Matching QSAR Sets

SPE & Clustering and Weighted/Sampled Classification

Rajarshi Guha

Penn State University

Stochastic Proximity Embedding

- Multidimensional scaling algorithm
- A stochastic form of a steepest descent algorithm
- Linear scaling (good for large datasets)
- In principle allows you to get the *intrinsic dimension* of the dataset

Stochastic Proximity Embedding

- There are two main parameters that must be set: r_c and the final embedding dimension
- Quality of embedding is measured by the Sammon stress
- Ideally, in the *intrinsic dimensionality* the stress will be 0 or very close
- Currently the optimal parameters are obtained by an exhaustive search: $.1 < r_c < 1$ and $2 < D_{\mathrm{emb}} < D_{\mathrm{inp}}$

Stochastic Proximity Embedding

- Strategy
  - Find optimal r_c and $D_{\mathbf{emb}}$
  - Cluster the dataset on the reduced coordinates
  - Badly predicted points should lie outside main clusters
- Problem
  - It assumes that the dataset can be clustered well

Classification

Weighted LDA

- Unweighted LDA is very biased towards the good class
- Used the artemisinin dataset, with model descriptors (4)

TSET Confusion Matrix

	b	g	✓
b	0	50	0%
g	0	111	100%

PSET Confusion Matrix

	b	g	✓
b	0	4	0%
g	0	14	100%

Weighted LDA

- We can provide prior weights for the 2 classes
- First guess is to let $W_{bad} = W_{good} = 0.5$
- The good class looses out and PSET is very poorly predicted

TSET Confusion Matrix

	b	g	✓
b	32	18	64%
g	73	38	34%

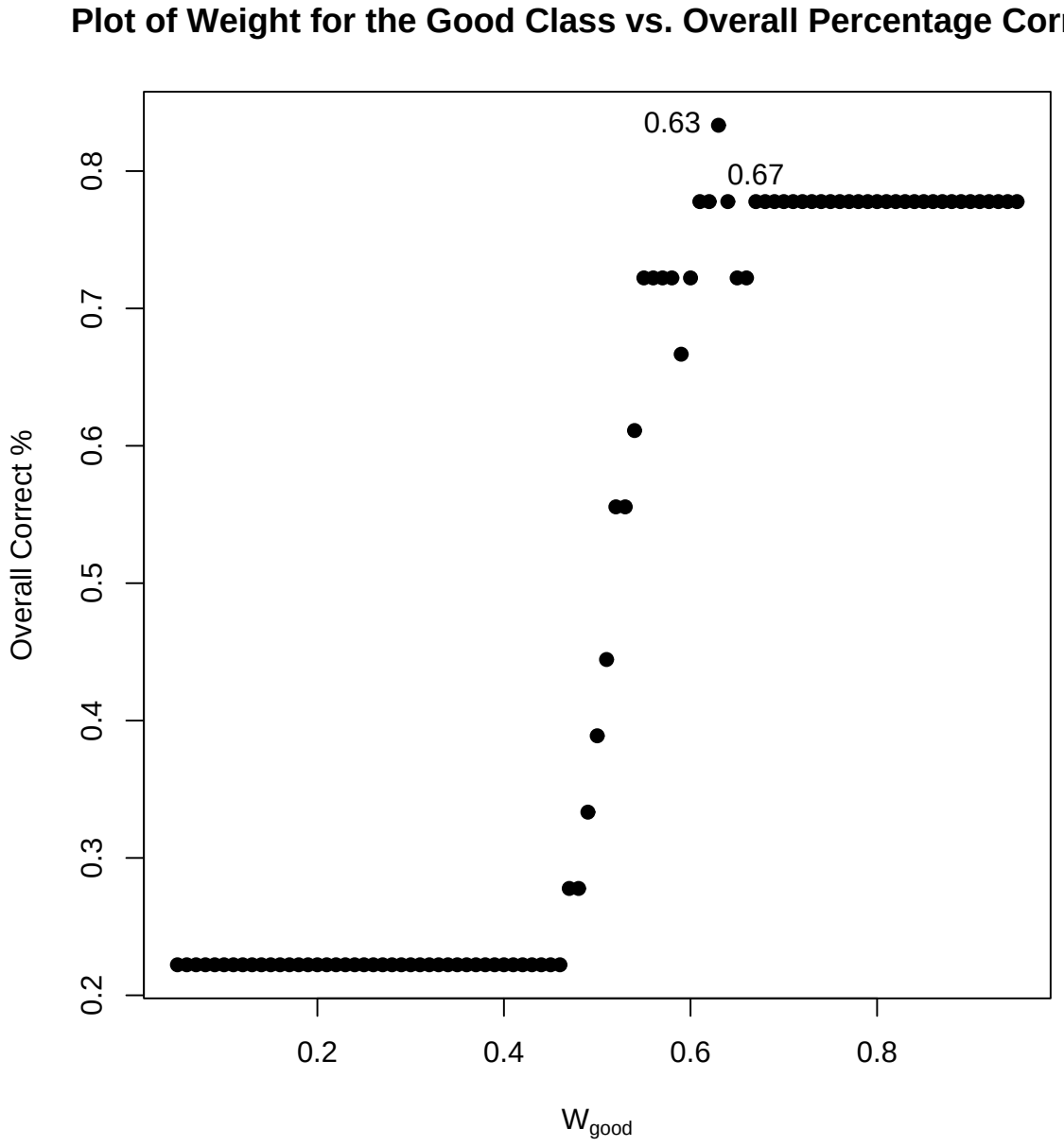
PSET Confusion Matrix

	b	g	✓
b	4	0	100%
g	11	3	21%

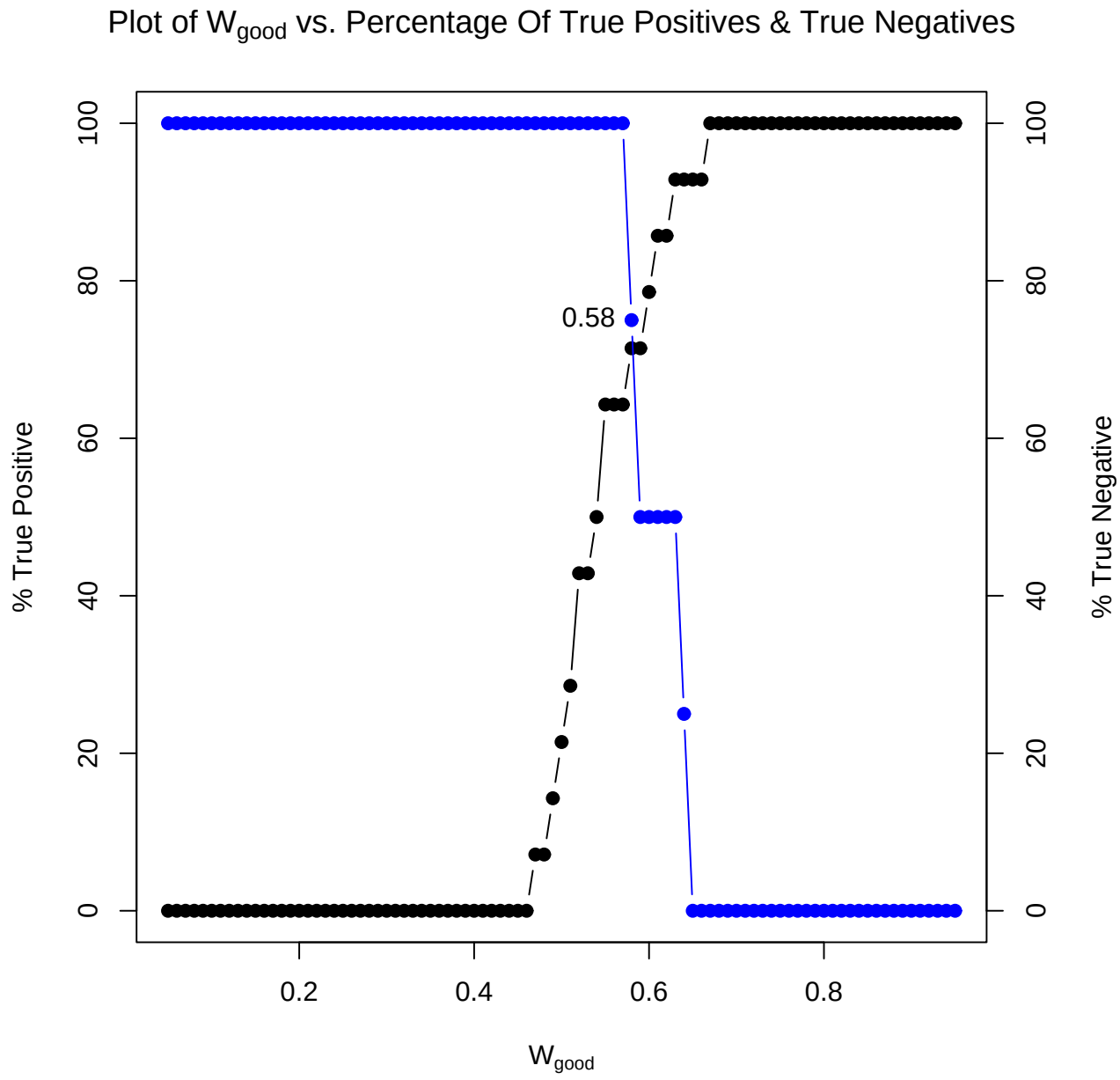
Weighted LDA

- What are we trying to optimize?
  - True positives
  - True negatives
  - Overall correct
- How can we choose weights?
  - Look at overall correct vs. $\boldsymbol{W}_{good}$
  - Look at how true positive and true negative rates vary with $\boldsymbol{W}_{good}$
  - Look at false positive vs true positive (ROC curve)

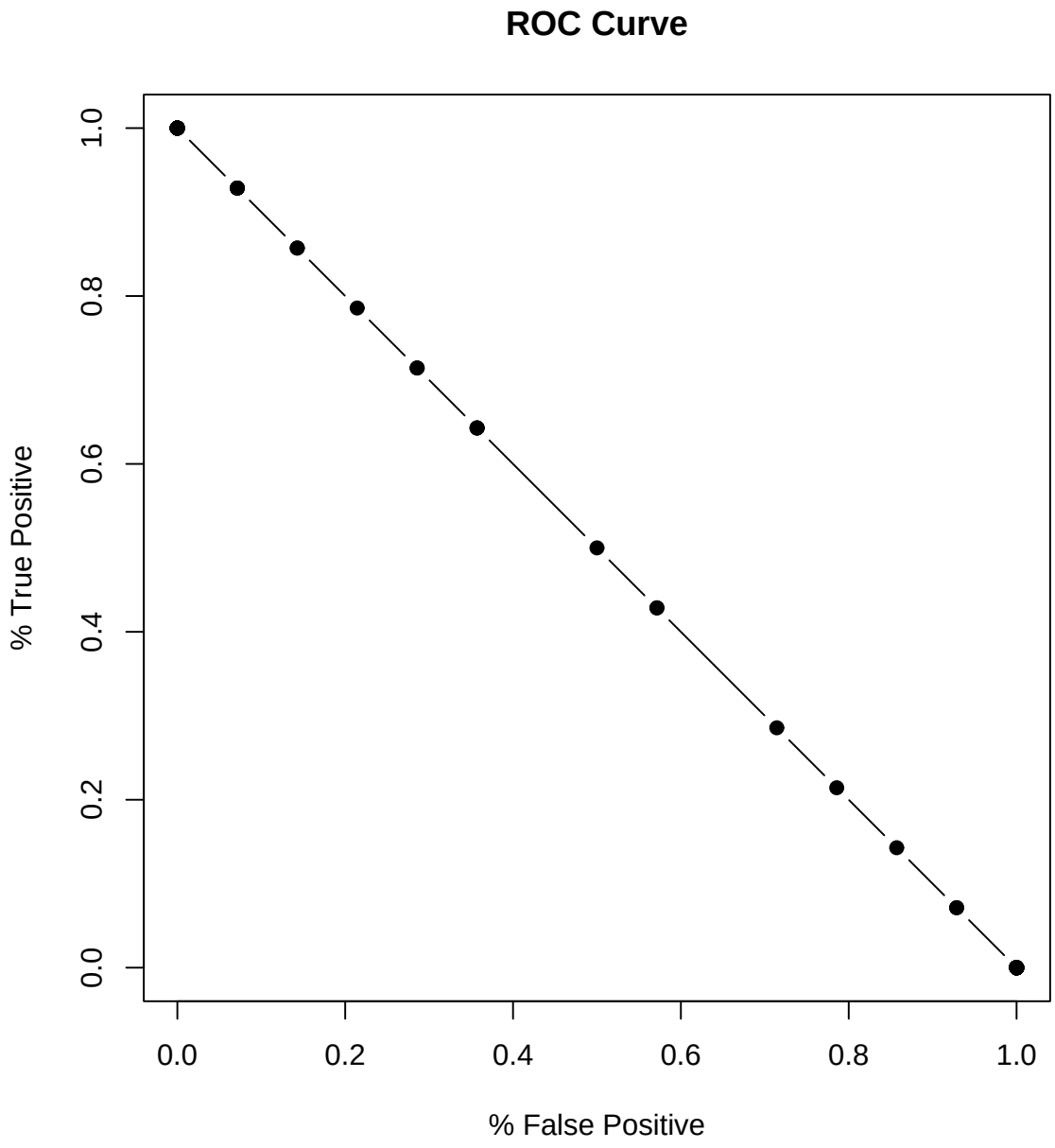
Weighted LDA



Weighted LDA



Weighted LDA



PC Classification

- Use the PC's (i.e., rotated data) as feature vectors
- Not apparent how many to take, so trial & error!
- However for discrimination purposes kernel-PLS has been shown to be more useful [a]
- Strategy
  - Evaulate TSET PC's
  - Find minimum number (n) of PC's that give best classification
  - Evaluate PSET PC's
  - Use n PSET PC's to classify the PSET

[a] Barker et al., *J. Chemom.*, **2003**, *17*, 166-173

PC Classification (Artemisinin)

PC Classification (Artemisinin)

- Results using 60 PC's

TSET Confusion Matrix

	b	g	✓
b	50	0	100%
g	0	111	100%

PSET Confusion Matrix

	b	g	✓
b	3	1	75%
g	1	13	92%

Jarvis Patrick Clustering And Classification of Residuals

JP - Overview

- kNN based classification scheme
- Molecules are in the same class if
  - they are in each others J neighbor list
  - they have K neighbors in common
- Lots of scope for tweaking
- Fast algorithm

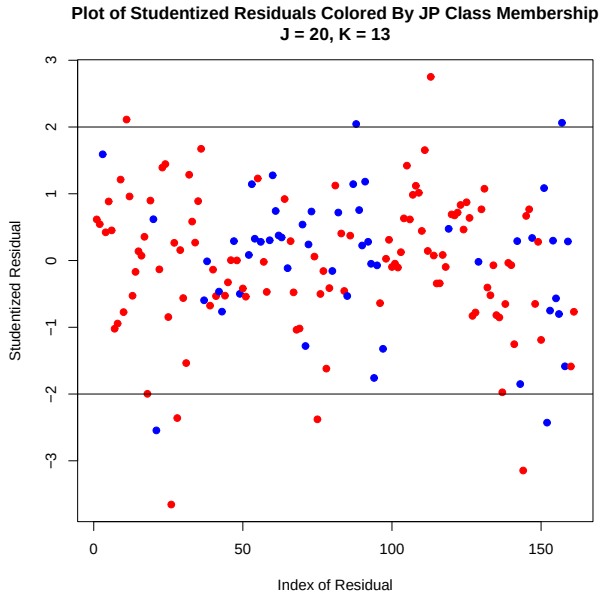
How Well Does JP Classify?

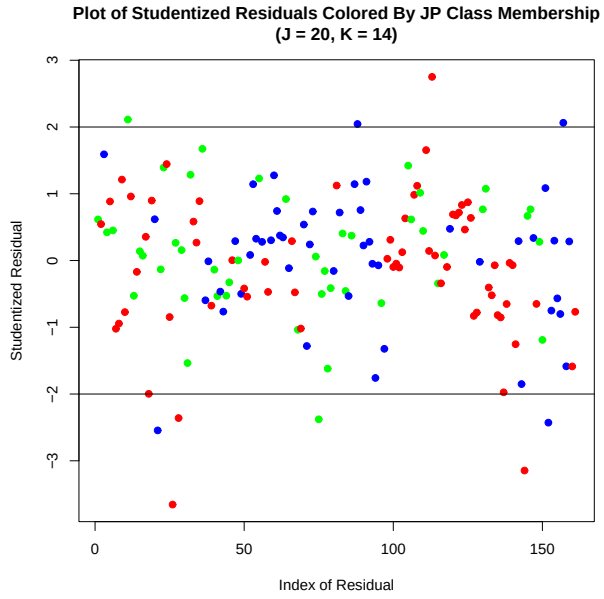
- How do we determine the quality of classification?
  - Look at AP similarity values within a class
  - Compare average AP similarity value between classes
- However, since the algorithm is based on similarities in descriptor space this may not carry over to similarities in *AP space*

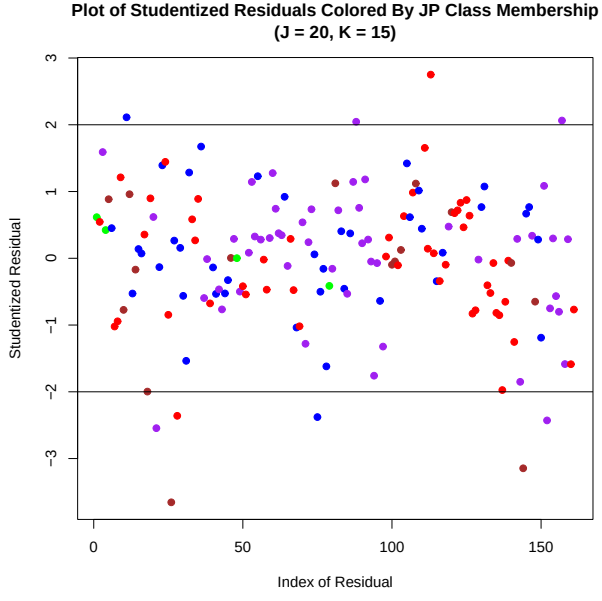
JP & AP Similarities

Artemisinin, TSET	
No. Class	Average In-Class AP Similarity
2	y
3	0.36, 0.37, 0.40
5	b

JP - Varying J & K







- Artemisinin dataset
- Only the TSET is considered
- All reduced pool descriptors were used
- $J = 20$ chosen arbitrarily