21st of February 2012

Trade-offs in Explanatory Model Learning

Data Analysis Project

Madalina Fiterau

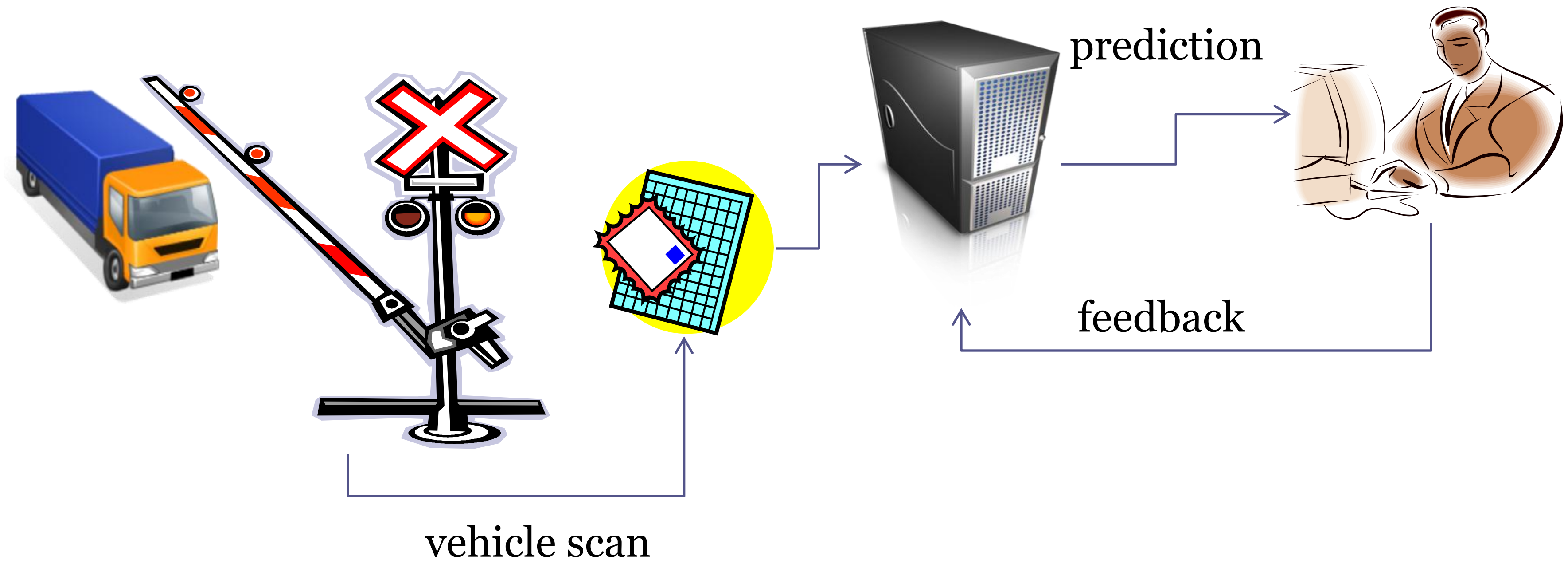
DAP Committee
Artur Dubrawski
Jeff Schneider
Geoff Gordon

Outline

- Motivation: need for interpretable models
- Overview of data analysis tools
- Model evaluation – accuracy vs complexity
- Model evaluation – understandability
- Example applications
- Summary

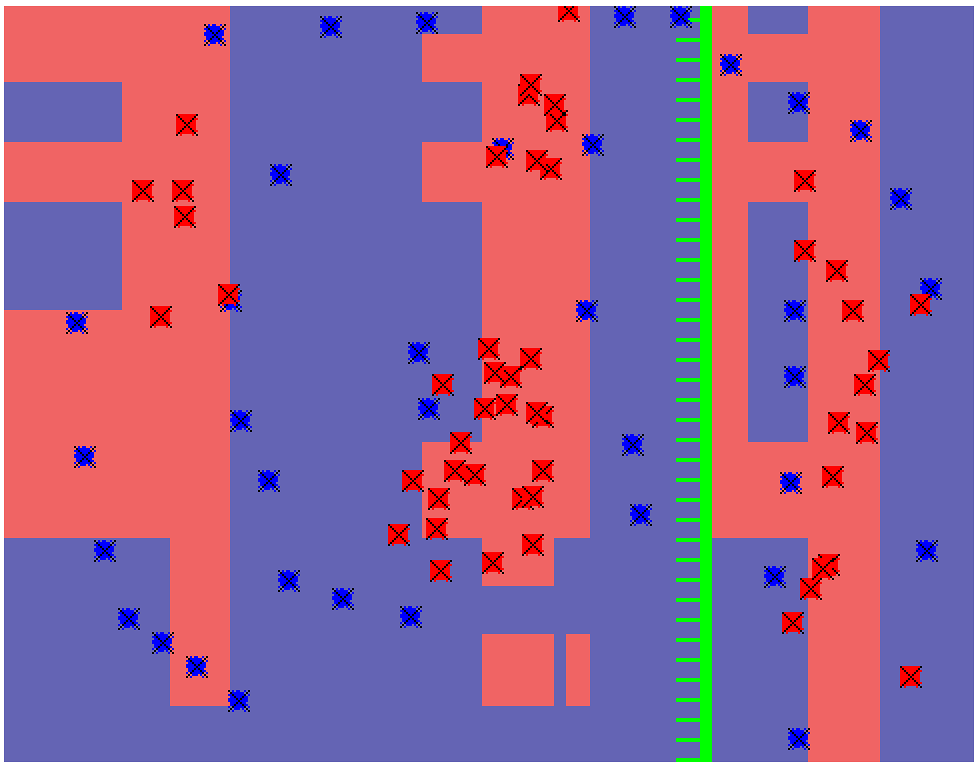
Example Application: Nuclear Threat Detection

- Border control: vehicles are scanned
- Human in the loop interpreting results



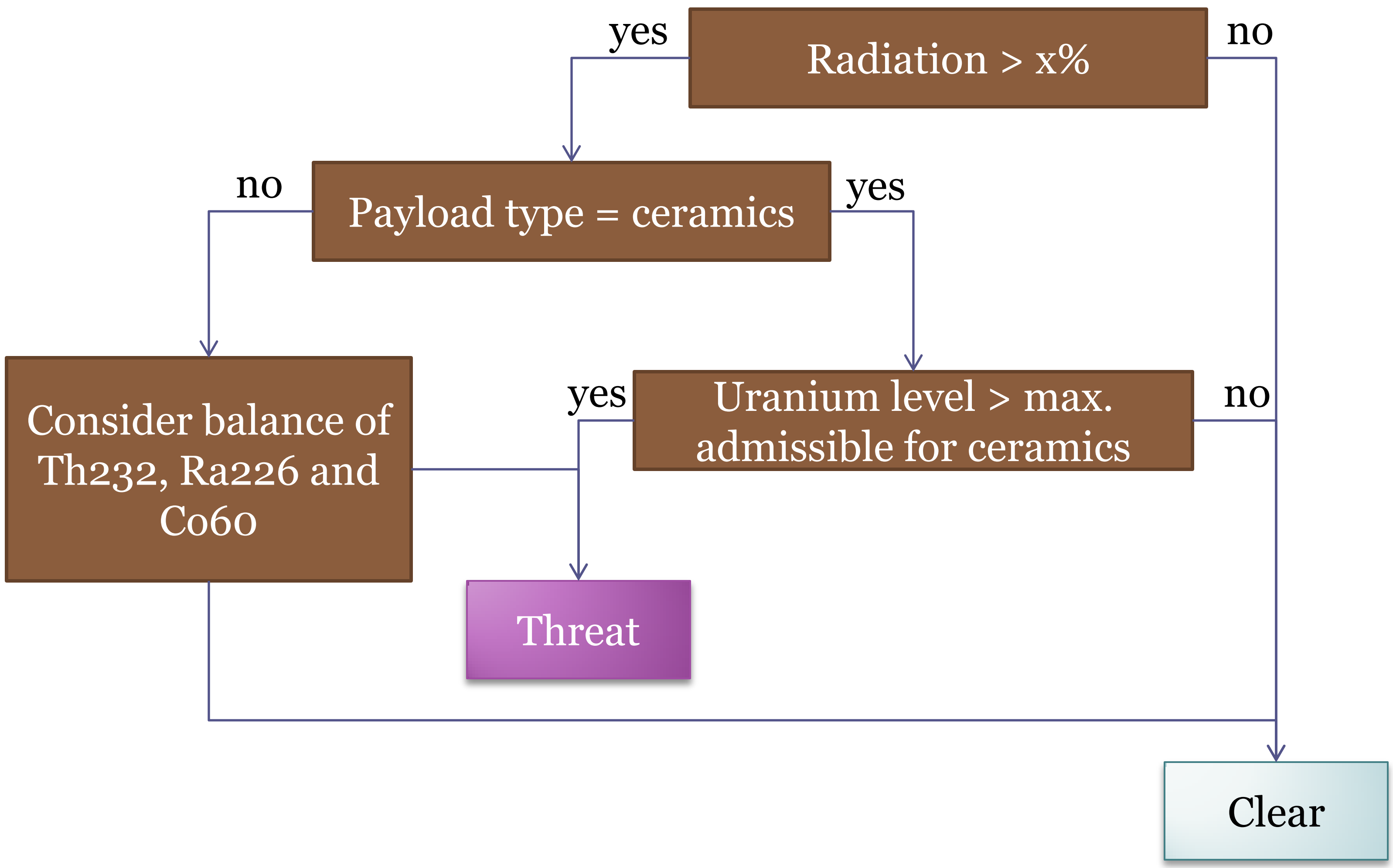
Boosted Decision Stumps

- Accurate, but hard to interpret



How is the prediction derived from the input?

Decision Tree – More Interpretable



Motivation

Many users are willing to trade accuracy to better understand the system-yielded results

Need: simple, interpretable model

Need: explanatory prediction process

Analysis Tools – Black-box

Random Forests

- Very accurate tree ensemble
- L. Breiman, 'Random Forests', 2001

Boosting

- Guarantee: decreases training error
- R. Schapire, '*The boosting approach to machine learning*'

Multi-boosting

- Bagged boosting
- G. Webb, '*MultiBoosting: A Technique for Combining Boosting and Weighted Bagging*'

Analysis Tools – White-box

CART

- Decision tree based on the Gini Impurity criterion

Feating

- Dec. tree with leaf classifiers
- K. Ting, G. Webb, '*FaSS: Ensembles for Stable Learners*'

Subspacing

- Ensemble: each discriminator trained on a random subset of features
- R. Bryll, '*Attribute bagging* '

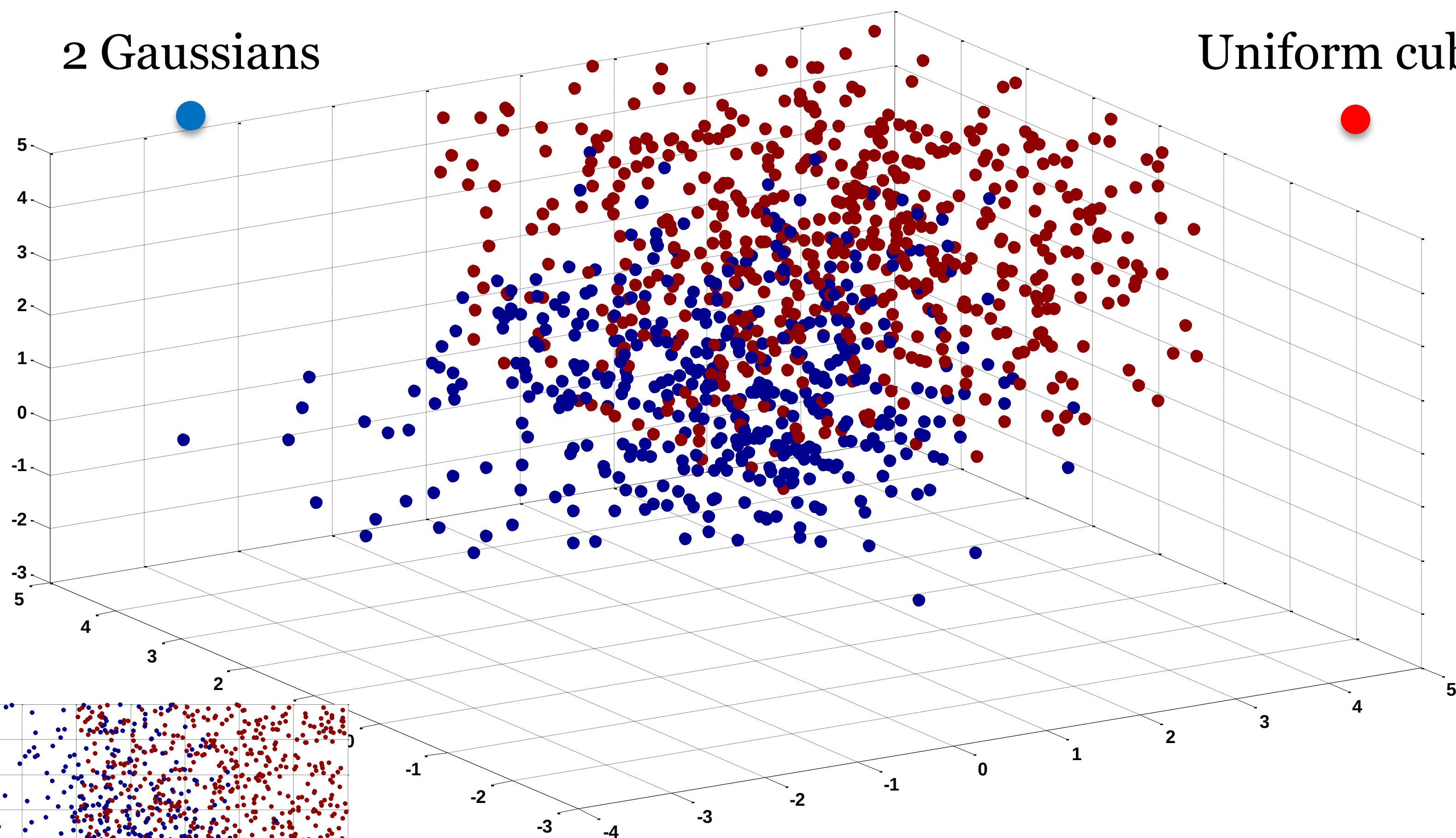
EOP

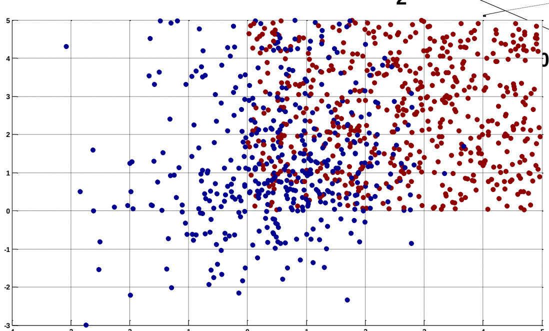
- Builds a decision list that selects the classifier to deal with a query point

Explanation-Oriented Partitioning

2 Gaussians

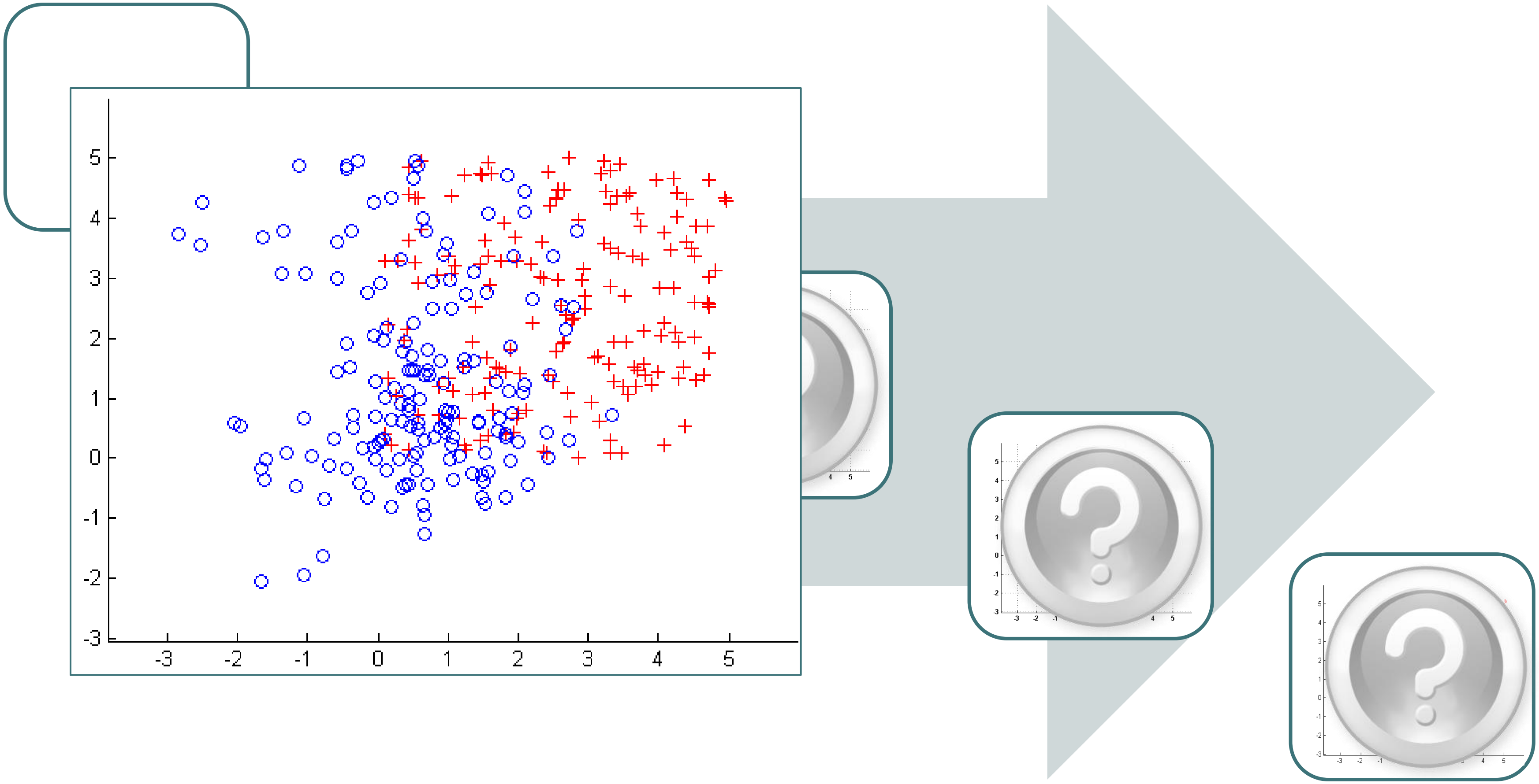
Uniform cube





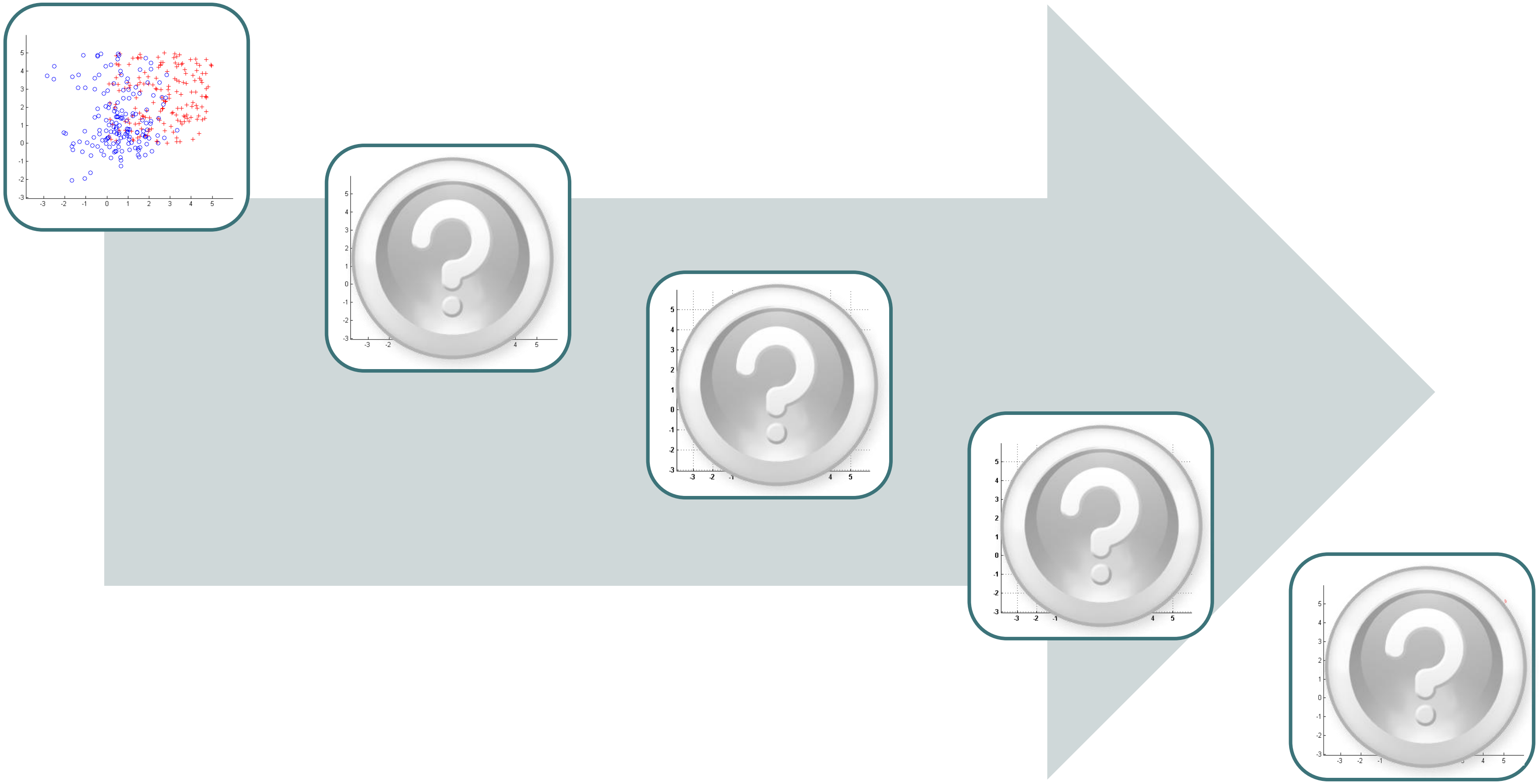
(X,Y) plot

EOP Execution Example – 3D data



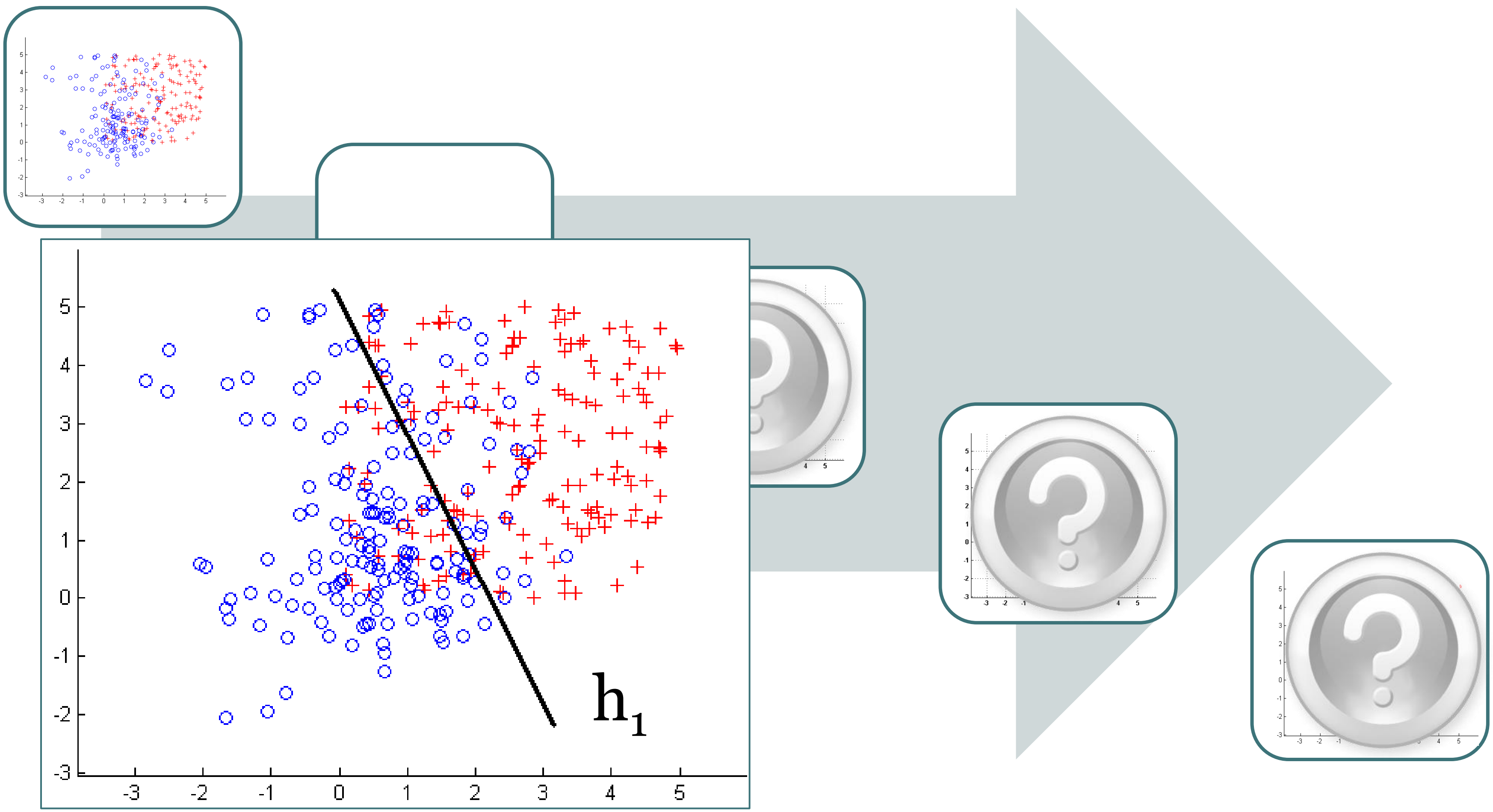
Step 1: Select a projection - (X_1,X_2)

EOP Execution Example – 3D data



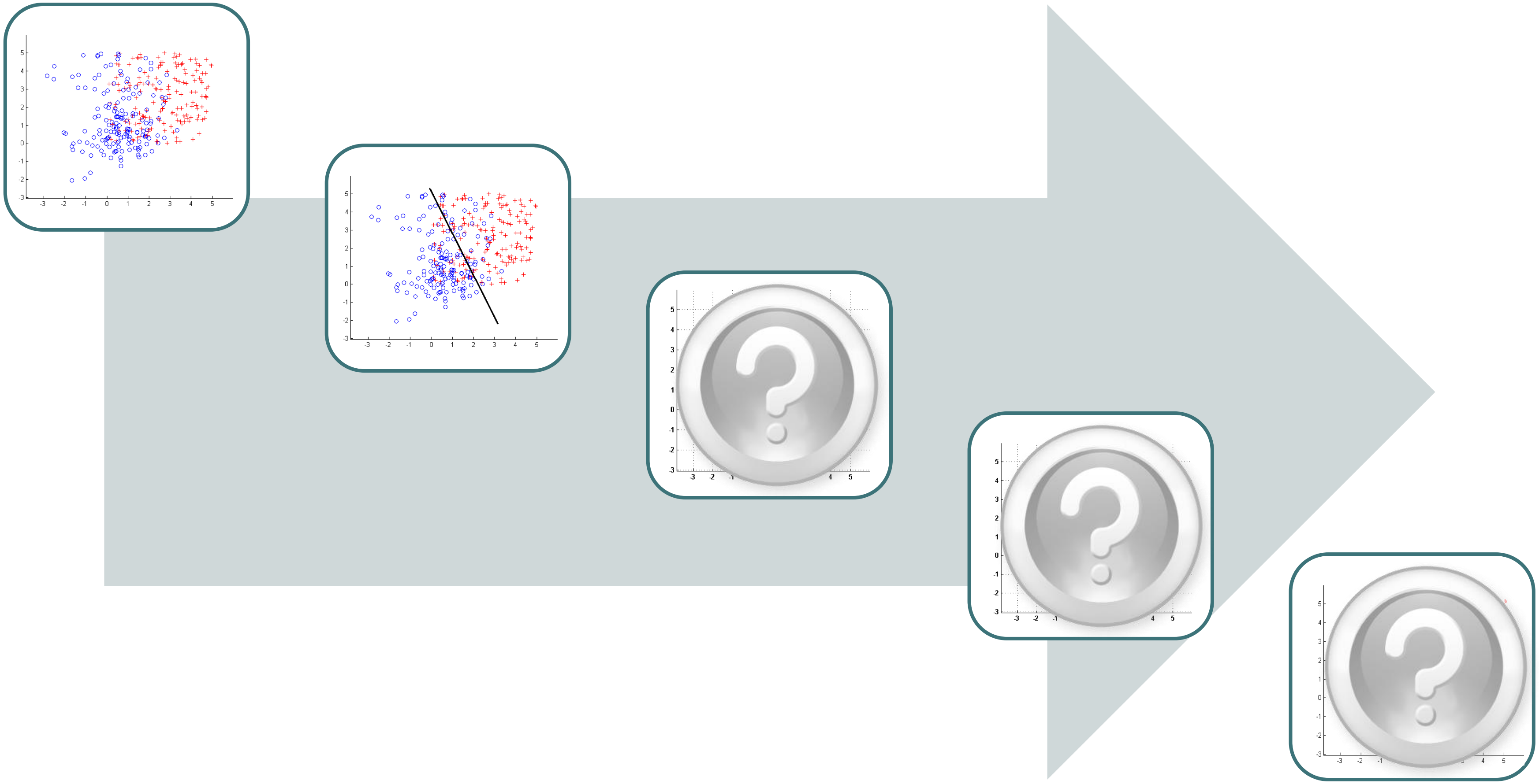
Step 1: Select a projection - (X_1,X_2)

EOP Execution Example – 3D data



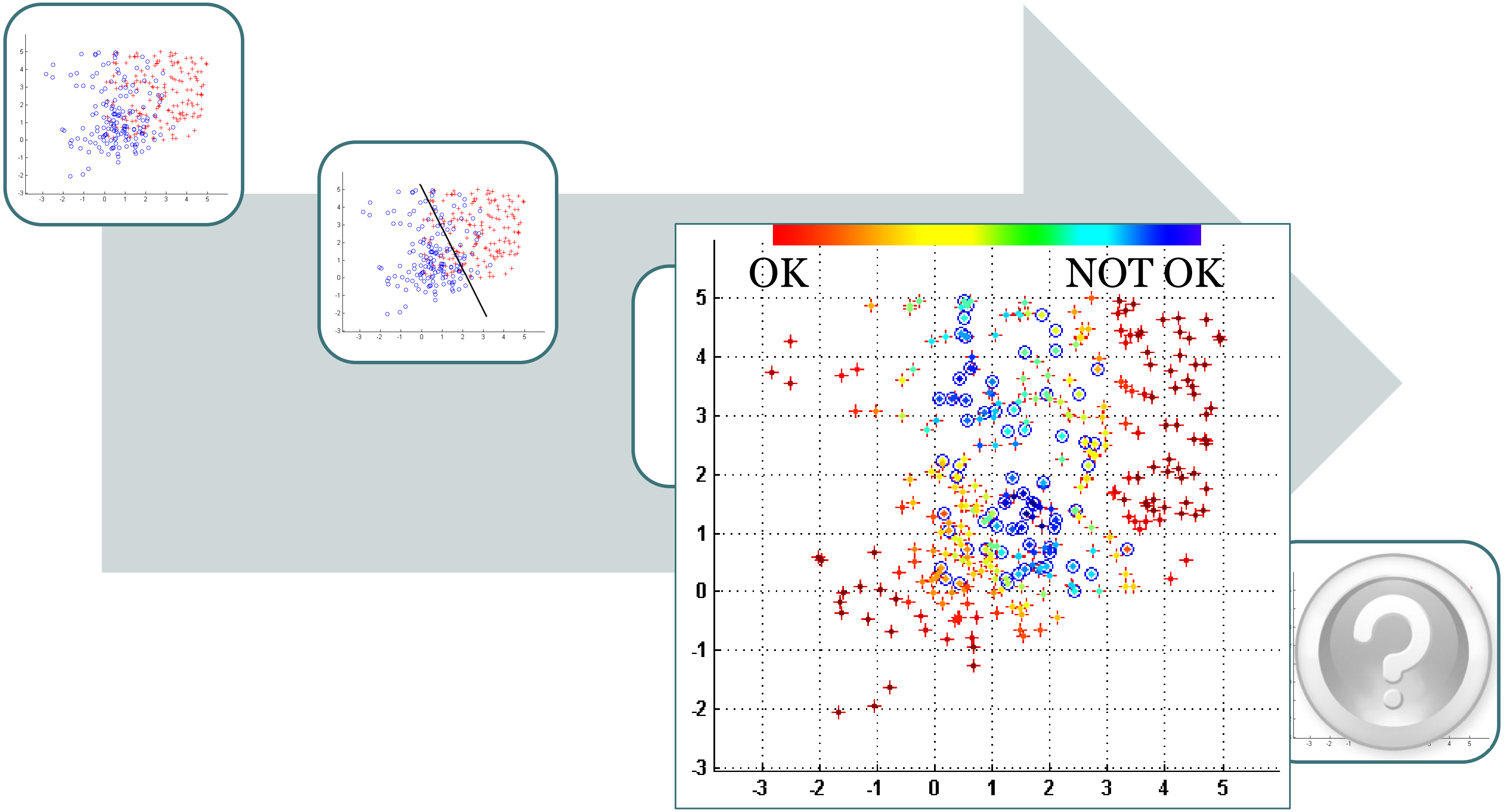
Step 2: Choose a good classifier - call it h_1

EOP Execution Example – 3D data



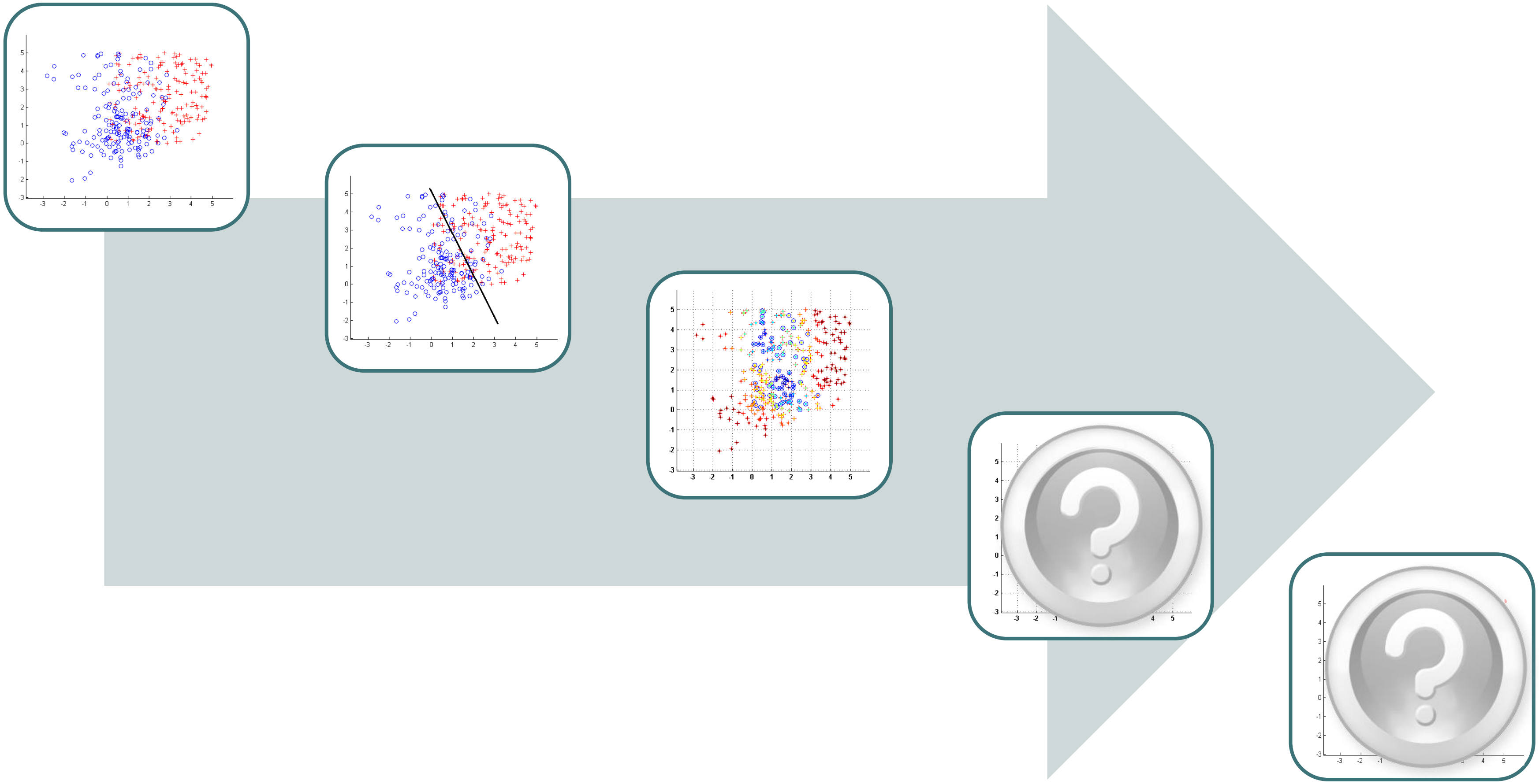
Step 2: Choose a good classifier - call it h_1

EOP Execution Example – 3D data



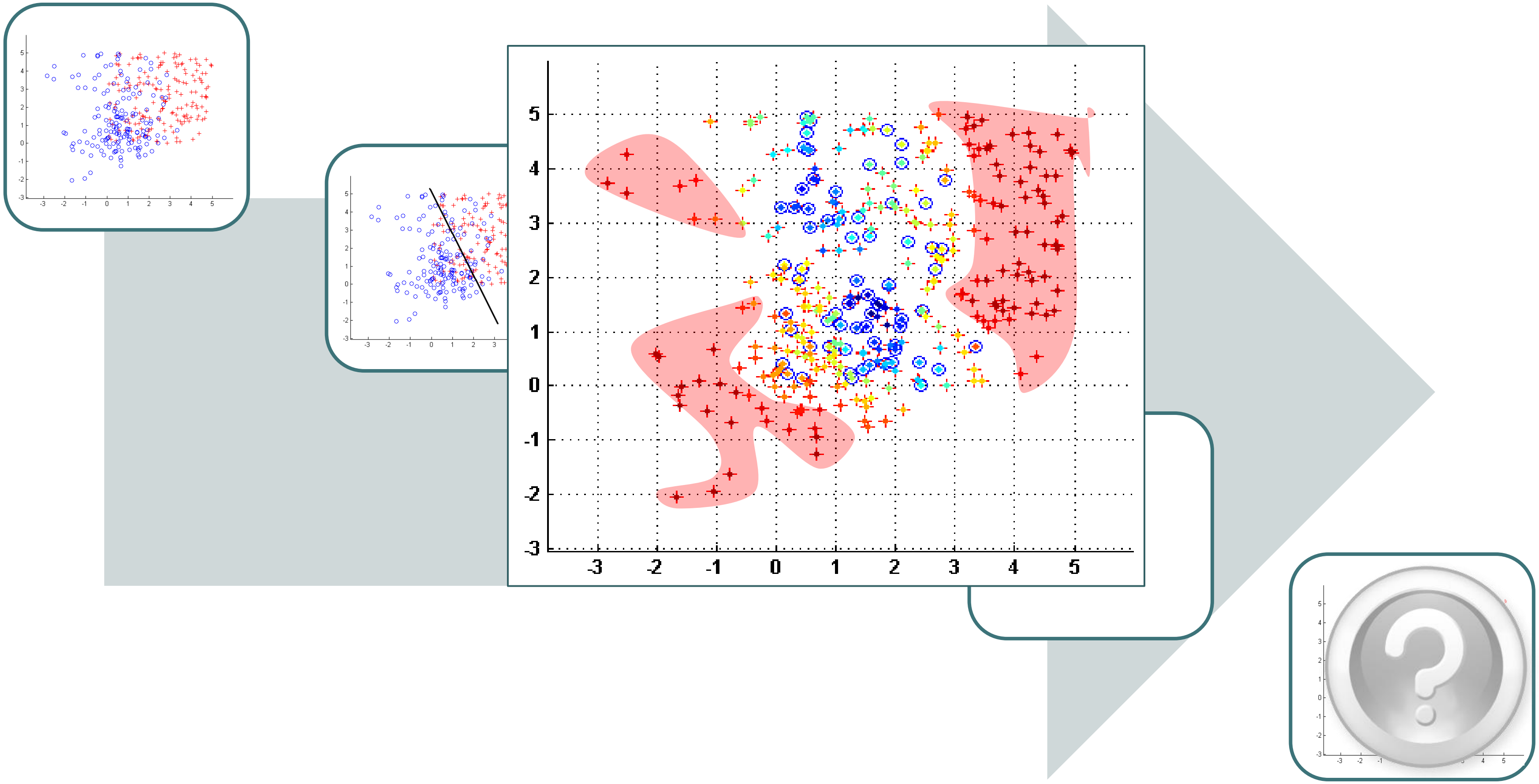
Step 3: Estimate accuracy of h_1 at each point

EOP Execution Example – 3D data



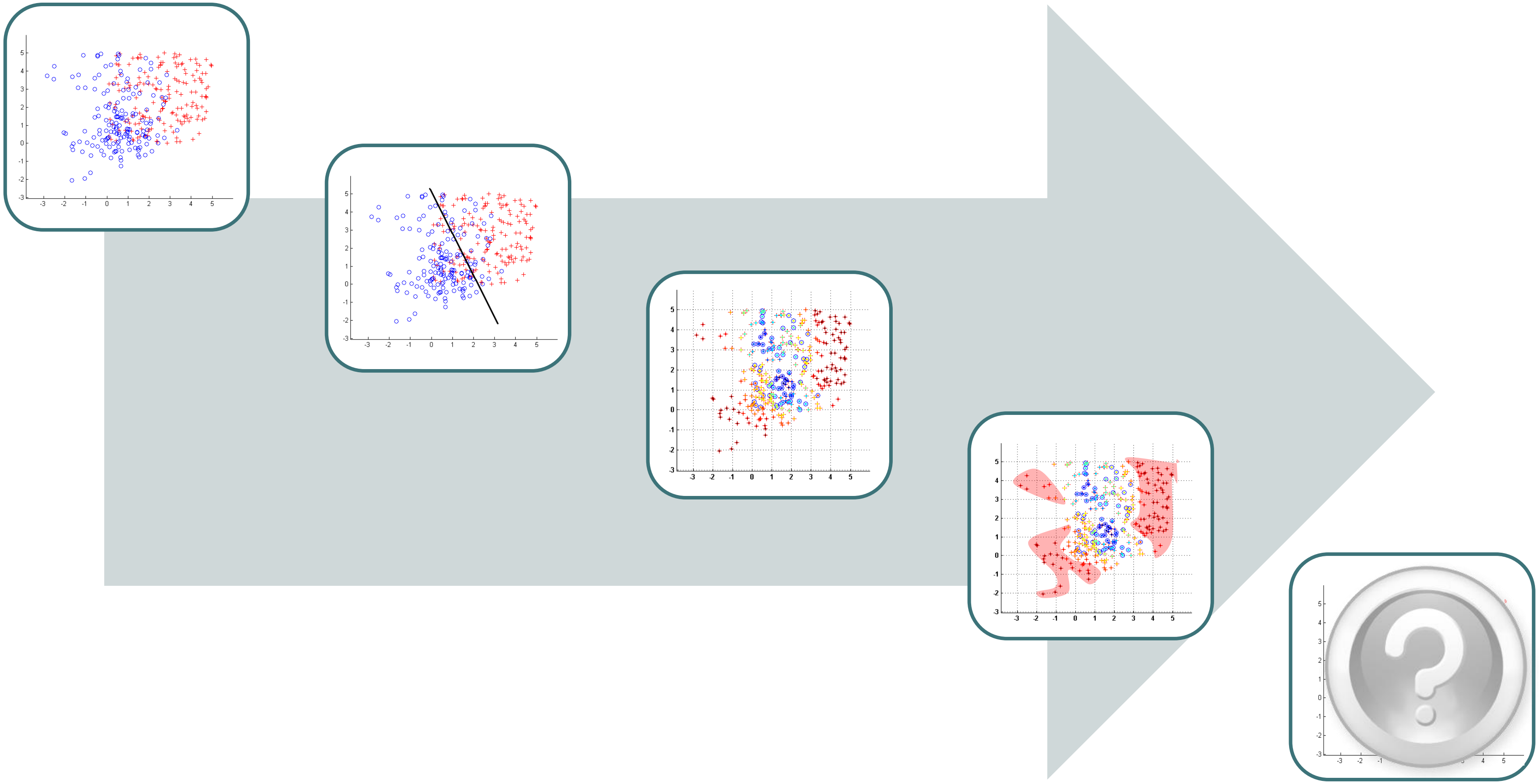
Step 3: Estimate accuracy of h_1 for each point

EOP Execution Example – 3D data



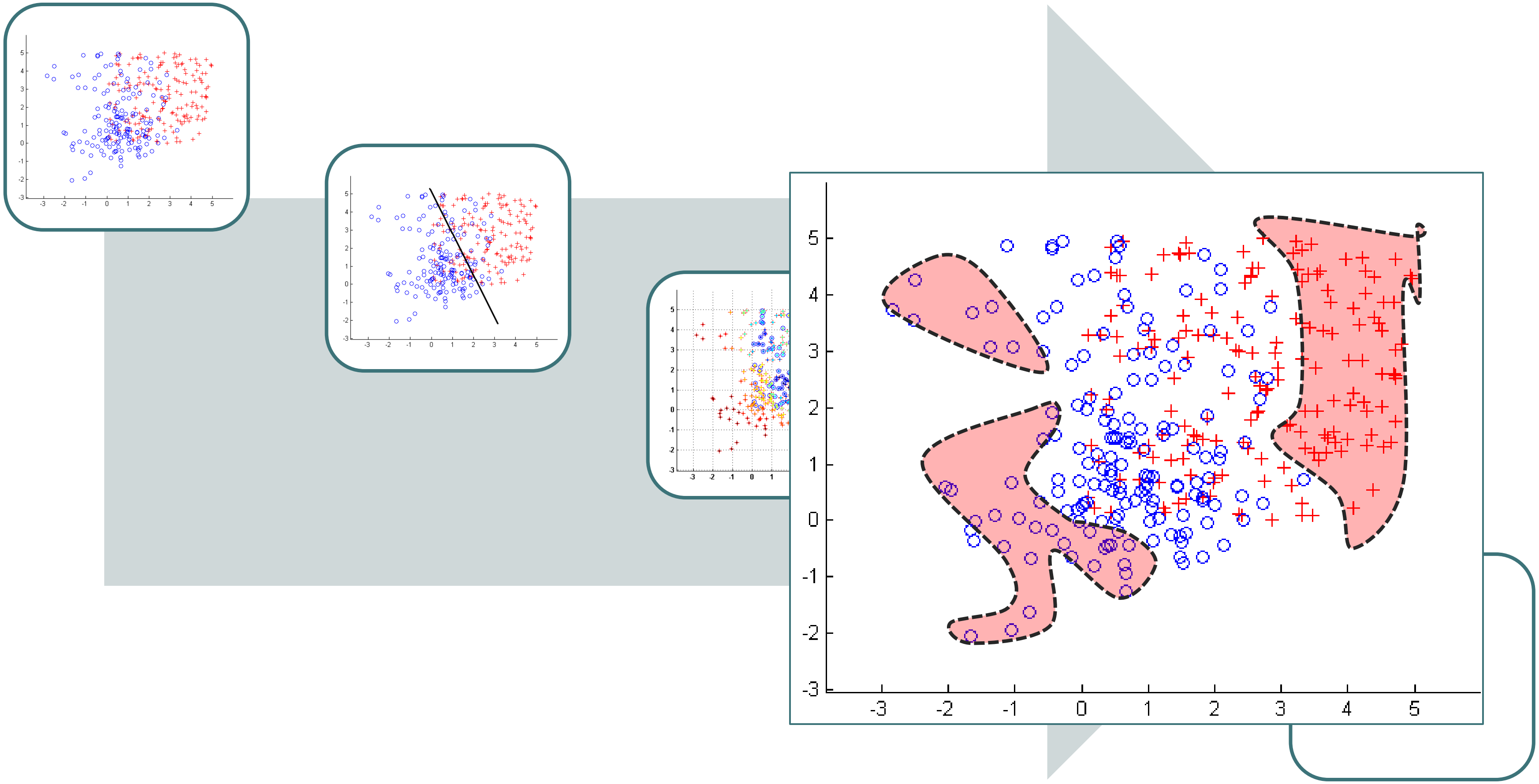
Step 4: Identify high accuracy regions

EOP Execution Example – 3D data



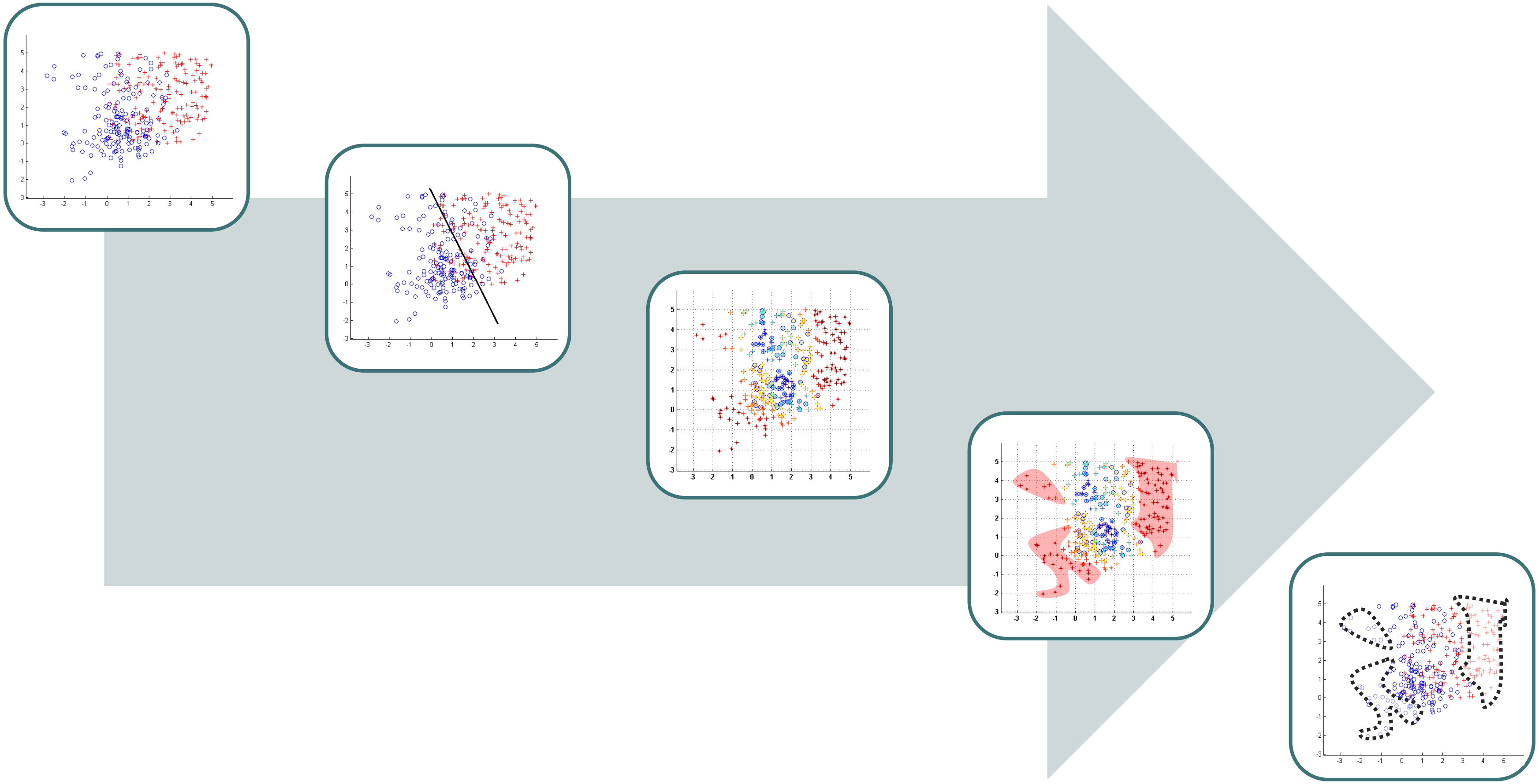
Step 4: Identify high accuracy regions

EOP Execution Example – 3D data



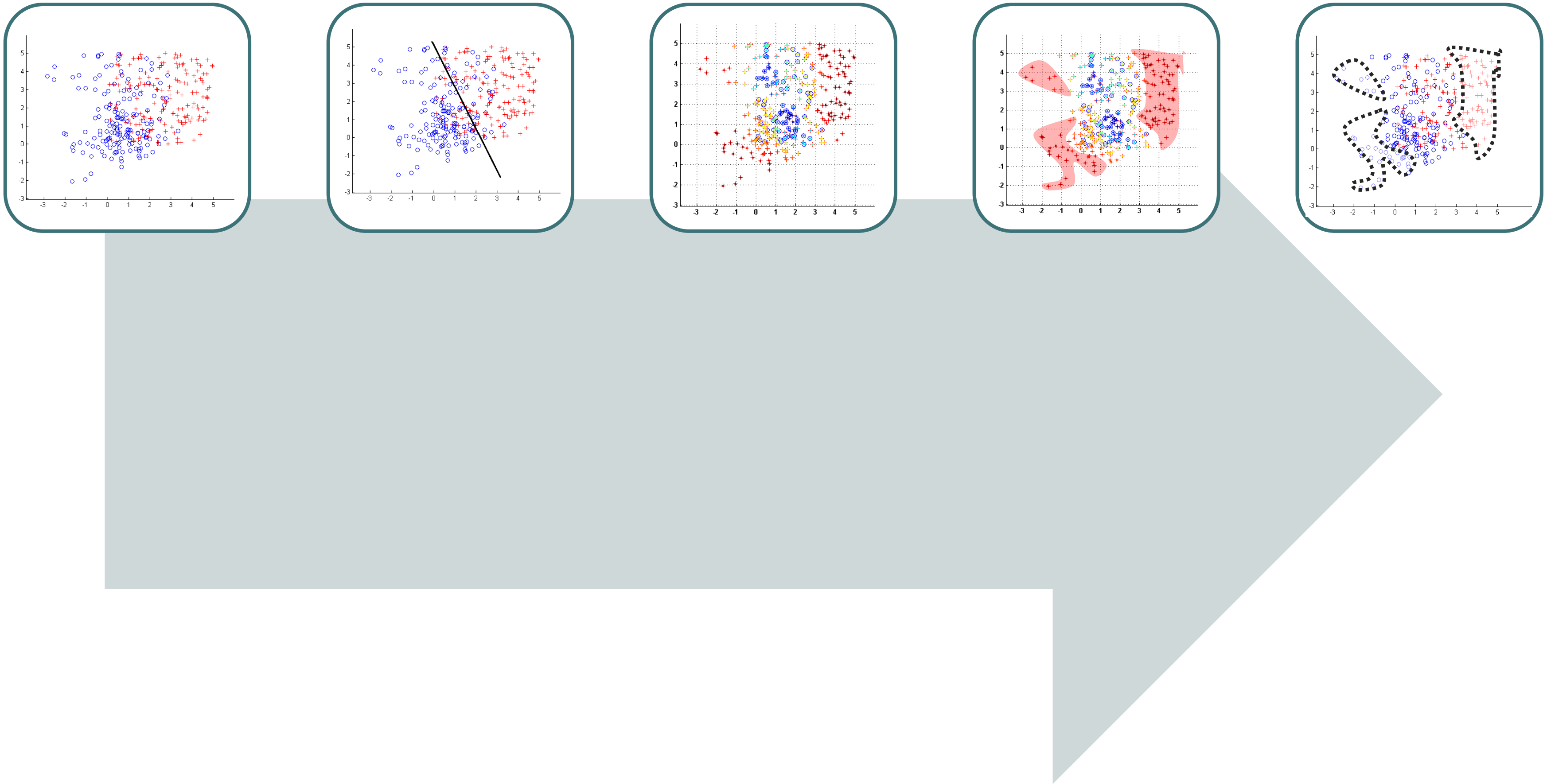
Step 5:Training points - removed from consideration

EOP Execution Example – 3D data



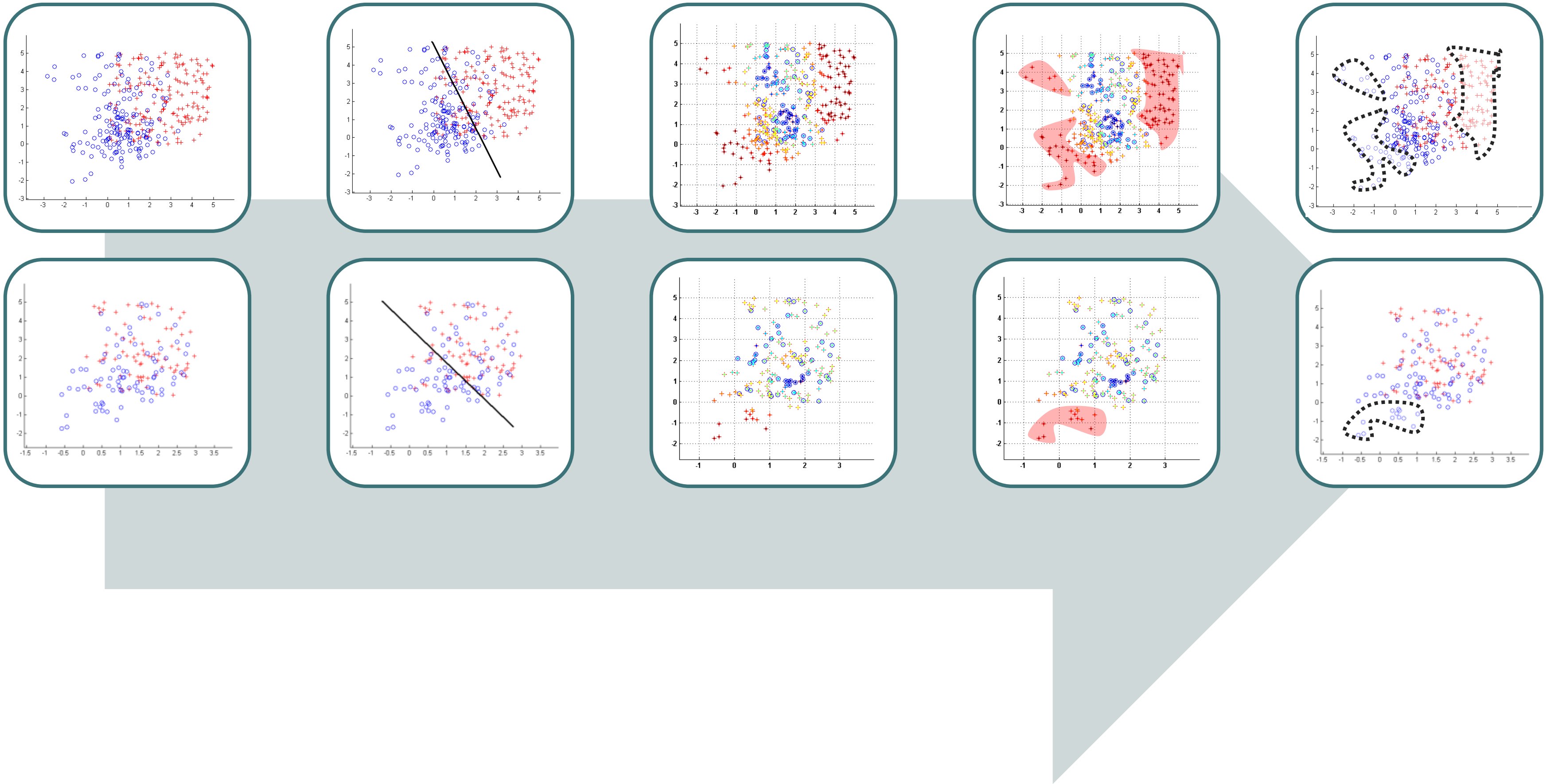
Step 5:Training points - removed from consideration

EOP Execution Example – 3D data



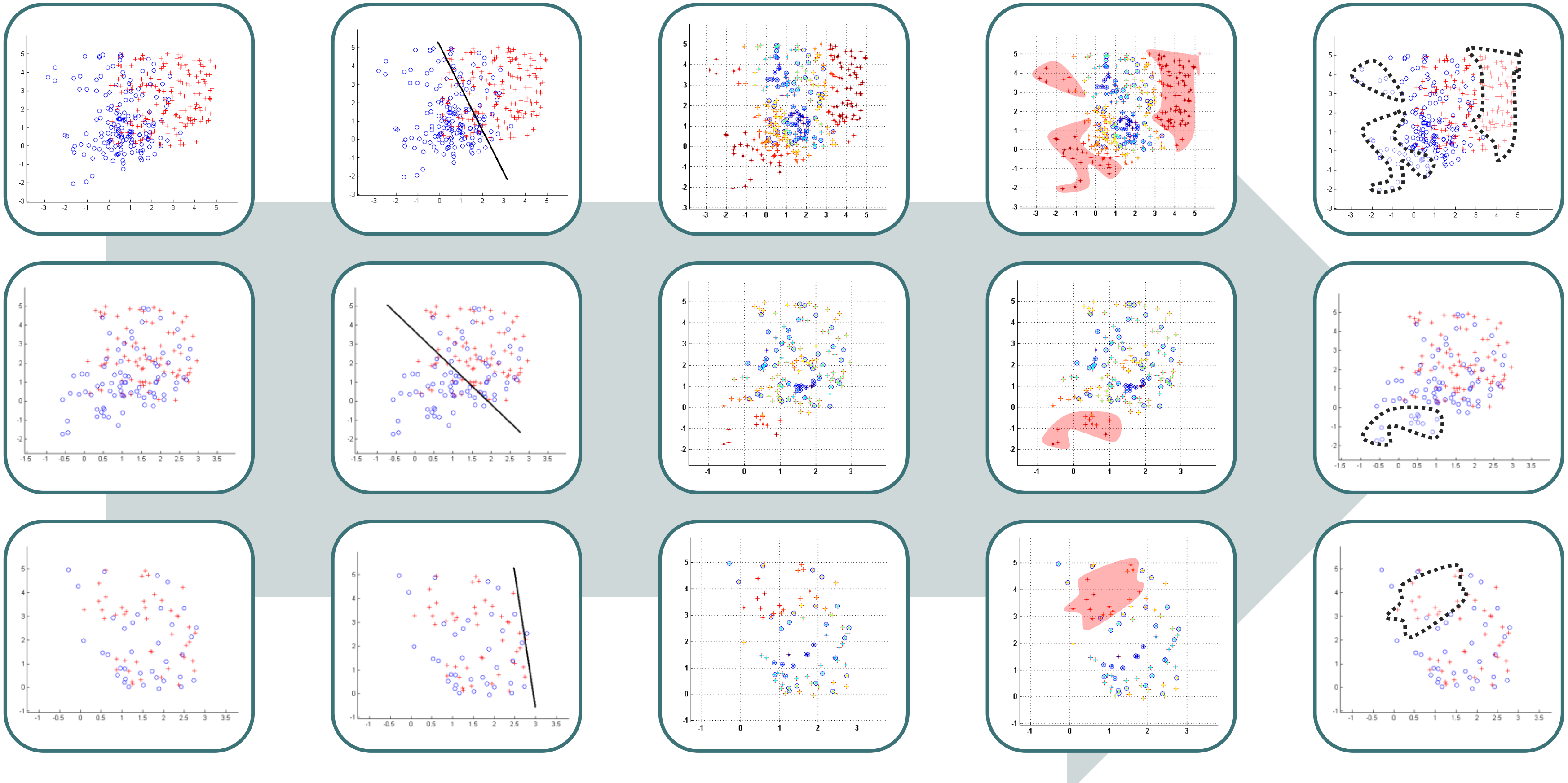
Finished first iteration

EOP Execution Example – 3D data



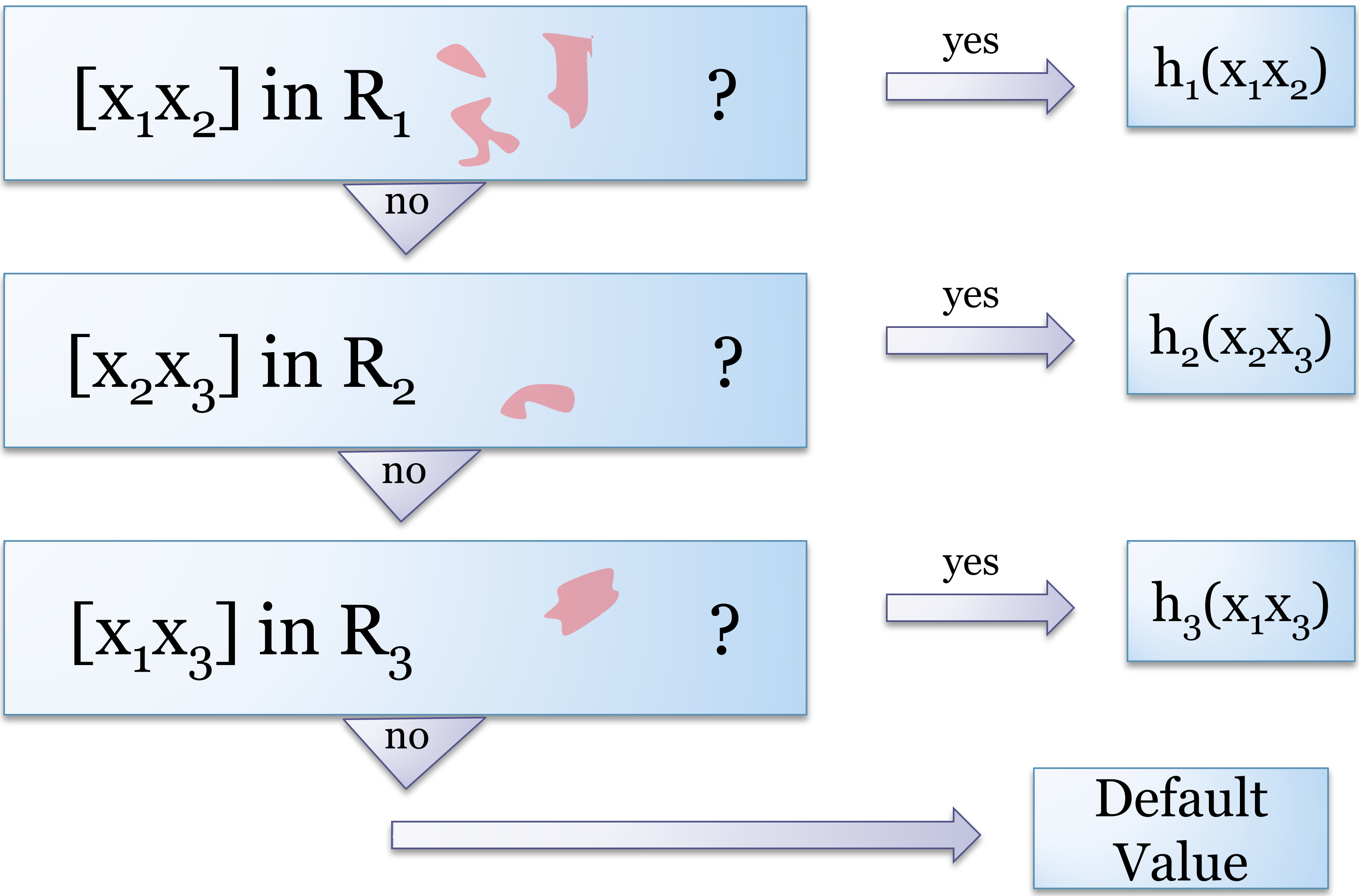
Finished second iteration

EOP Execution Example – 3D data



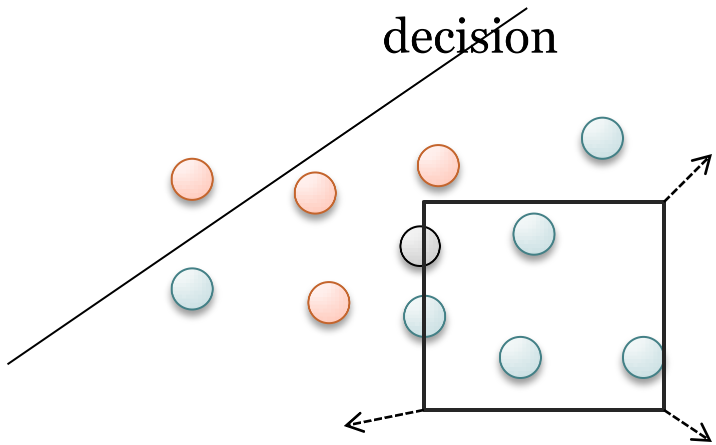
Iterate until all data is accounted for
or error cannot be decreased

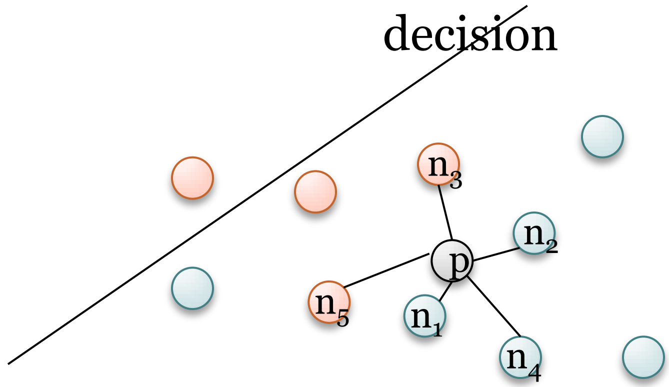
Learned Model – Processing query $[x_1x_2x_3]$



Parametric / Nonparametric Regions

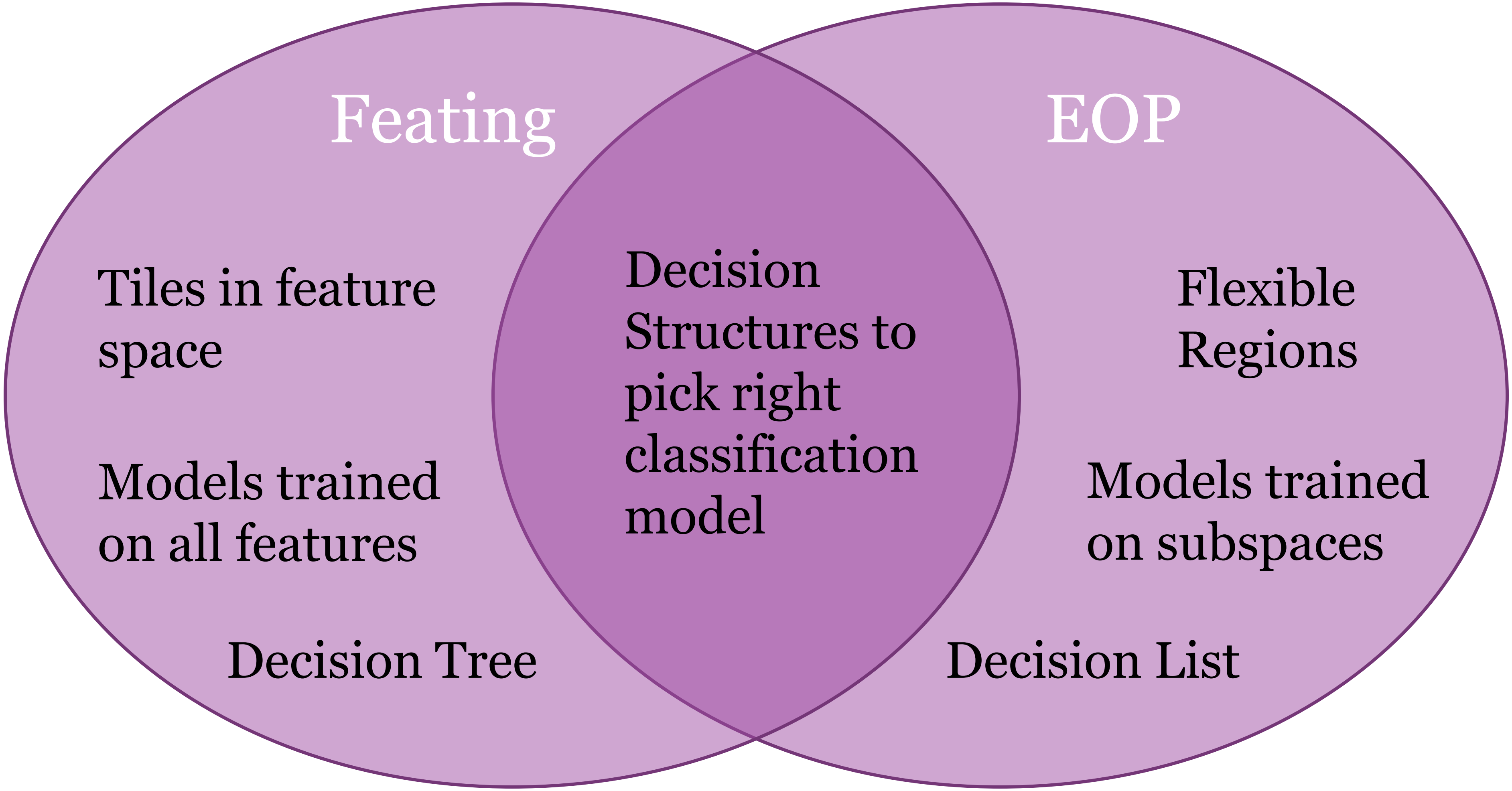
Bounding Polyhedra		Nearest-neighbor Score	
Enclose points in convex shapes (hyper-rectangles /spheres).		Consider the k-nearest neighbors Region: { X | Score(X) > t} t – learned threshold	
	Easy to test inclusion		Easy to test inclusion
	Visually appealing		Can look insular
	Inflexible		Deals with irregularities





Incorrectly classified   Correctly classified   Query point

Feating and EOP



Outline

- Motivation: need for interpretable models
- Overview of data analysis tools
- Model evaluation – accuracy vs complexity
- Model evaluation – understandability
- Example applications
- Summary

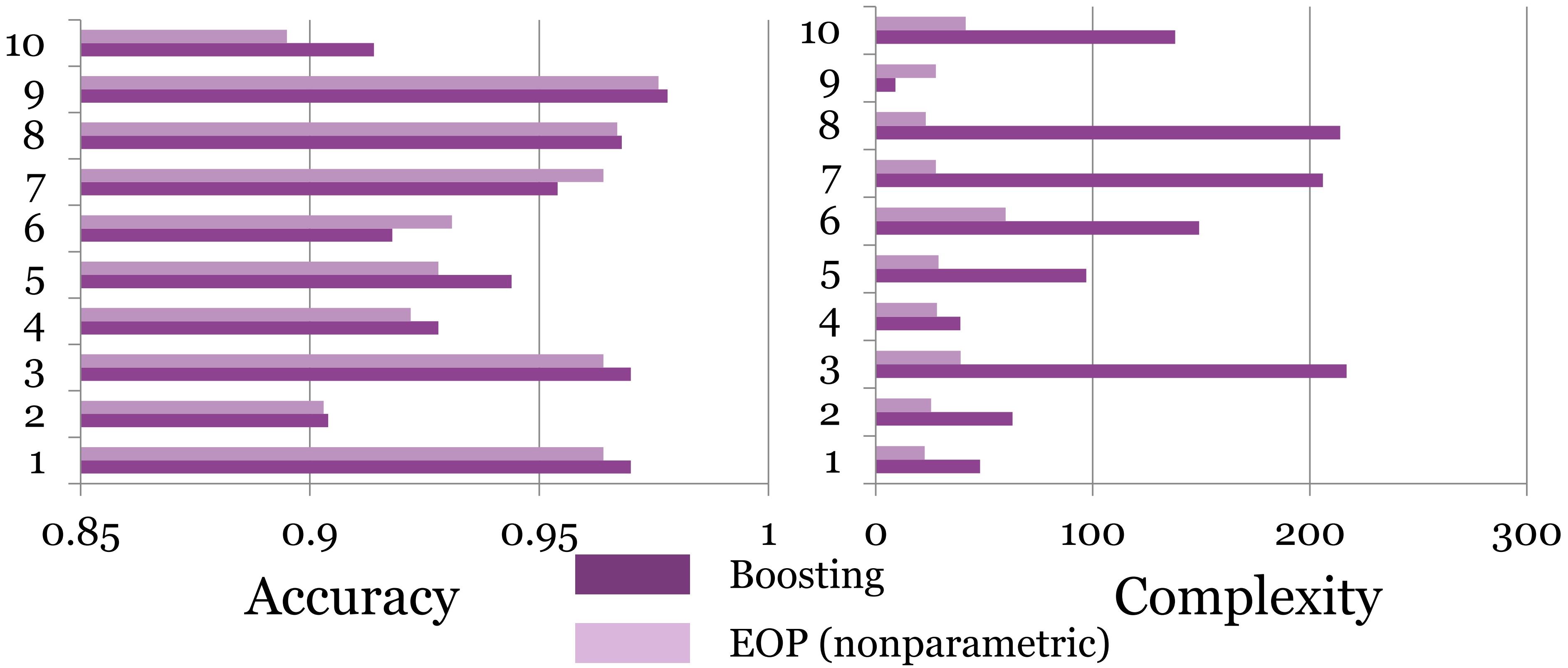
Overview of datasets

- Real valued features, binary output
- Artificial data – 10 features
  - Low-d Gaussians/uniform cubes
- UCI repository
- Application-related datasets

- Results by k-fold cross validation
  - Complexity = expected number of vector operations performed for a classification task

EOP vs AdaBoost - SVM base classifiers

- EOP is often less accurate, but not significantly
- the reduction of complexity is statistically significant



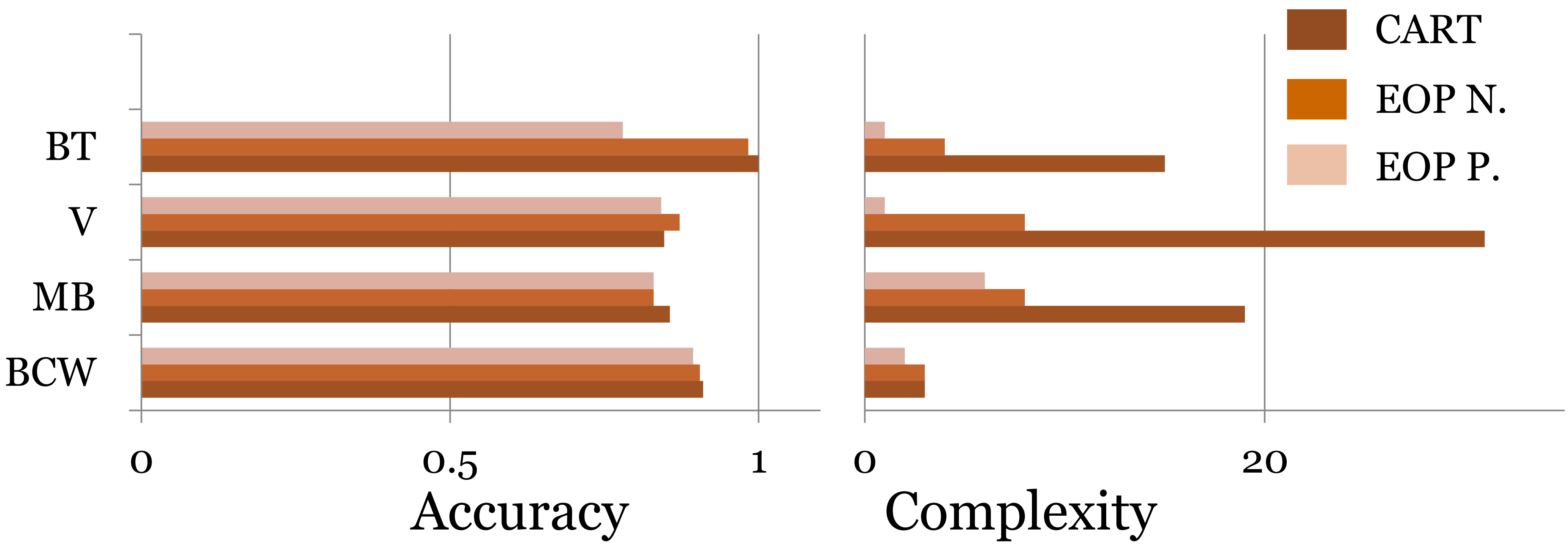
mean diff in accuracy: 0.5%

p-value of 2-sided test: 0.832

mean diff in complexity: 85

p-value of 2-sided test: 0.003

EOP (stumps as base classifiers) vs CART on data from the UCI repository



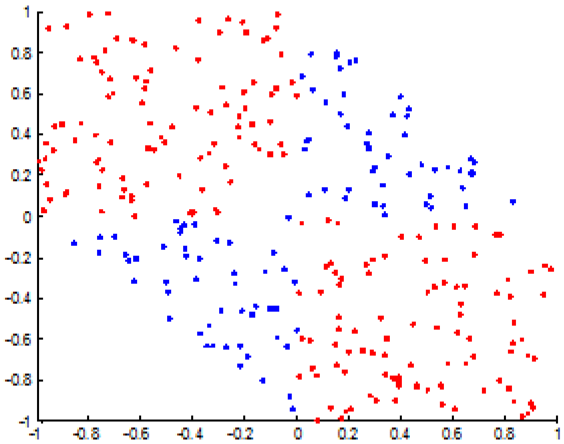
- CART is the most accurate

Dataset	***# of Features***	***# of Points***
Breast Tissue	10	1006
Vowel	9	990
MiniBOONE	10	5000
Breast Cancer	10	596

- Parametric EOP yields the simplest models

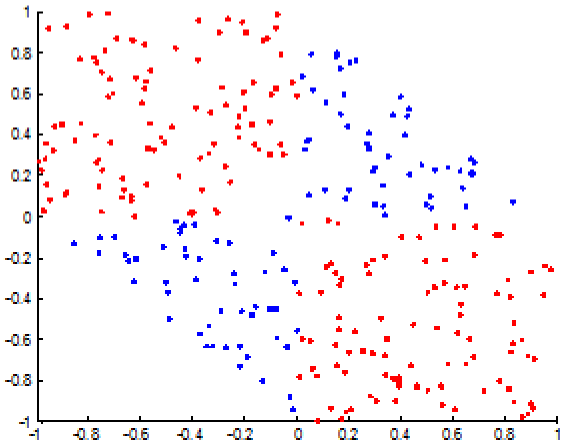
Why are EOP models less complex?

Typical XOR dataset

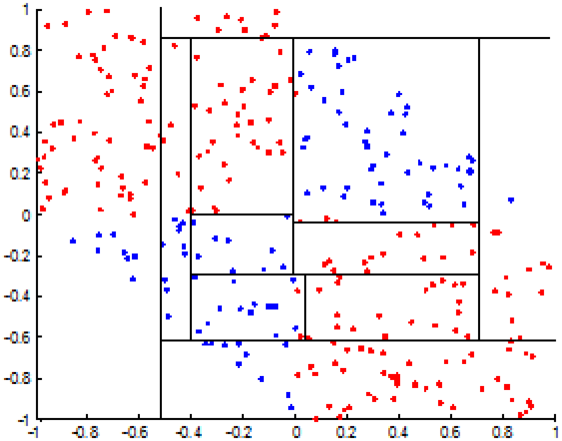


Why are EOP models less complex?

Typical XOR dataset

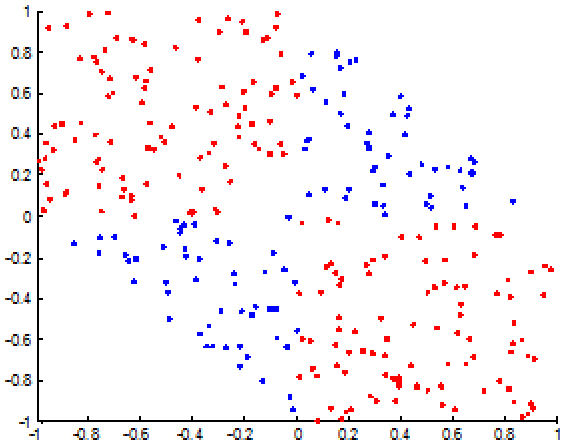


CART
• is accurate
• takes many iterations
• does not uncover or leverage structure of data

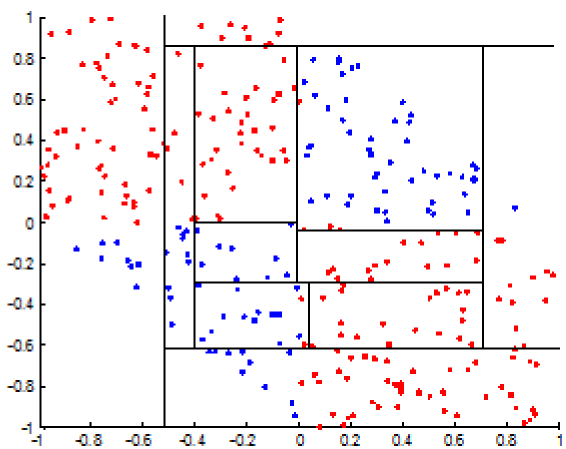


Why are EOP models less complex?

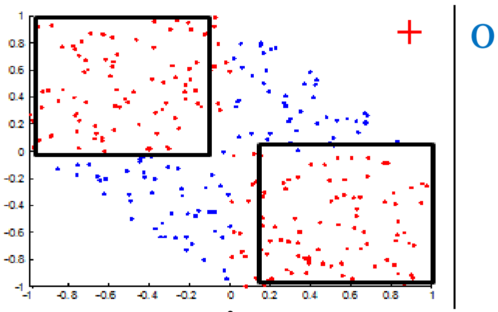
Typical XOR dataset



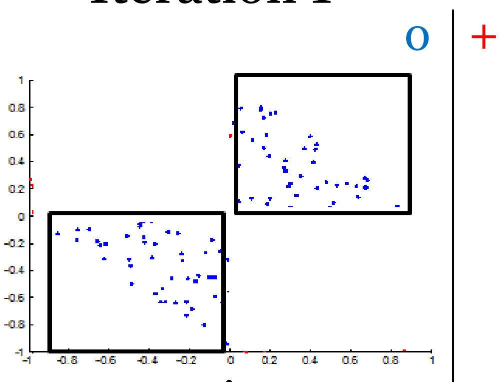
CART
- is accurate
- takes many iterations
- does not uncover or leverage structure of data



EOP
- equally accurate
- uncovers structure

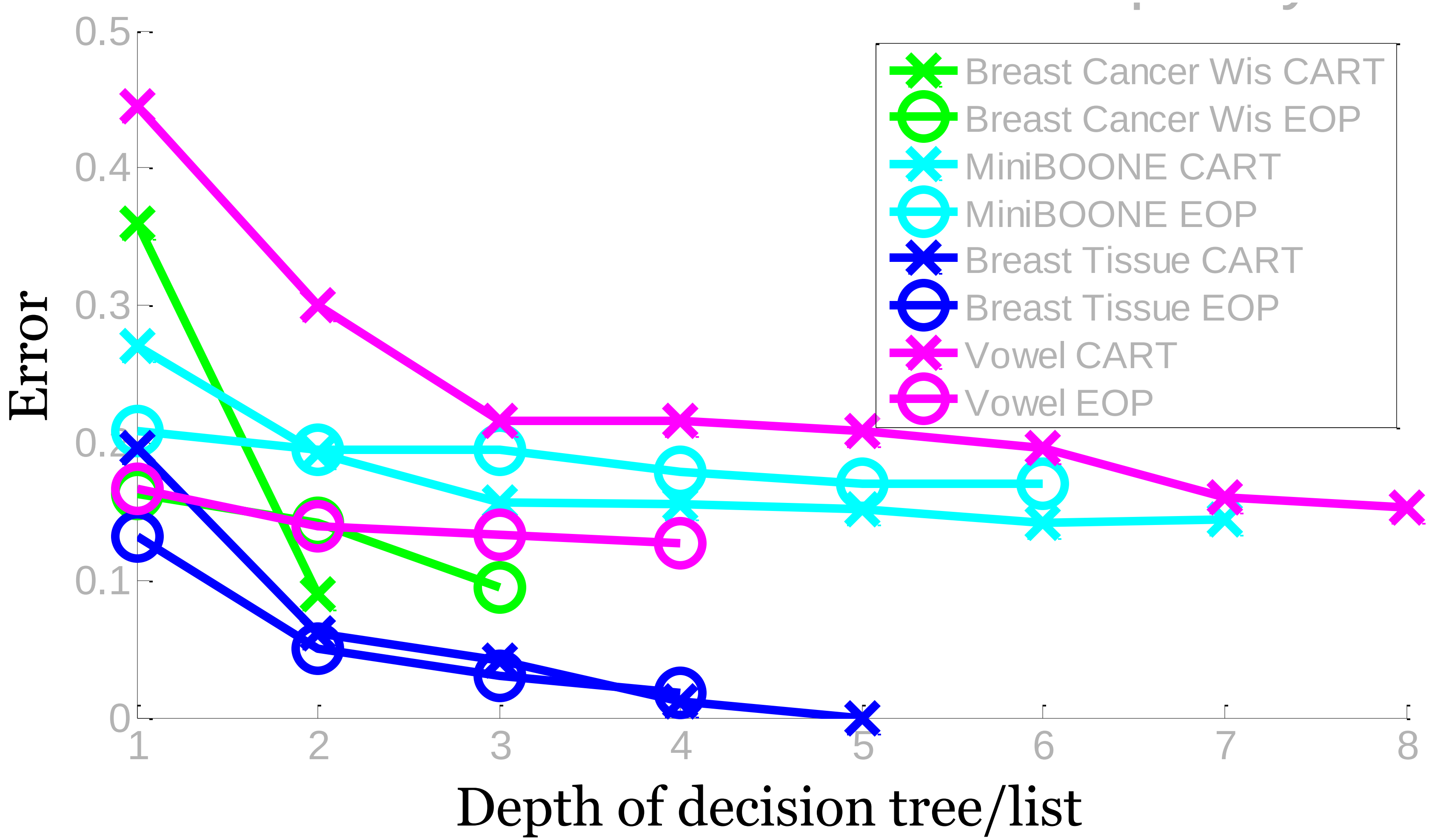


Iteration 1



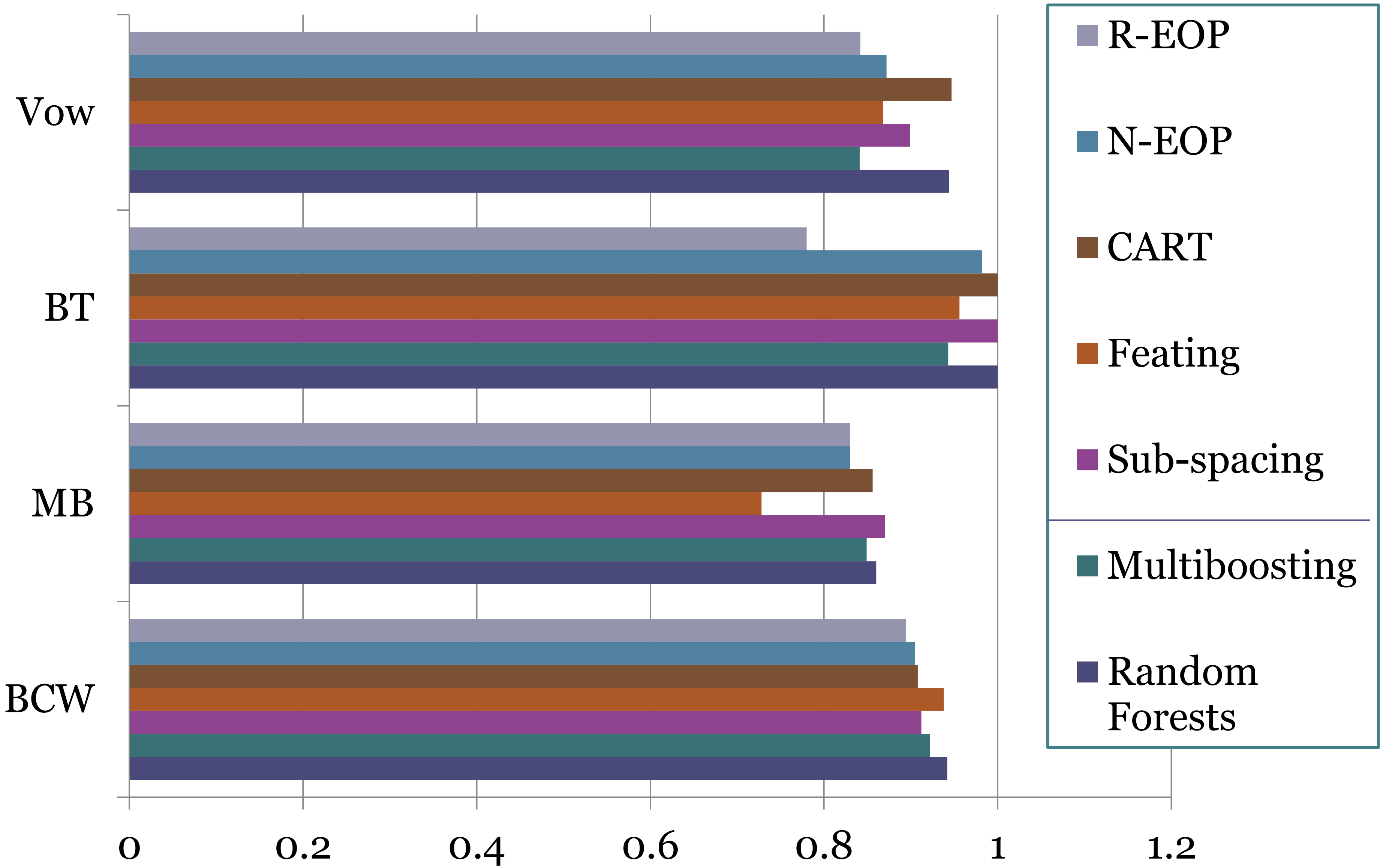
Iteration 2

Error Variation With Model Complexity for EOP and CART

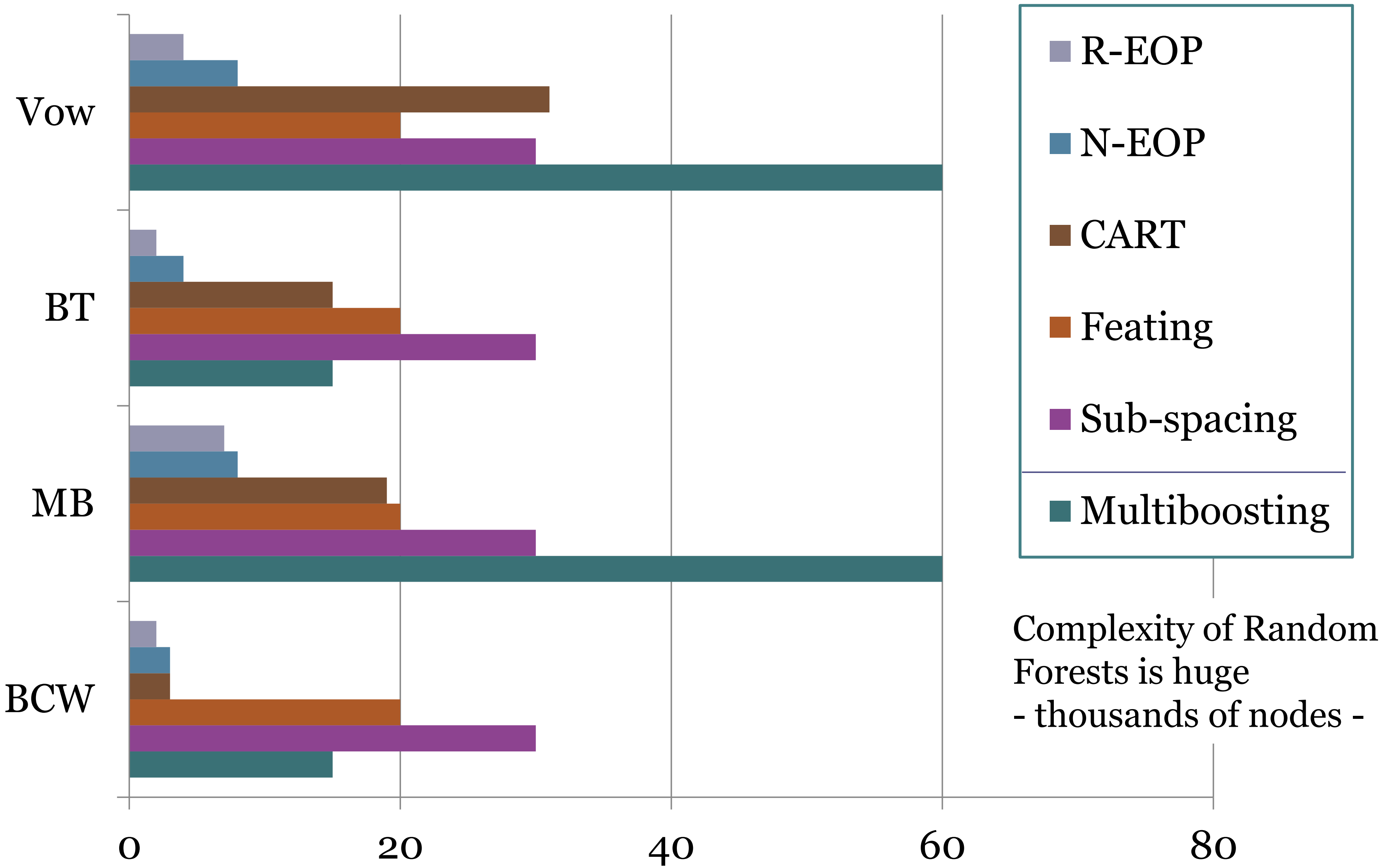


- At low complexities, EOP is typically more accurate

UCI data – Accuracy



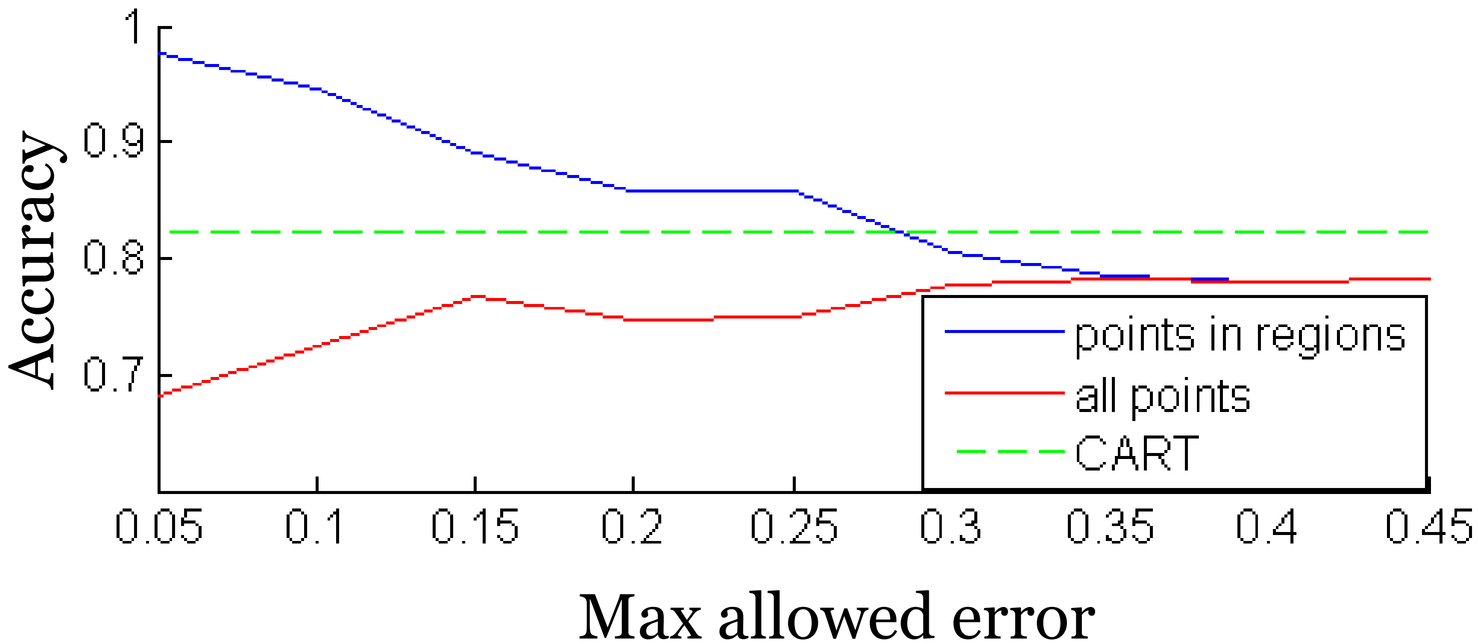
UCI data – Model complexity



Robustness

- Accuracy-targeting EOP
  - identifies which portions of the data can be confidently classified with a given rate.

Accuracy of EOP when regions do not include noisy data



Outline

- Motivation: need for interpretable models
- Overview of data analysis tools
- Model evaluation – accuracy vs complexity
- Model evaluation – understandability
- Example applications
- Summary

Metrics of Explainability

Lift

$$L(A \rightarrow B) = \frac{p(B|A)}{p(B)} = \frac{n \cdot n_{AB}}{n_A n_B}$$

Bayes Factor

$$BF(A \rightarrow B) = \frac{p(A|B)}{p(A|\overline{B})} = \frac{n_{AB} n_{\overline{B}}}{n_B n_{A\overline{B}}}$$

J-Score

$$J(A \rightarrow B) = p(A)\left(p(B|A)\log\frac{p(B|A)}{p(B)} + (1-p(B|A))\log\frac{1-p(B|A)}{1-p(B)}\right)$$

Normalized Mutual Information

$$NMI(A \rightarrow B) = \frac{\left(\sum_{i=1}^{d} p(a_i, b)\log_2\frac{p(a_i,B)}{p(a_i)p(b)}\right)}{-\sum_{i=1}^{d} p(a_i)\log_2 p(a_i)}$$

Evaluation with usefulness metrics

- For 3 out of 4 metrics, EOP beats CART

	CART				EOP			
	BF	L	J	NMI	BF	L	J	NMI
MB	1.982	0.004	0.389	0.040	1.889	0.007	0.201	0.502
BCW	1.057	0.007	0.004	0.011	2.204	0.069	0.150	0.635
BT	0.000	0.009	0.210	0.000	Inf	0.021	0.088	0.643
V	Inf	0.020	0.210	0.010	2.166	0.040	0.177	0.383
Mean	1.520	0.010	**0.203**	0.015	**2.047**	**0.034**	0.154	**0.541**

BF =Bayes Factor. L = Lift. J = J-score. NMI = Normalized Mutual Info

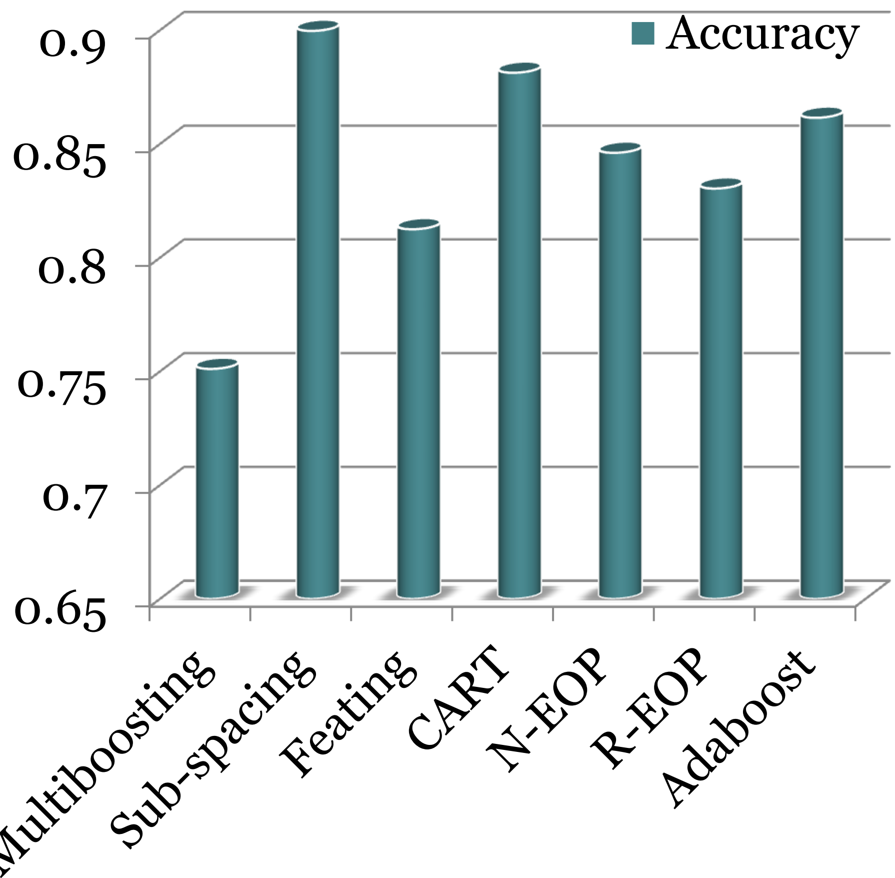
Higher values are better

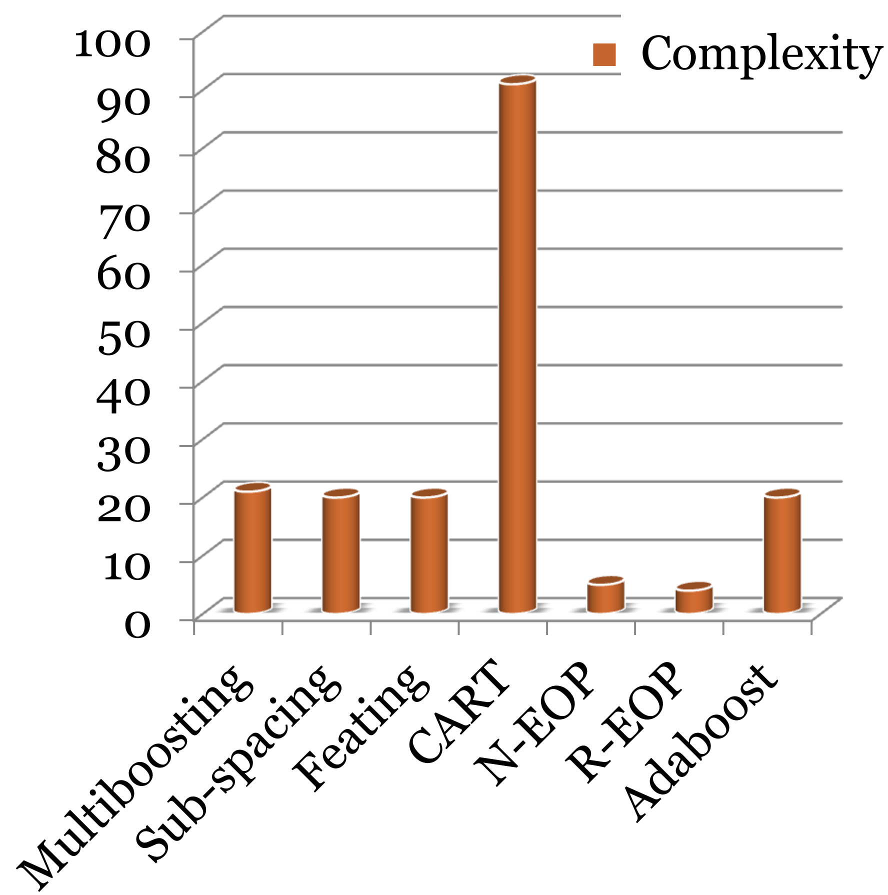
Outline

- Motivation: need for interpretable models
- Overview of data analysis tools
- Model evaluation – accuracy vs complexity
- Model evaluation – understandability
- Example application
- Summary

Spam Detection (UCI ‘SPAMBASE’)

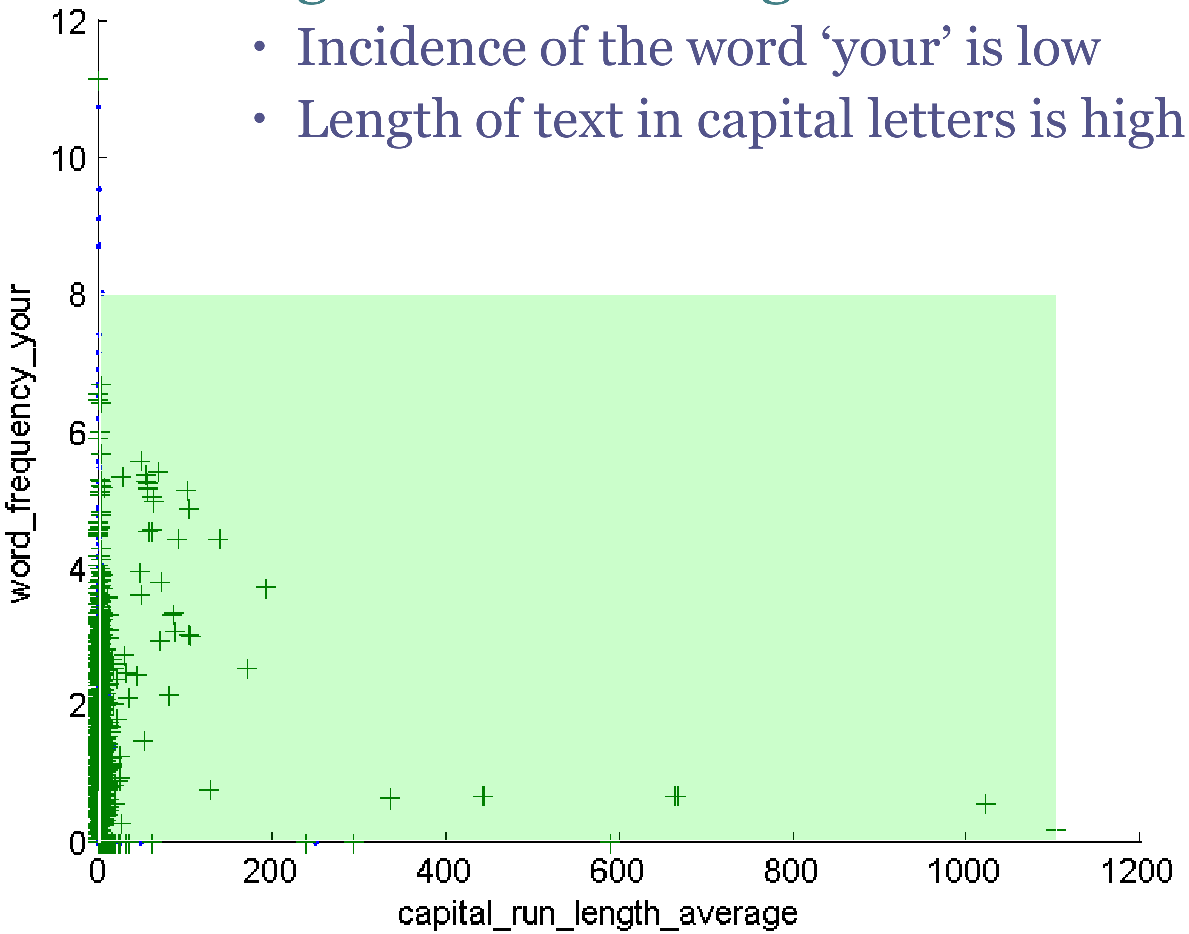
- 10 features: frequencies of misc. words in e-mails
- Output: spam or not



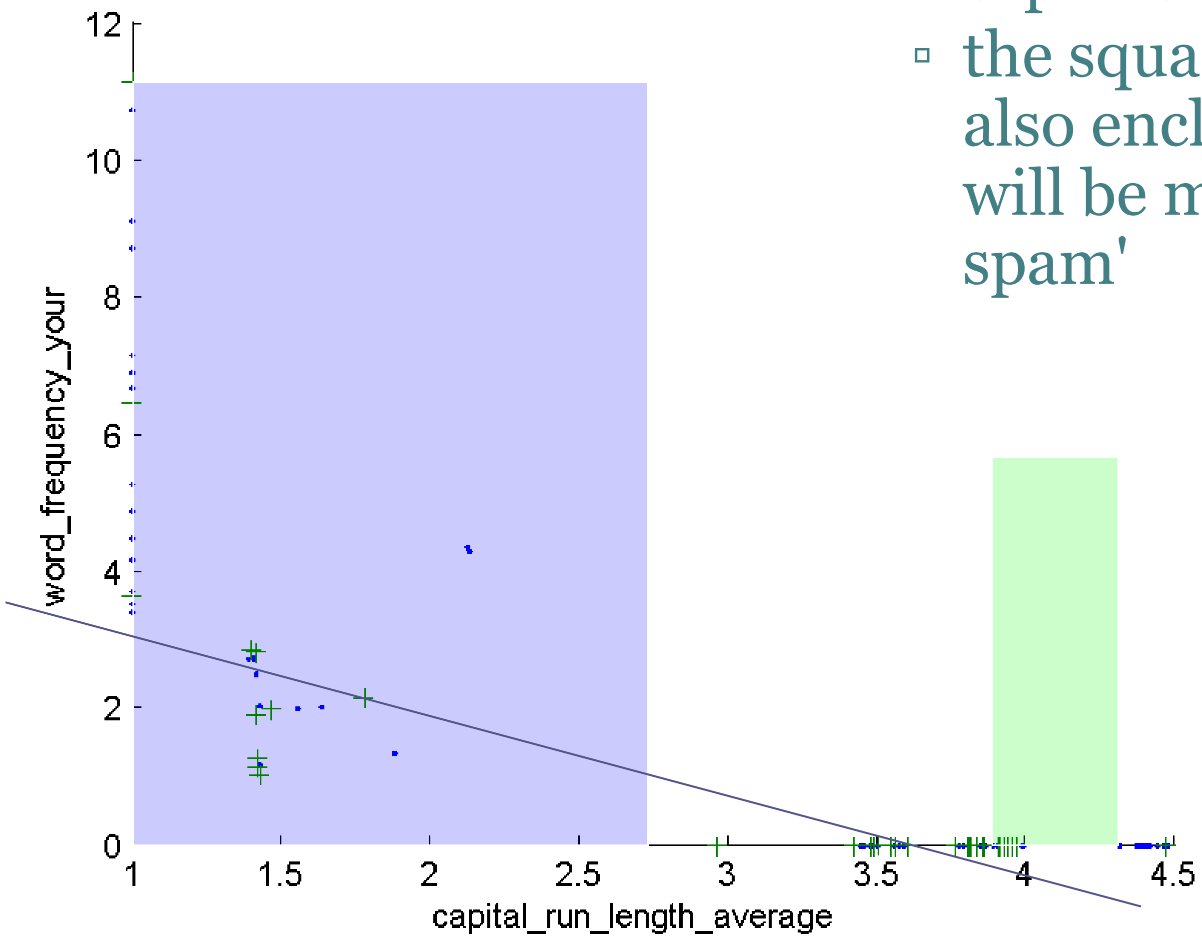


Spam Detection – Iteration 1

- classifier labels everything as spam
- high confidence regions do enclose mostly spam and:
  - Incidence of the word 'your' is low
  - Length of text in capital letters is high



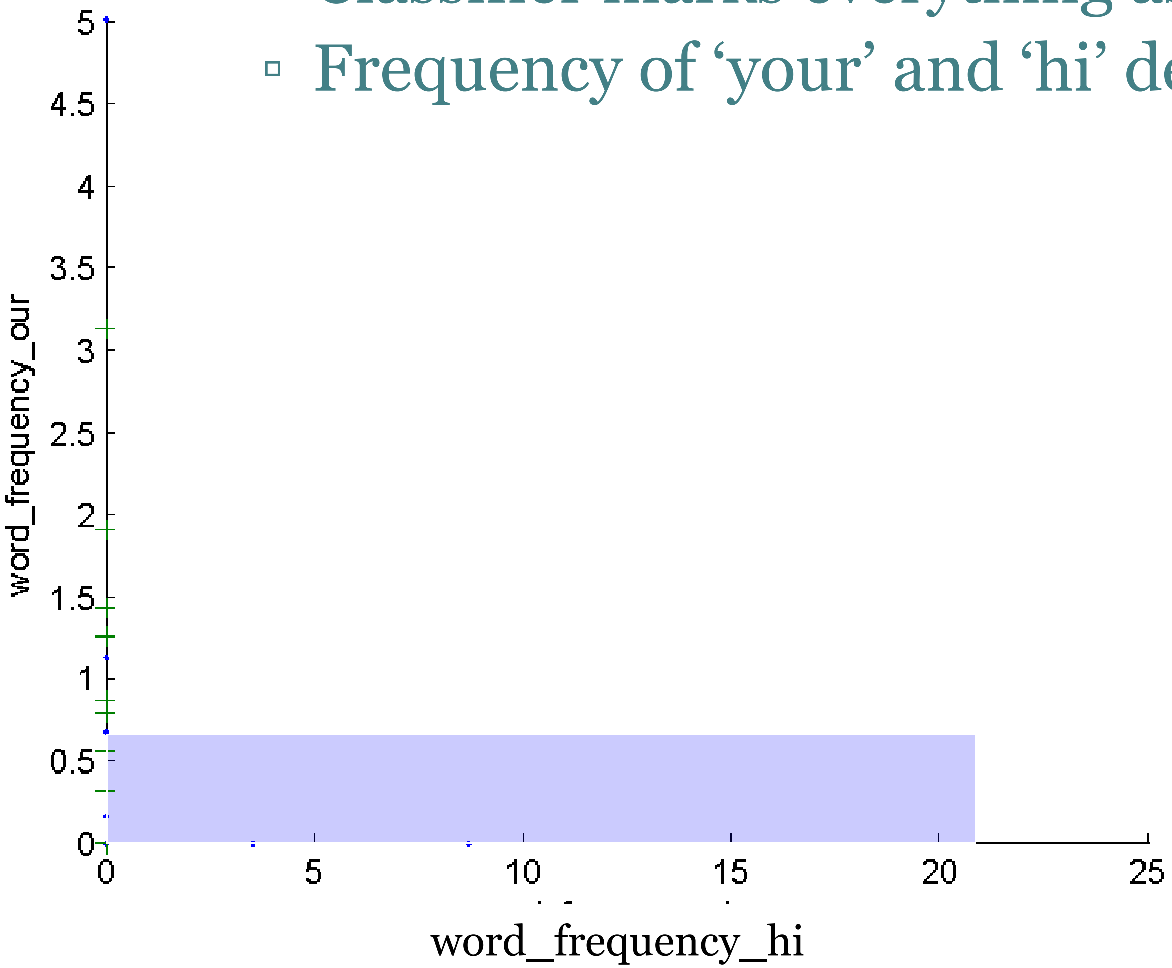
Spam Detection – Iteration 2



- the required incidence of capitals is increased
- the square region on the left also encloses examples that will be marked as `not spam'

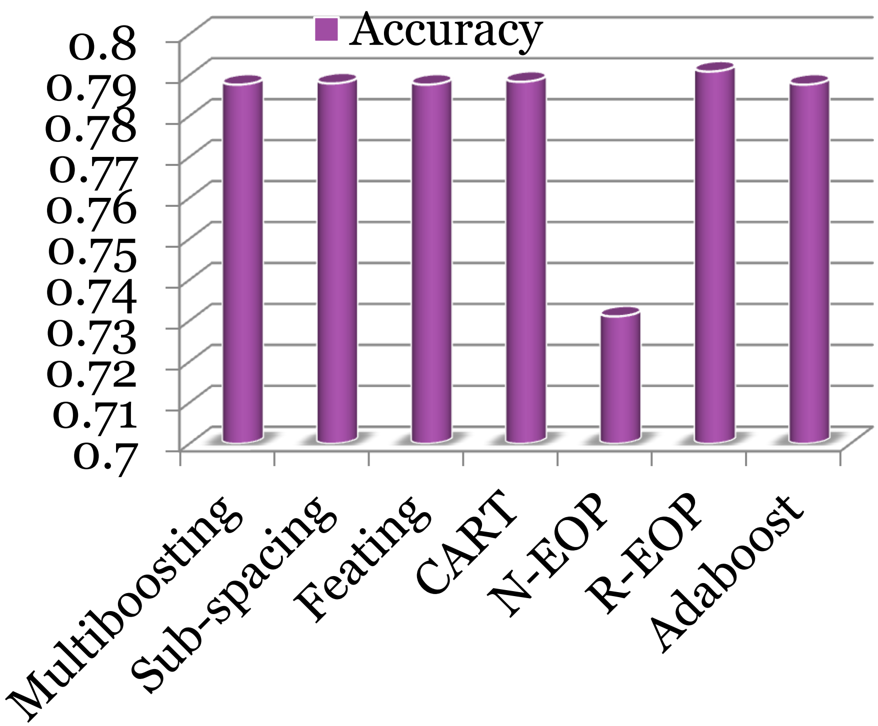
Spam Detection – Iteration 3

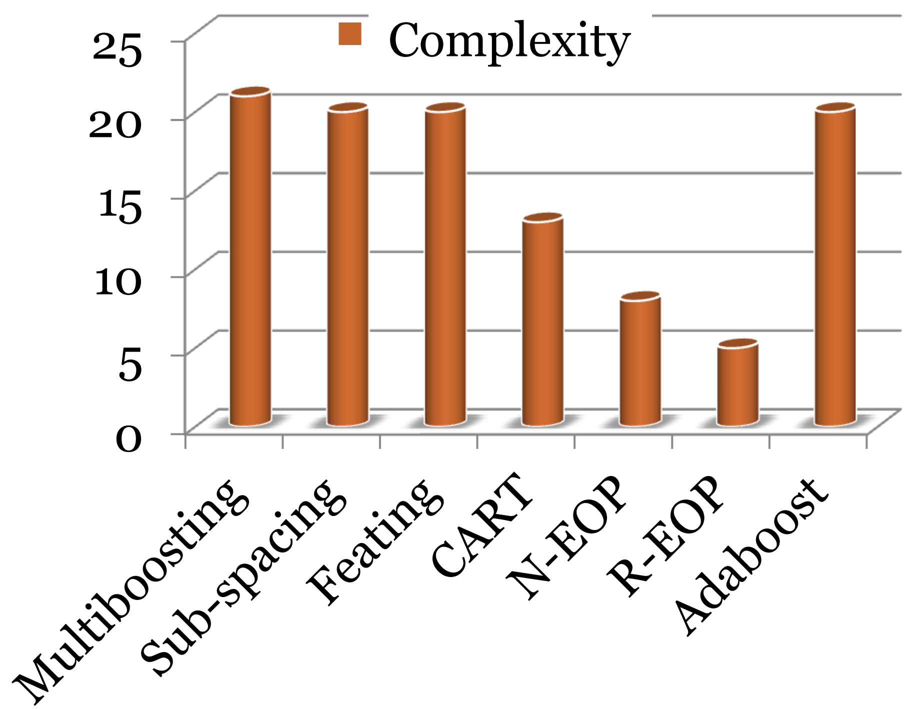
- Classifier marks everything as spam
- Frequency of ‘your’ and ‘hi’ determine the regions

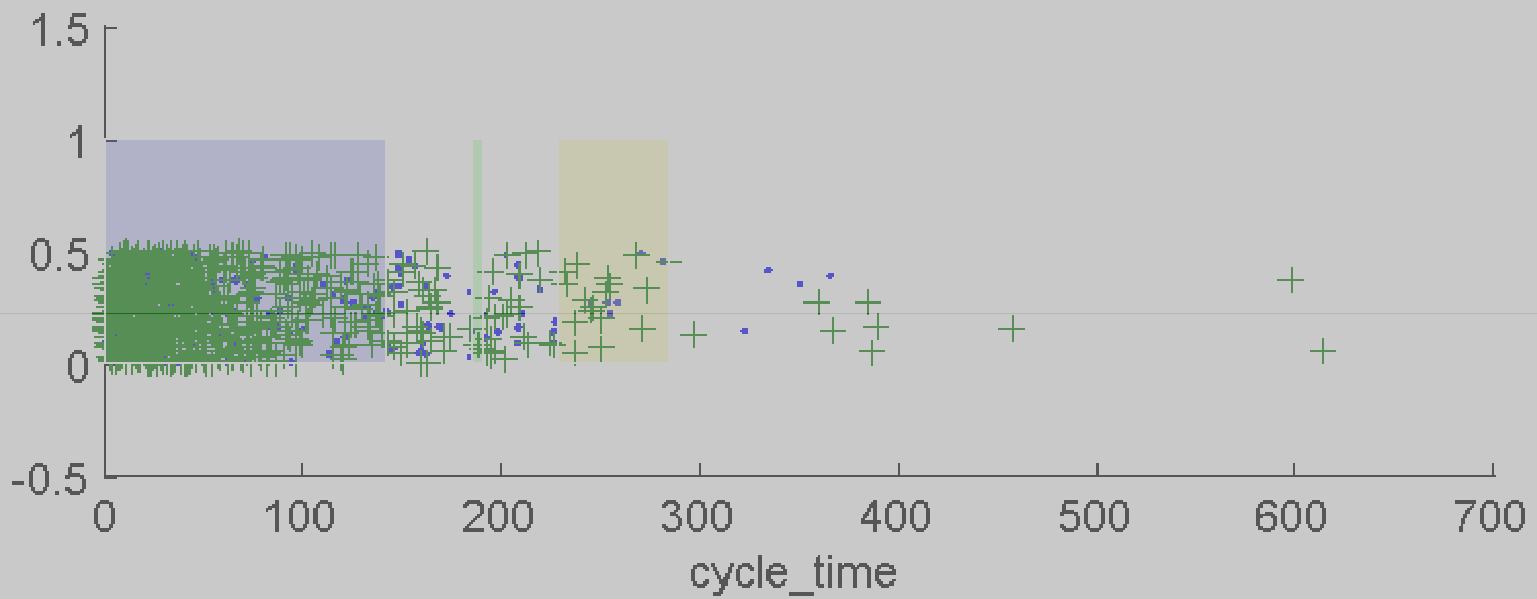


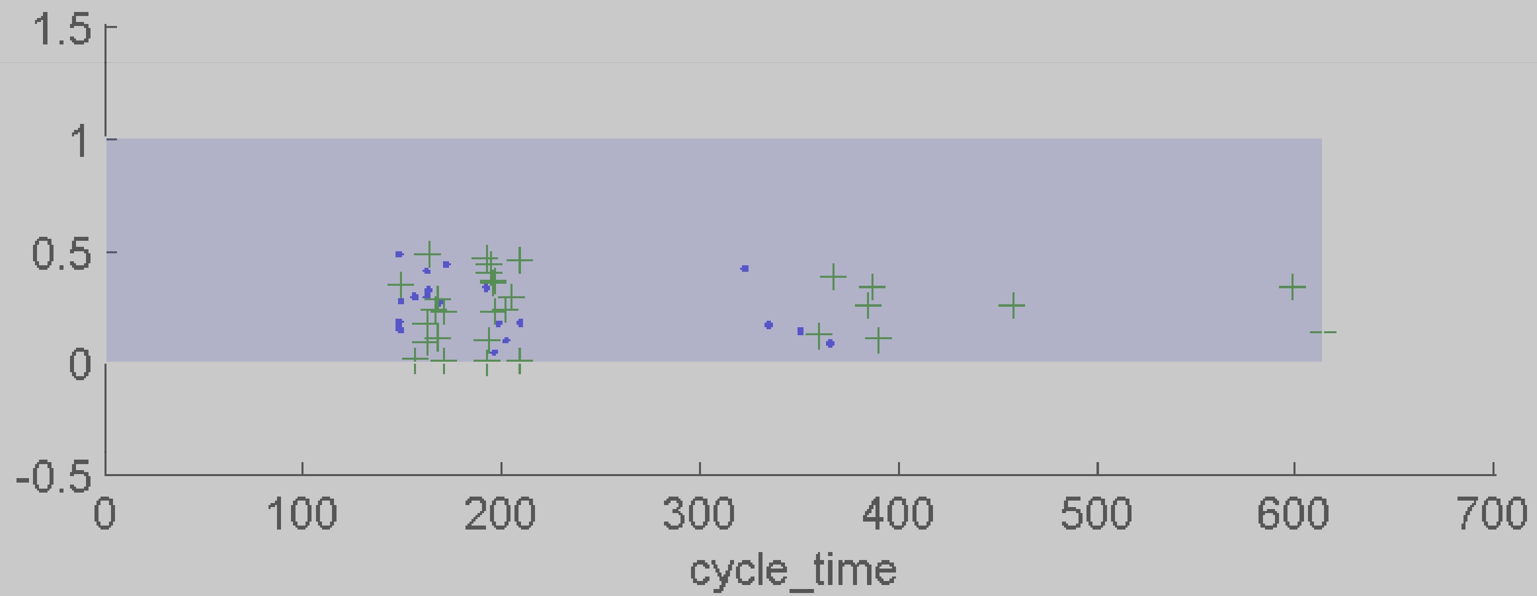
Effects of Cell Treatment

- Monitored population of cells
- 7 features: cycle time, area, perimeter …
- Task: determine which cells were treated



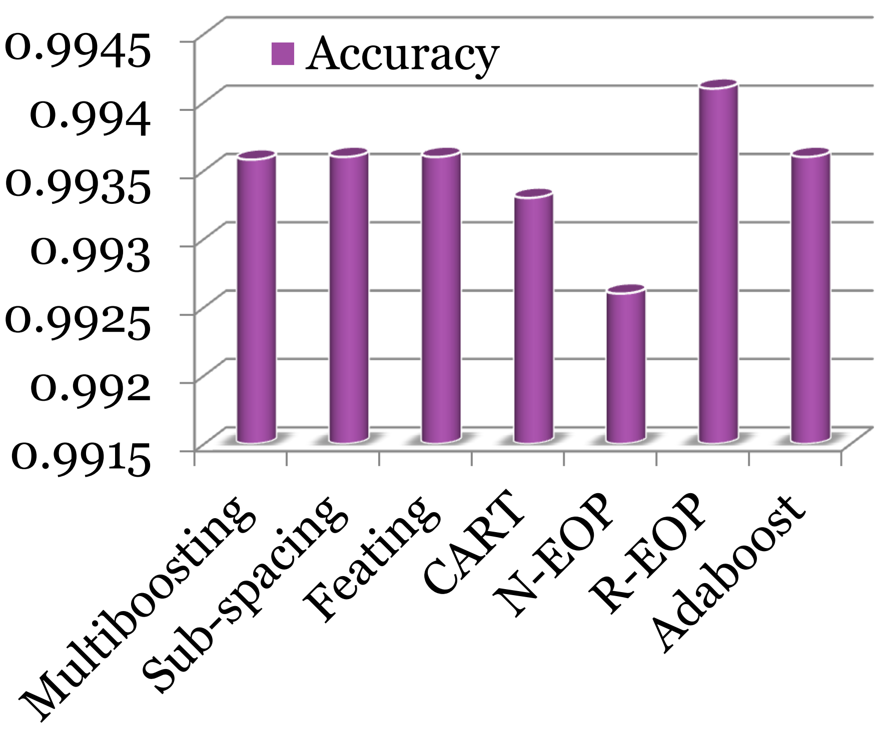


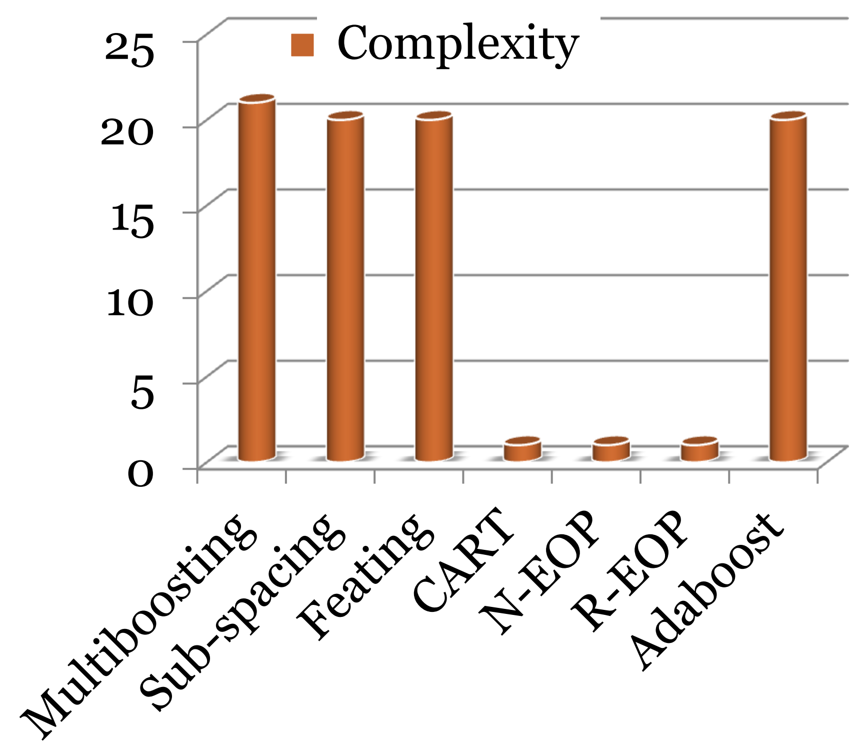

1.5
1
0.5
0
-0.5
0
100
200
300
400
500
600
700
cycle_time

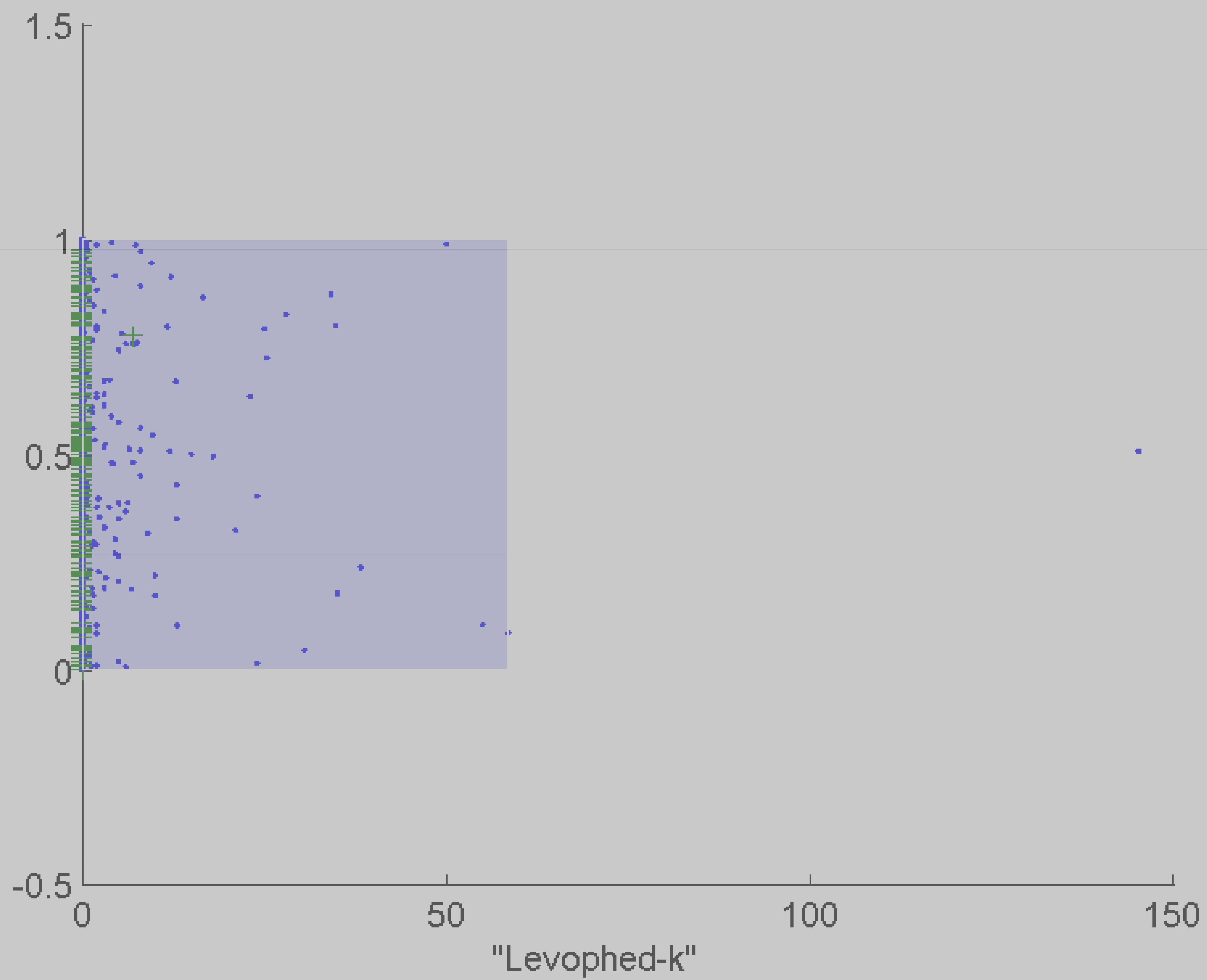

1.5
1
0.5
0
-0.5
0
100
200
300
400
500
600
700
cycle_time

Mimic Medication Data

- Information about administered medication
- Features: dosage for each drug
- Task: predict patient return to ICU

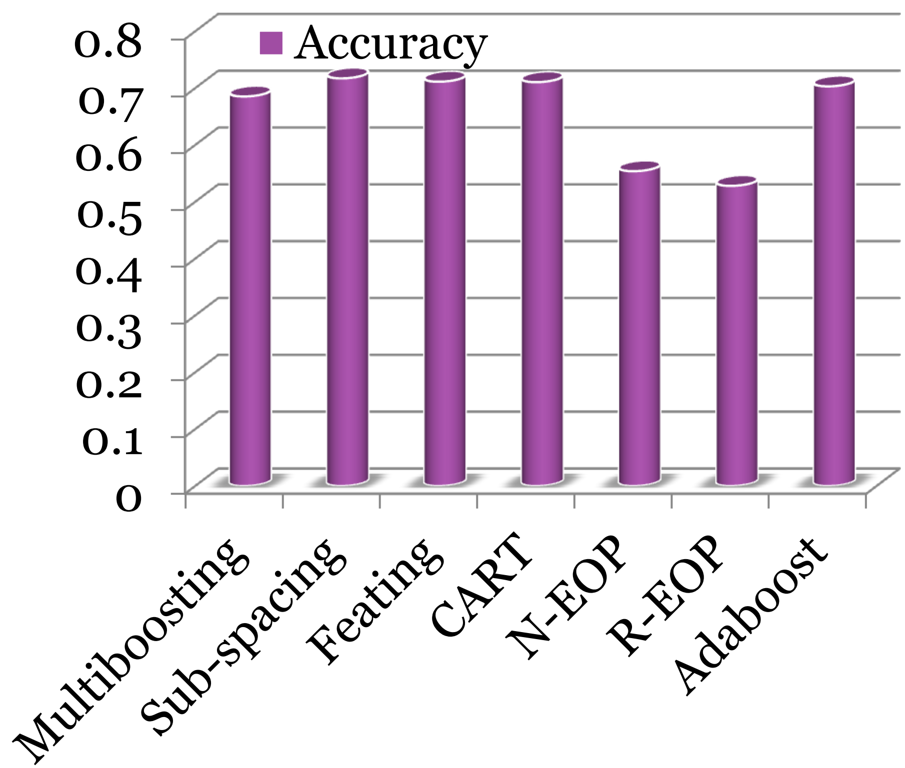


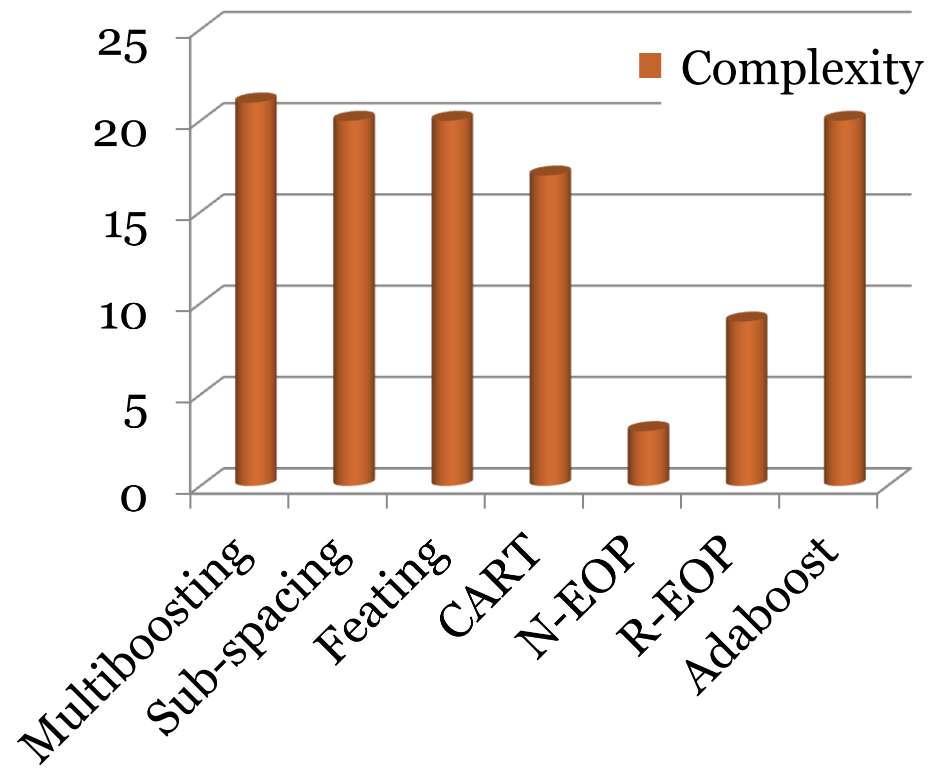


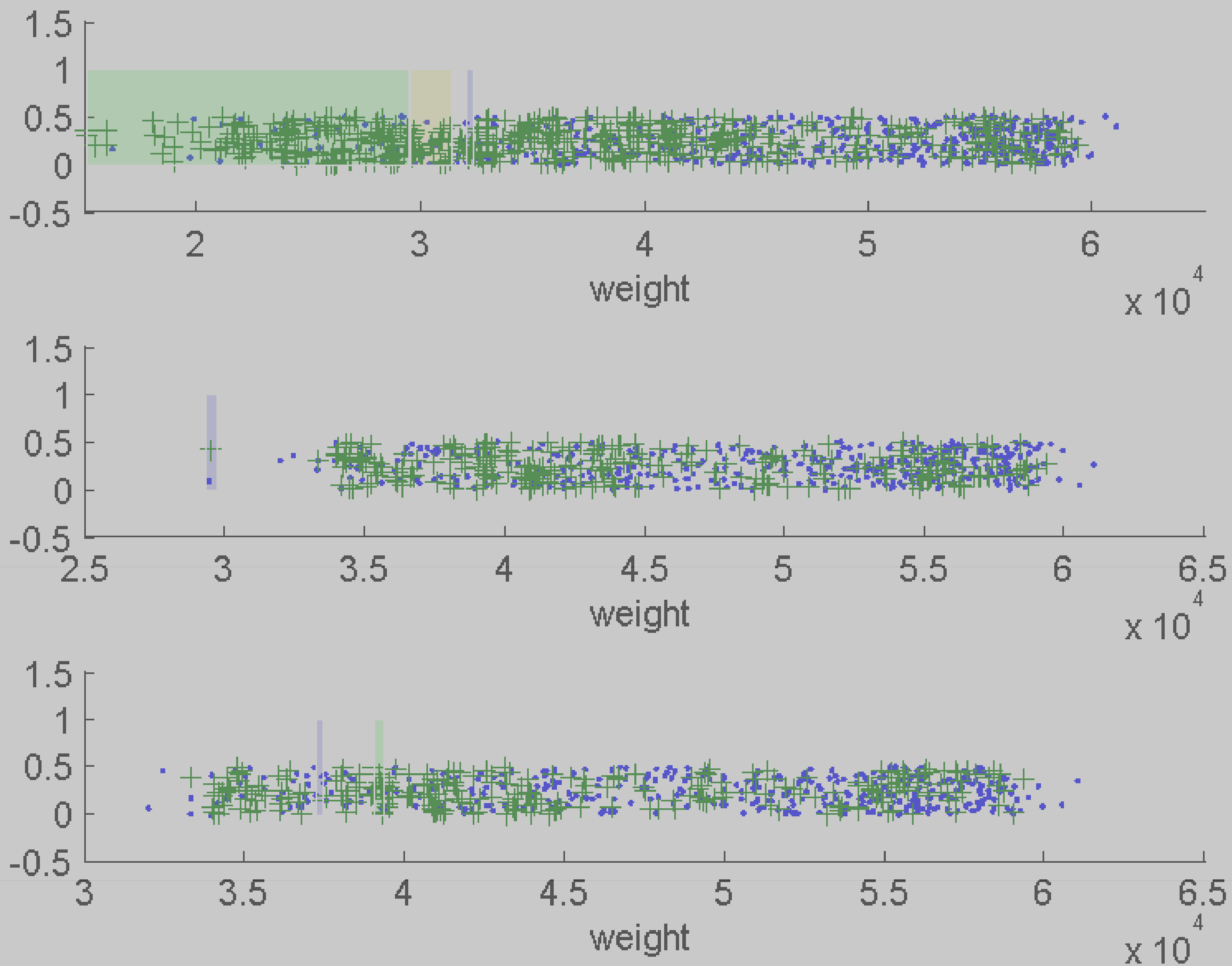

1.5
1
0.5
0
-0.5
0
50
100
150
"Levophed-k"

Predicting Fuel Consumption

- 10 features: vehicle and driving style characteristics
- Output: fuel consumption level (high/low)






1.5
1
0.5
0
-0.5
2
3
4
5
6
weight
x 10^4
1.5
1
0.5
0
-0.5
2.5
3
3.5
4
4.5
5
5.5
6
6.5
weight
x 10^4
1.5
1
0.5
0
-0.5
3
3.5
4
4.5
5
5.5
6
6.5
weight
x 10^4

Nuclear threat detection data

- Random Forests accuracy: 0.94
- Rectangular EOP accuracy: 0.881

… but

No match

Regions found in 1st iteration for Fold 0:
- incident.riidFeatures.SNR [2.90,9.2]
- Incident.riidFeatures.gammaDose [0,1.86]*10^{-8}

Regions found in 2st iteration for Fold 1:
- incident.rpmFeatures.gamma.sigma [2.5, 17.381]
- incident.rpmFeatures.gammaStatistics.skewdose [1.31,…]

Summary

- White box models (CART, Feating, Sub-spacing)
  - ~ as accurate as typical black-box models - B, MB
- In most cases EOP:
  - *maintains* accuracy
  - *reduces* complexity
  - identifies *useful* aspects of the data
- EOP *wins* in terms of *expressiveness*
- Trade-offs
  - Accuracy vs Complexity
  - Accuracy vs Coverage
- Open questions:
  - What if no good low-dimensional projections found?
  - What to do with inconsistent models in different folds of cv?