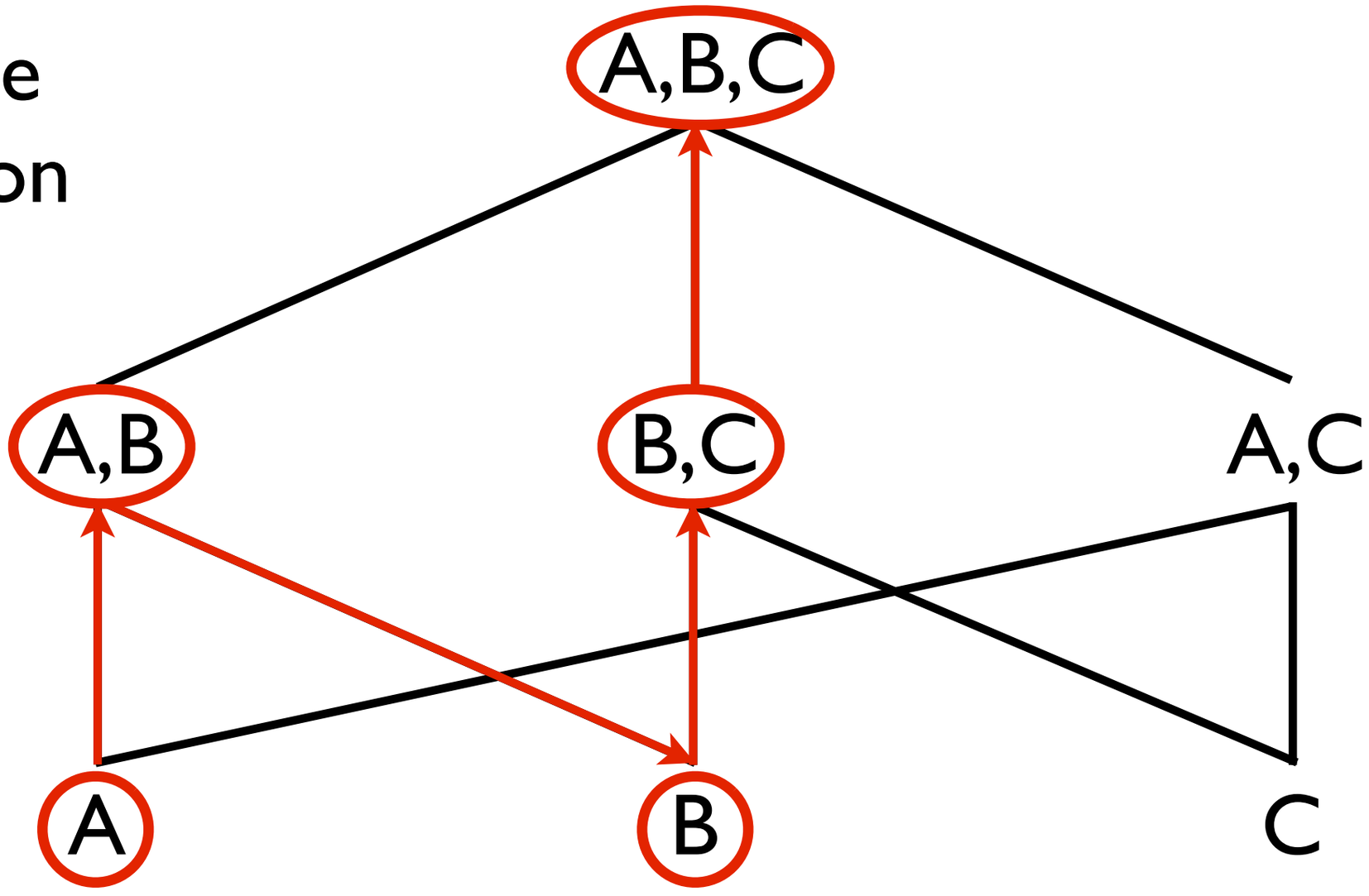
Variable Selection (2)

stepwise regression



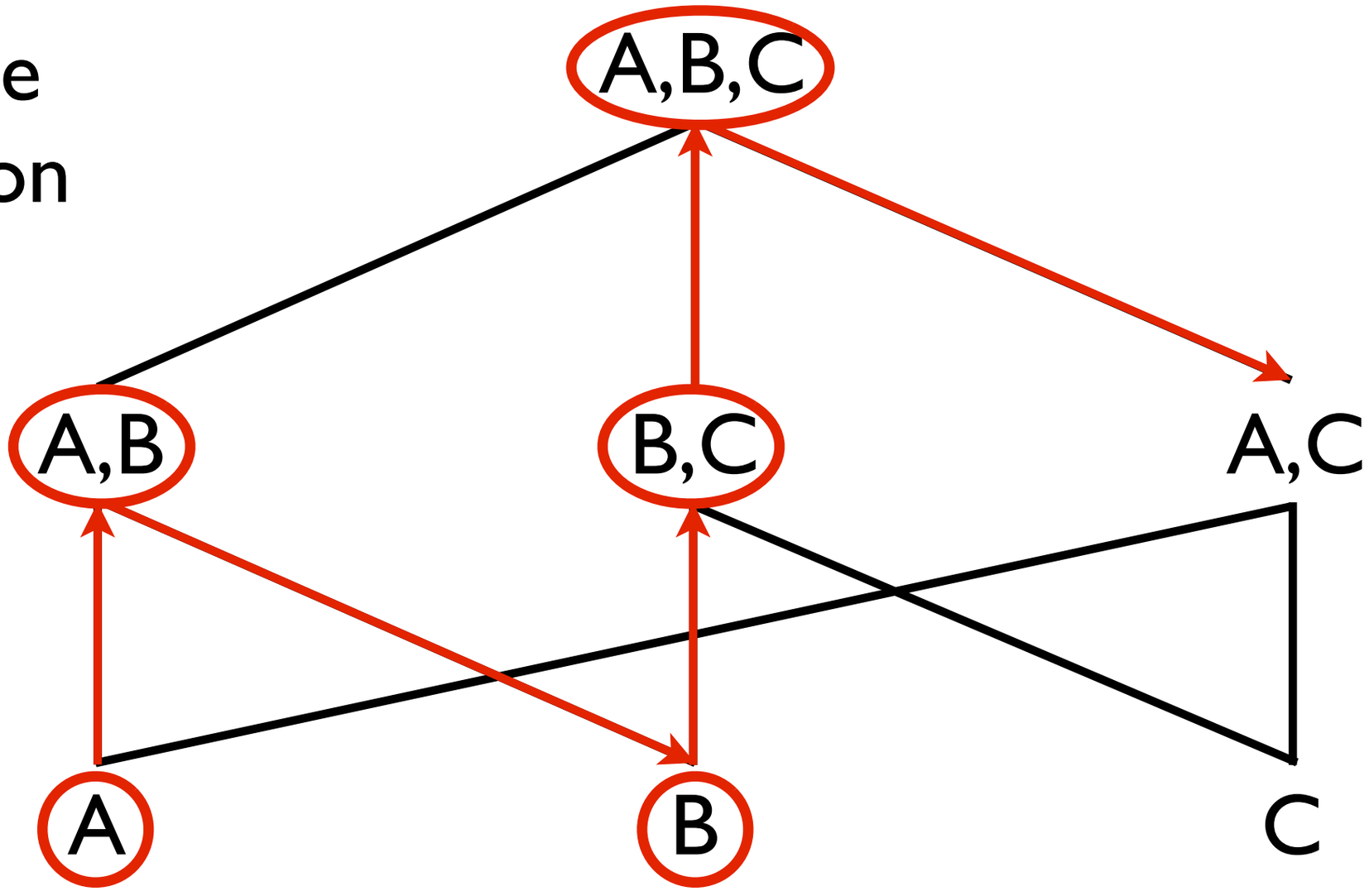
Variable Selection (2)

stepwise regression



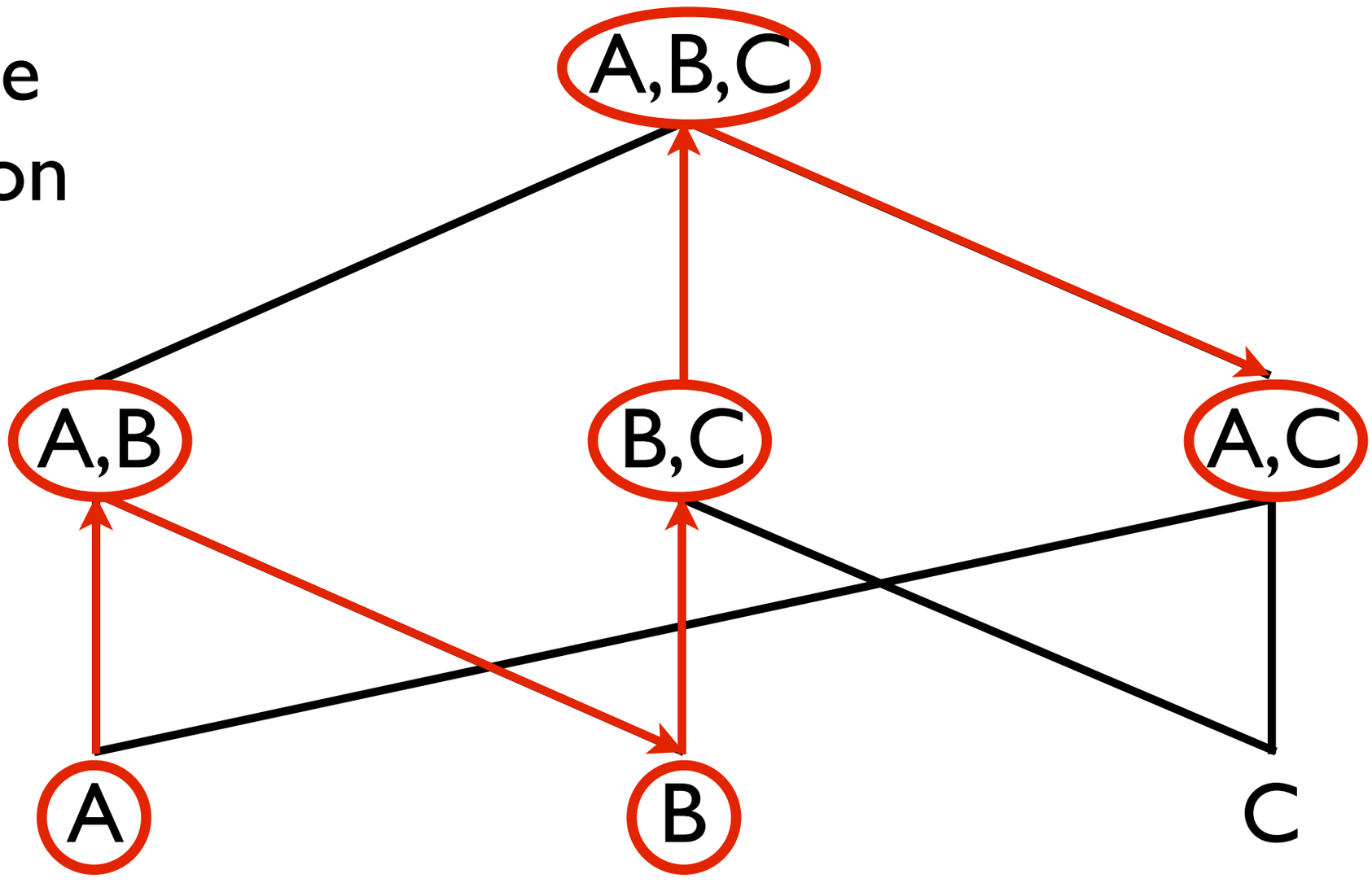
Variable Selection (2)

stepwise regression



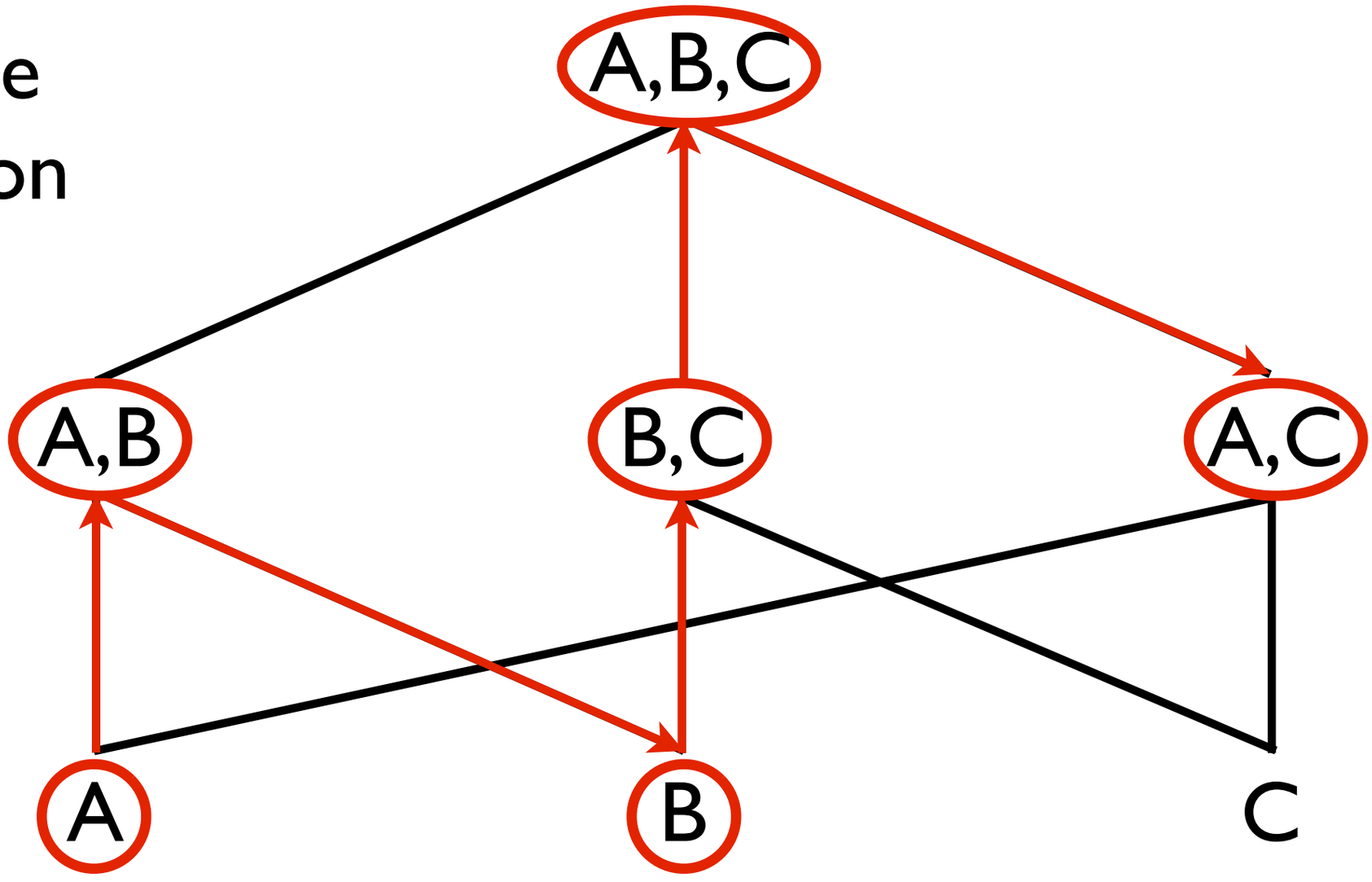
Variable Selection (2)

stepwise regression



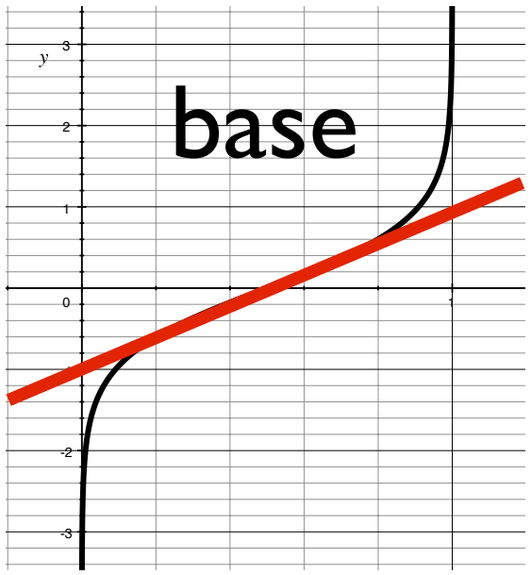
Variable Selection (2)

stepwise regression



Akaike Information Criterion (AIC): the smaller, the better performance with simpler model

Hierarchical Modeling



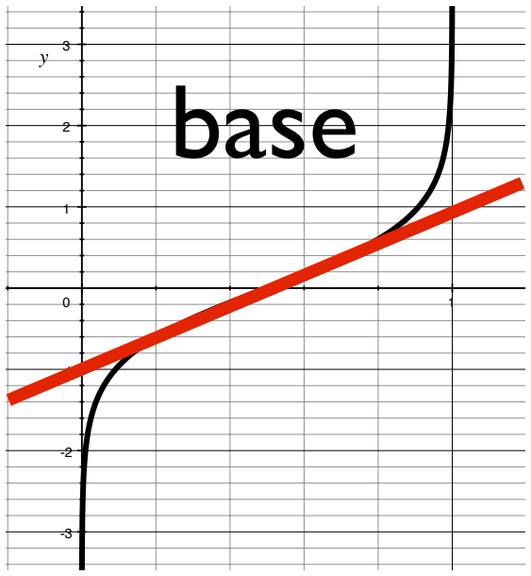
$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta Type_i$$

dummy variable

type C: $log(\frac{\pi_i}{1-\pi_i}) = \alpha$

type I: $log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta$

Hierarchical Modeling



what if Type has
3 possible values?

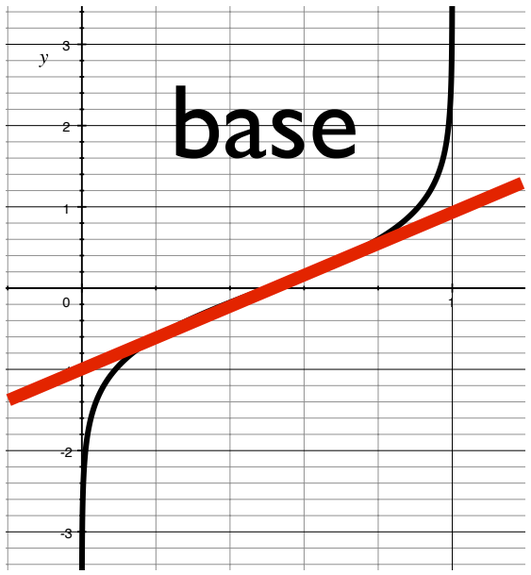
$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta.Type_i + \gamma.Type'_i$$

type C: $log(\frac{\pi_i}{1-\pi_i}) = \alpha$

type I: $log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta$

type A: $log(\frac{\pi_i}{1-\pi_i}) = \alpha + \gamma$

Hierarchical Modeling



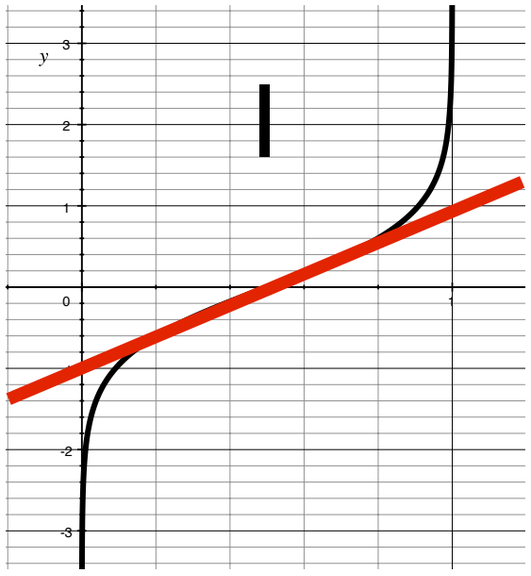
$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta.Type_i$$

```
Call:
glm(formula = (Bugs > 0) ~ Type, family = binomial(), data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.8518  -0.8518  -0.8518   1.5427   2.7672

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.8271     0.1028  -8.046 8.56e-16 ***
TypeI        -2.9795     1.0162  -2.932  0.00337 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 582.68  on 492  degrees of freedom
Residual deviance: 558.93 >> 491  degrees of freedom
AIC: 562.93
```

Deviance explained: 4.08%

overdispersion? quasibinomial

Hierarchical Modeling



$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta.Type_i$$

$$+\gamma.log(Getters_i + 1)$$

$$+\delta.log(Setters_i + 1)$$

```
Call:
glm(formula = (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters +
    1), family = binomial(), data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5231  -0.7734  -0.5315   0.7724   2.7672

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.8858     0.1882 -10.018  < 2e-16 ***
TypeI              -1.9208     1.0284  -1.868   0.0618 .
log(Getters + 1)    0.3166     0.1602   1.977   0.0481 *
log(Setters + 1)    0.8837     0.1396   6.331 2.44e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

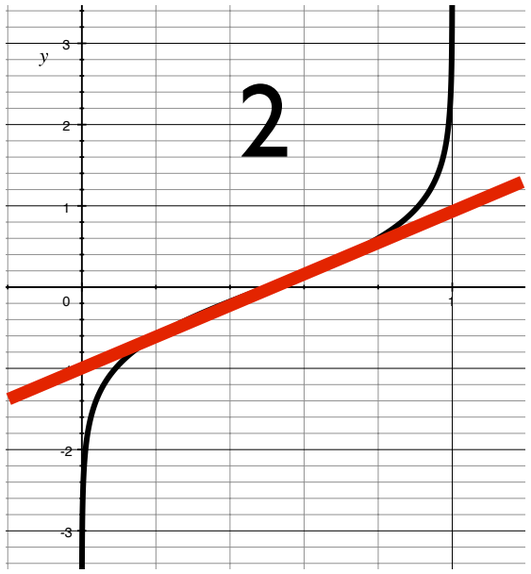
Deviance explained: 15.84%

```
> anova(baseline,model1, test="Chisq")
Analysis of Deviance Table

Model 1: (Bugs > 0) ~ Type
Model 2: (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters + 1)
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1       491     558.93
2       489     490.40  2   68.534 1.312e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> AIC(baseline,model1)
          df      AIC
baseline   2 562.9294
model1     4 498.3951
```

<

statistically significantly better model than baseline

Hierarchical Modeling



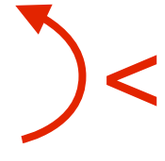
$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta.Type_i$$
$$+\gamma.log(Getters_i + 1)$$
$$+\delta.log(Setters_i + 1)$$
$$+\epsilon.log(InDegrees_i + 1)$$
$$+\zeta.log(OutDegrees_i + 1)$$

```
Call:
glm(formula = (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters +
    1) + log(InDegrees + 1) + log(OutDegrees + 1), family = binomial(),
    data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2984  -0.6779  -0.3461   0.5705   2.9239

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          -5.3767     0.5596  -9.608  < 2e-16 ***
TypeI                -0.3162     1.0828  -0.292  0.77024
log(Getters + 1)      0.1020     0.1920   0.531  0.59520
log(Setters + 1)      0.4553     0.1630   2.793  0.00522 **
log(InDegrees + 1)    0.1761     0.1349   1.306  0.19155
log(OutDegrees + 1)   1.6101     0.2376   6.777 1.23e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Deviance explained: 28.12%

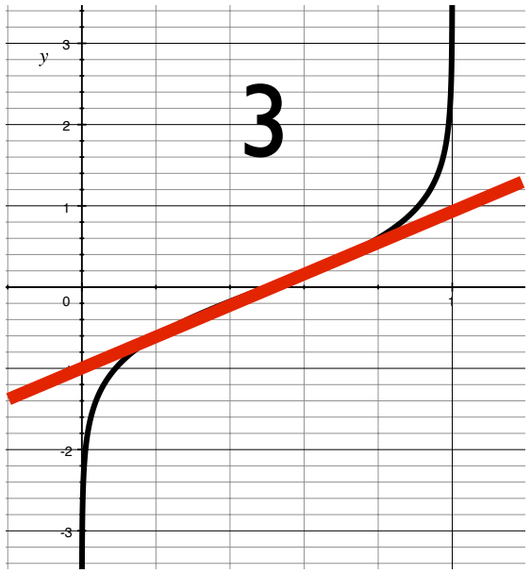
```
> anova(model1,model2, test="Chisq")
Analysis of Deviance Table

Model 1: (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters + 1)
Model 2: (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters + 1) +
    log(InDegrees + 1) + log(OutDegrees + 1)
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1       489     490.40
2       487     418.82  2   71.575 2.868e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
> AIC(model1,model2)
       df      AIC
model1  4 498.3951
model2  6 430.8198
```



statistically significantly better model than model 1

Hierarchical Modeling



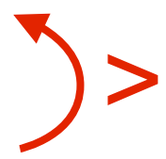
$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \beta.Type_i$$
$$+\gamma.log(Getters_i + 1)$$
$$+ \delta.log(Setters_i + 1)$$
$$+\epsilon.log(InDegrees_i + 1)$$
$$+ \zeta.log(OutDegrees_i + 1)$$
$$+\eta.log(Clustering_i + 1)$$

```
glm(formula = (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters +
    1) + log(InDegrees + 1) + log(OutDegrees + 1) + log(Clustering +
    1), family = binomial(), data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3005  -0.6839  -0.3533   0.5721   2.9569

Coefficients:
                                 Estimate Std. Error z value Pr(>|z|)
(Intercept)                       -5.4757     0.6217  -8.807  < 2e-16 ***
TypeI                             -0.3190     1.0830  -0.295  0.76835
log(Getters + 1)                   0.1058     0.1924   0.550  0.58233
log(Setters + 1)                   0.4552     0.1629   2.795  0.00519 **
log(InDegrees + 1)                 0.1916     0.1413   1.356  0.17513
log(OutDegrees + 1)                1.6113     0.2378   6.776 1.23e-11 ***
log(ClusteringCoefficient + 1)     0.2741     0.7215   0.380  0.70405
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Deviance explained: 28.15%

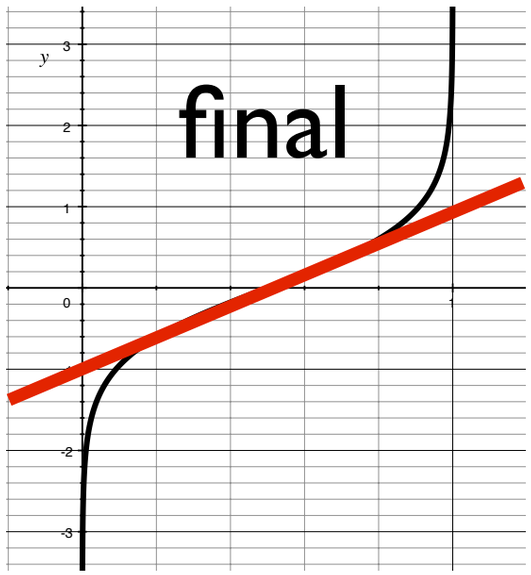
```
> anova(model2,model3, test="Chisq")
Analysis of Deviance Table

Model 1: (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters + 1) +
    log(InDegrees + 1) + log(OutDegrees + 1)
Model 2: (Bugs > 0) ~ Type + log(Getters + 1) + log(Setters + 1) +
    log(InDegrees + 1) + log(OutDegrees + 1) + log(Clustering + 1)
  Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1       487     418.82
2       486     418.68  1  0.14438      0.704
```

```
> AIC(model2,model3)
       df      AIC
model2  6 430.8198
model3  7 432.6755
```



no statistically significant improvement compared to model 2

Hierarchical Modeling

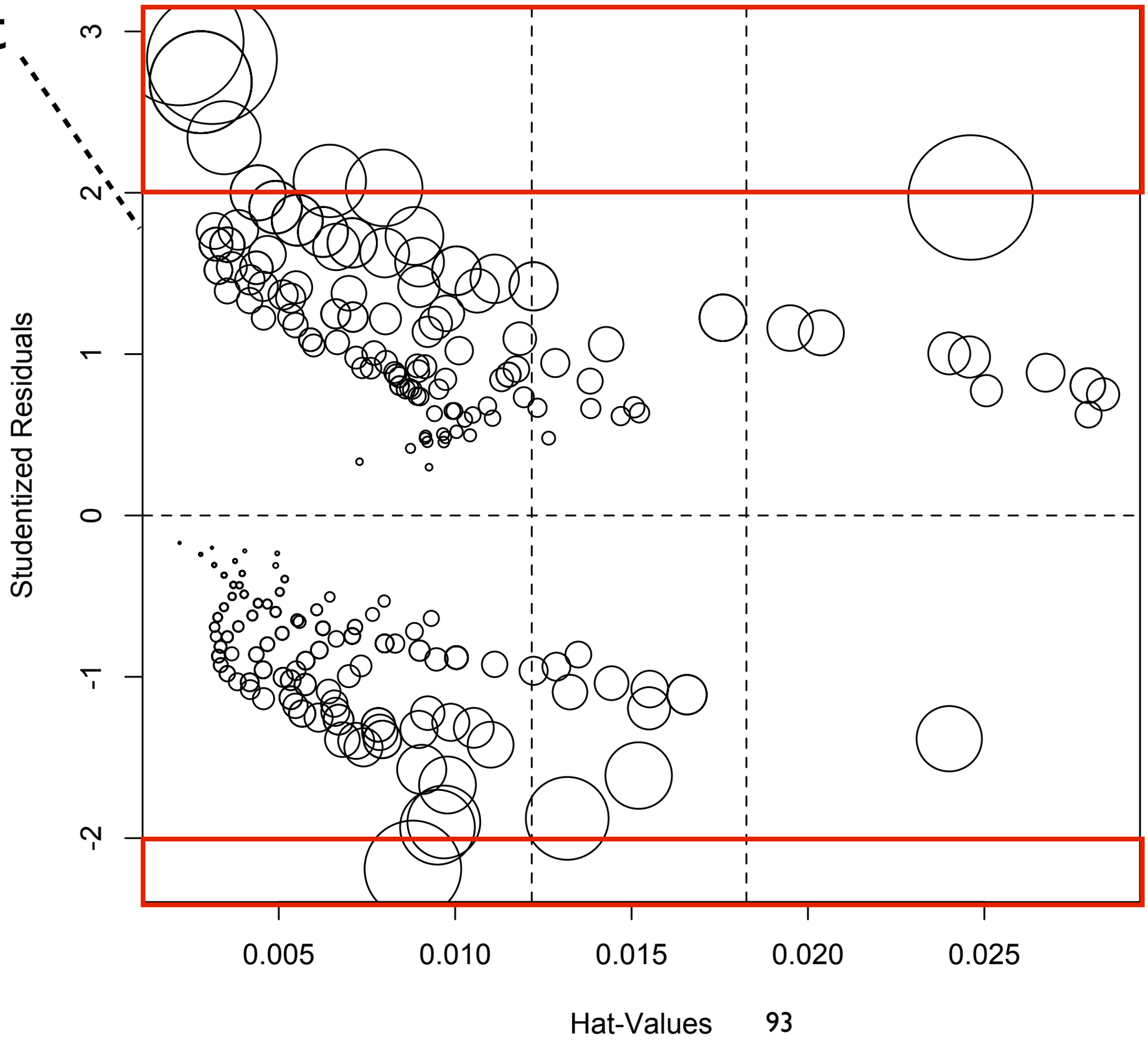


$$log(\frac{\pi_i}{1-\pi_i}) = \alpha + \delta.log(Setters_i + 1) + \zeta.log(OutDegrees_i + 1)$$

```
Call:
glm(formula = (Bugs > 0) ~ log(Setters + 1) + log(OutDegrees +
    1), family = binomial(), data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1712  -0.6978  -0.3592   0.6010   2.9131

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)         -5.4239     0.5325 -10.186  < 2e-16 ***
log(Setters + 1)     0.4686     0.1497   3.130  0.00175 **
log(OutDegrees + 1)  1.7245     0.2213   7.794  6.5e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

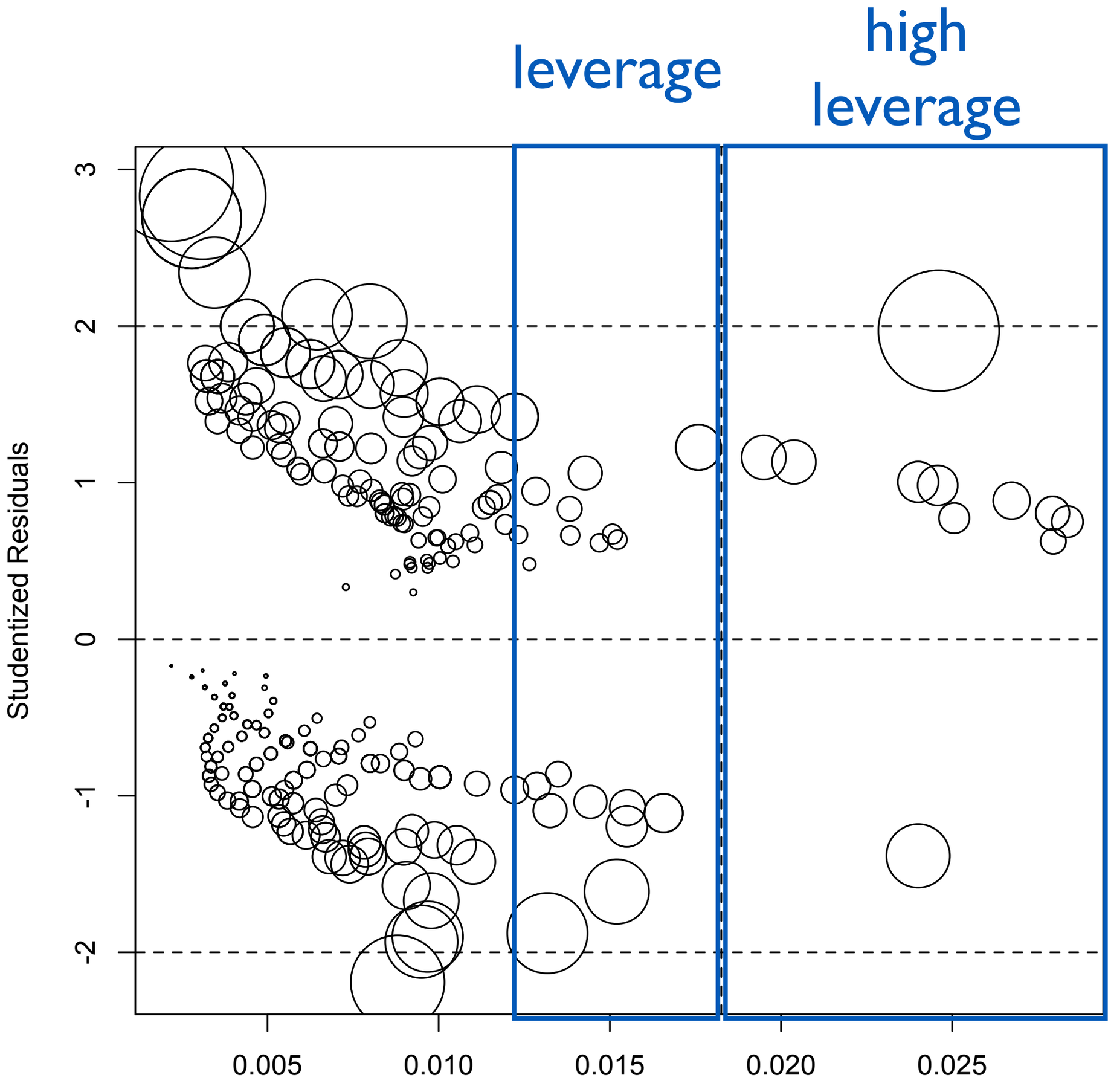
Deviance explained: 27.61%

influence plot

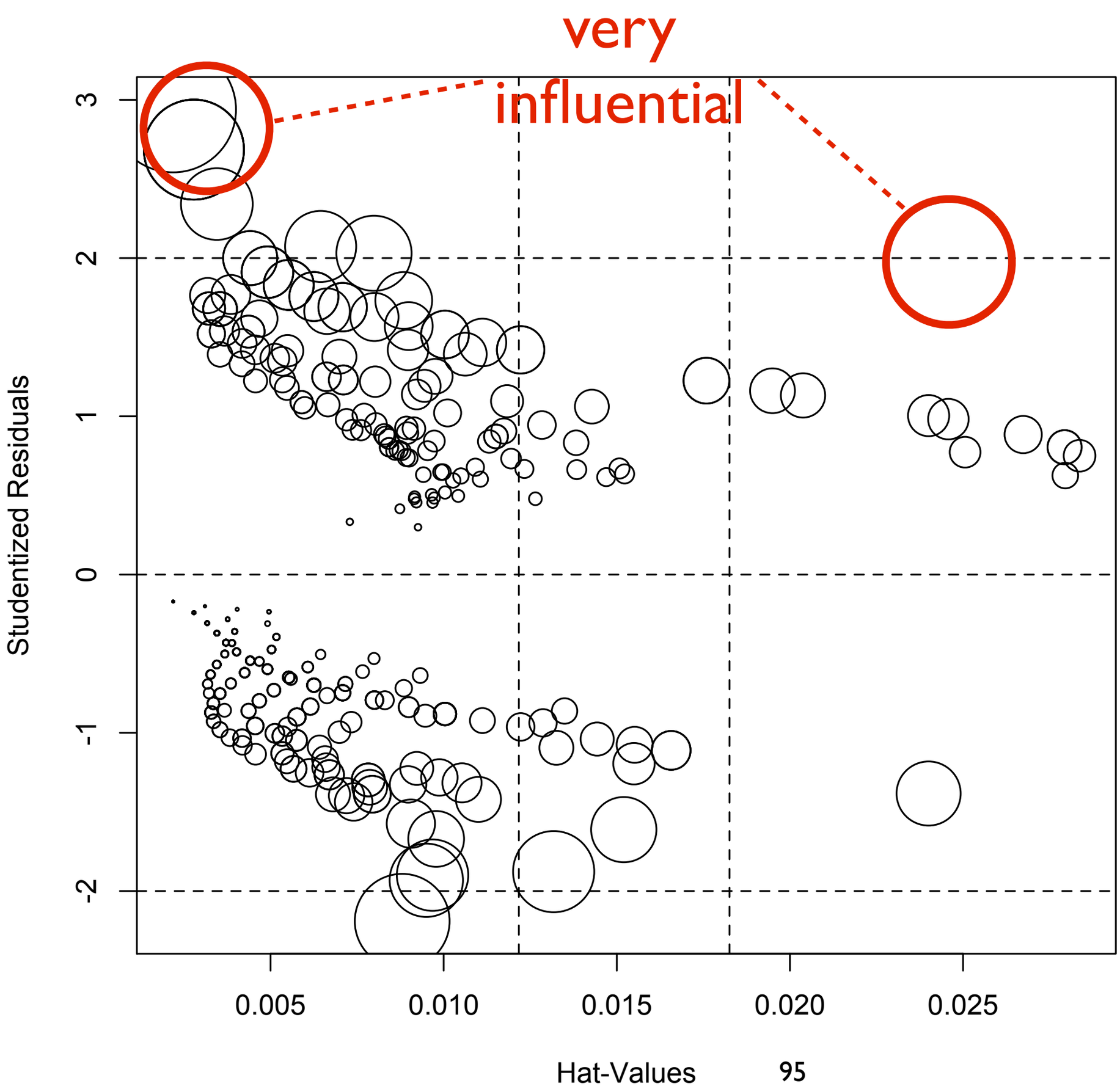


leverage

high leverage



leave out 4 most influential instances



```
Call:
glm(formula = (Bugs > 0) ~ log(Setters + 1) + log(OutDegrees +
    1), family = binomial(), data = d1[-c(66, 79, 157, 329),
    ])

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2925  -0.6655  -0.3220   0.4544   2.8259

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          -6.2122     0.6034 -10.295  < 2e-16 ***
log(Setters + 1)      0.4194     0.1550   2.705  0.00683 **
log(OutDegrees + 1)   2.0371     0.2468   8.255  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Deviance explained: 31.4%

statistically significantly better than final model

Interpretation of Model

$$\frac{\pi_i}{1-\pi_i} = e^{(\alpha+\delta.log(Setters_i+1)+\zeta.log(OutDegrees_i+1))}$$

$$= e^{\alpha} \times (e^{\delta})^{log(Setters_i+1)} \times (e^{\zeta})^{log(OutDegrees_i+1)}$$

$$= 0.002 \times 1.52^{log(Setters_i+1)} \times 7.67^{log(OutDegrees_i+1)}$$

odds ratios >1 => increasing effect

$$= 0.002 \times (Setters_i + 1)^{0.4194} \times (OutDegrees_i + 1)^{2.0371}$$

Interpretation of Model

$$\frac{\pi_i}{1-\pi_i} = e^{(\alpha+\delta.log(Setters_i+1)+\zeta.log(OutDegrees_i+1))}$$

$$= e^{\alpha} \times (e^{\delta})^{log(Setters_i+1)} \times (e^{\zeta})^{log(OutDegrees_i+1)}$$

$$= 0.002 \times 1.52^{log(Setters_i+1)} \times 7.67^{log(OutDegrees_i+1)}$$

odds ratios >1 => increasing effect

$$= 0.002 \times (Setters_i + 1)^{0.4194} \times (OutDegrees_i + 1)^{2.0371}$$

do **NOT** compare coefficients
(unless same units)

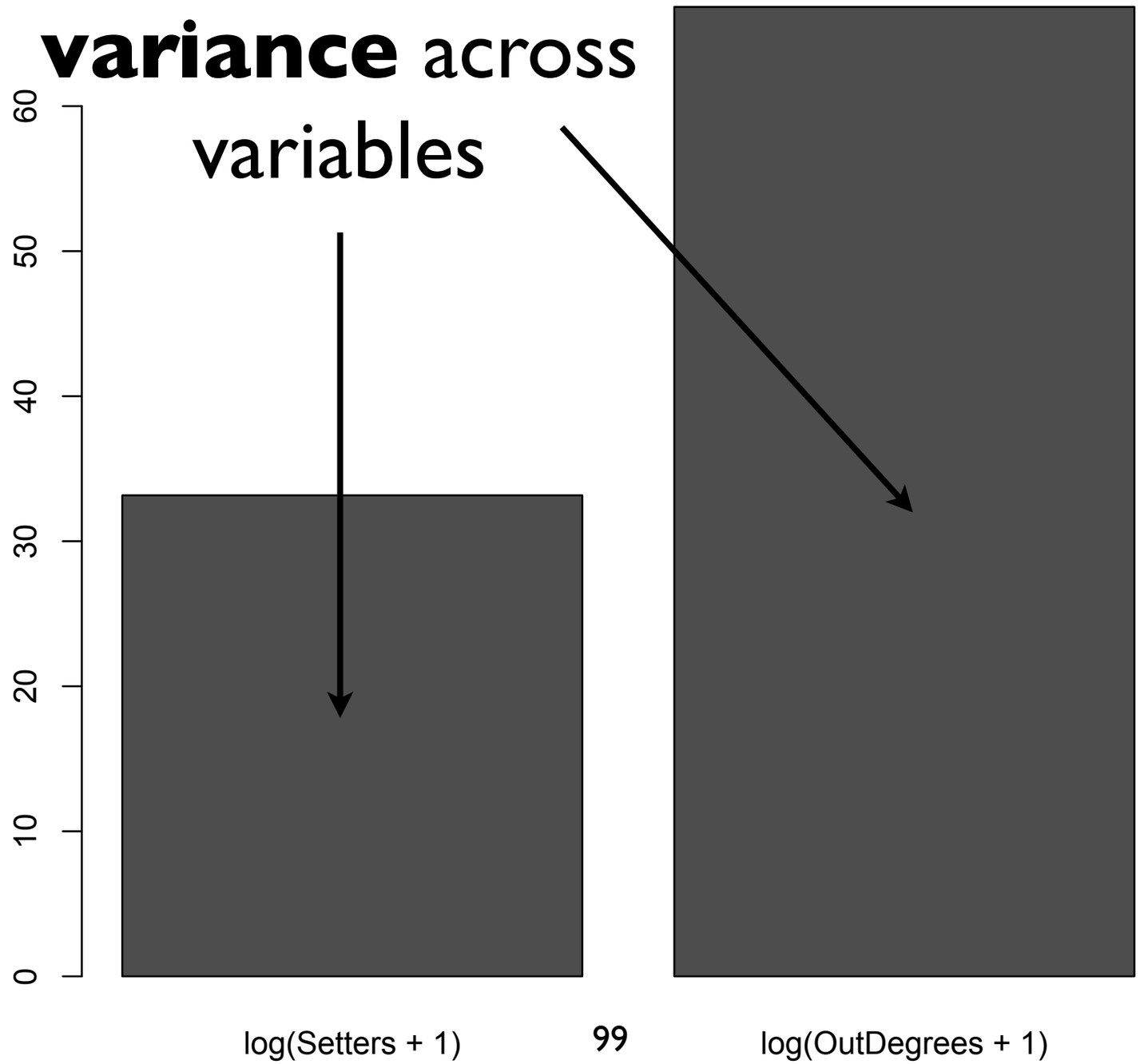
Comparing the Importance of Variables

distribution of **variance** across variables

%variance

relative weights

dominance analysis



Scott Tonidandel and James M. LeBreton. "Determining the Relative Importance of Predictors in Logistic Regression: An Extension of Relative Weight Analysis", Organizational Research Methods, 13:767, 2010. http://orm.sagepub.com/content/13/4/767

Points in Favour/Against

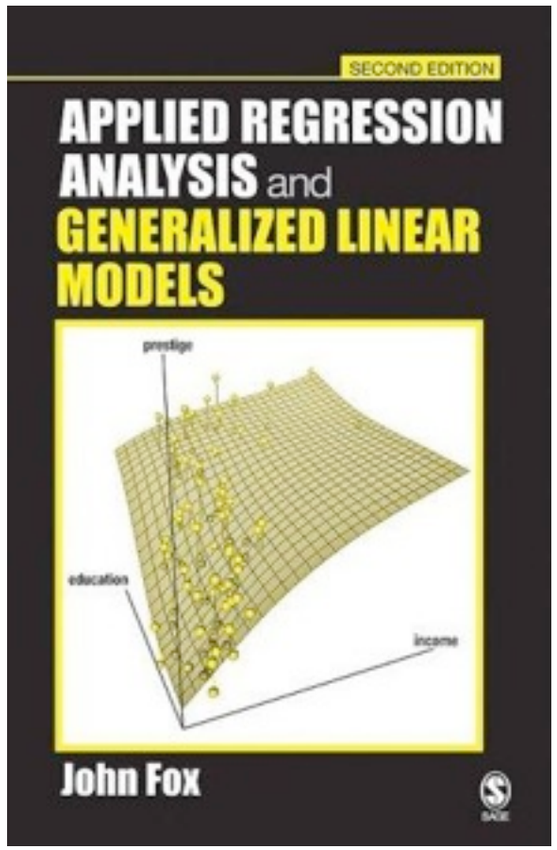
- classifications and probabilities
- large body of work

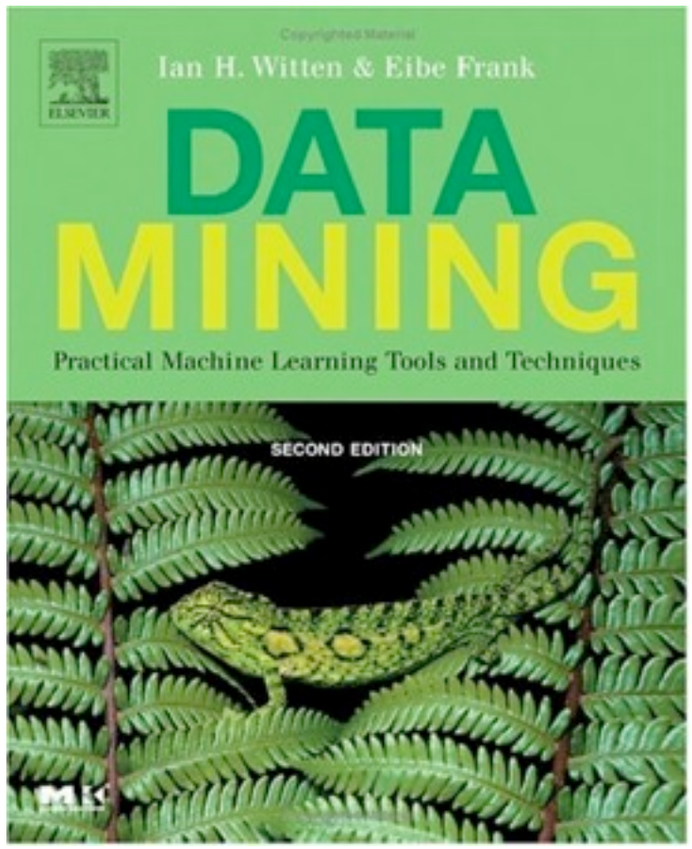
- complex interpretation
- dummy coding

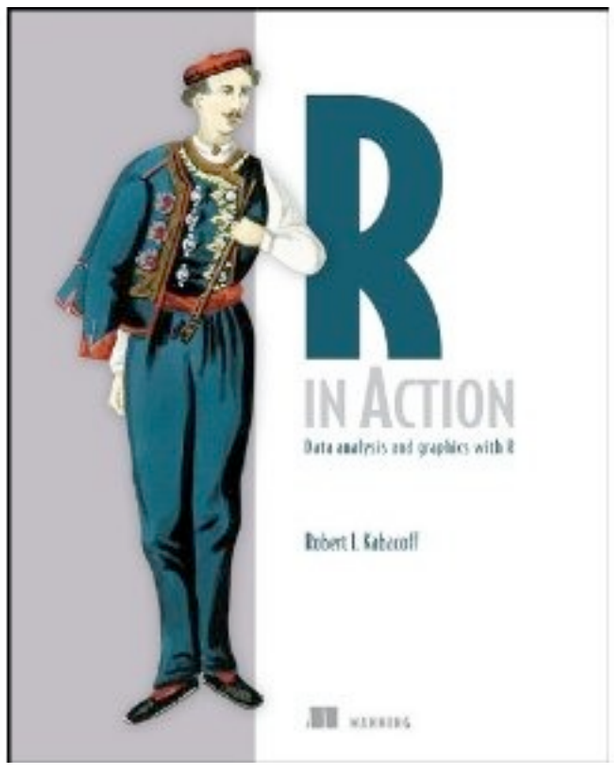
What Now?

- Other classifiers: Naive Bayes, Random Forest and Neural Networks
- Linear Regression
- Clustering: K-means and Agglomer
- Association Rule Mining

References







+ check the proceedings of ICSE, MSR, FSE, ESEM, ASE, ICSM, PROMISE, TSE, ... !