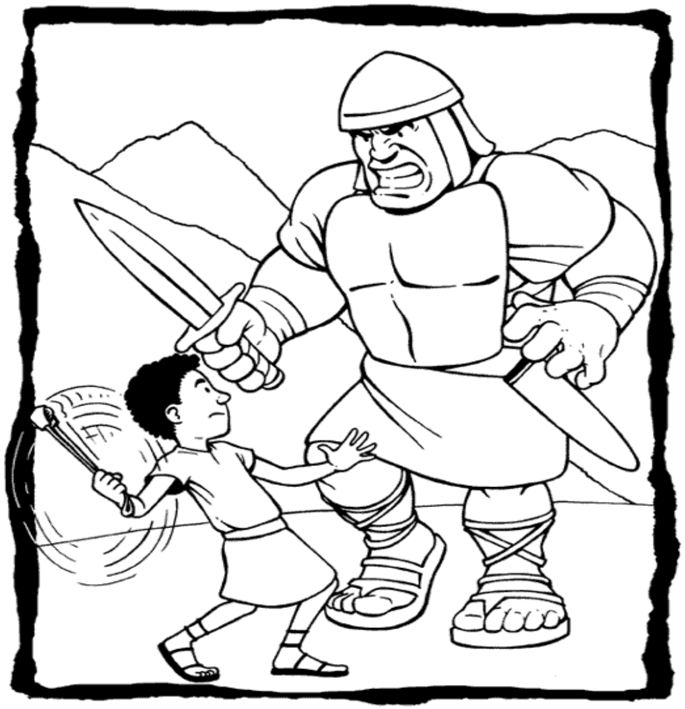
Instance-based Reasoning (Less is More!)



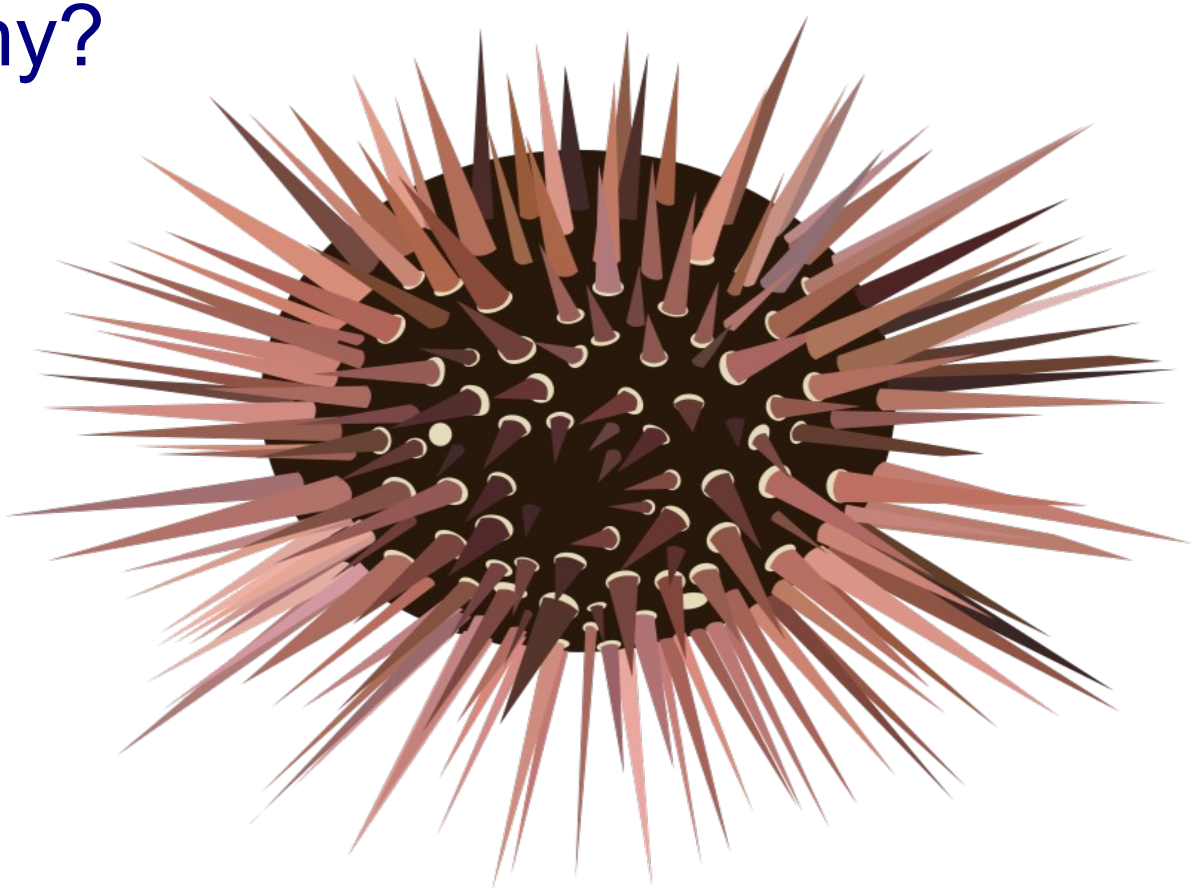
David & Goliath

1 Samuel 17:1–58

Fayola Peters
Lane Department of Computer Science and Electrical Engineering,
West Virginia University
fpeters@mix.wvu.edu

The Problem?

- Only a few instances matter...
- But why?



Outline

- **Previous research → Few instances matter**
- Why? - The Answer lies in the E(k) matrix
- Now - we exploit instance space

Previous Research = Less is More!

- **Chang 1974 – Finding Prototypes for Nearest Neighbor Classifiers**
- Kim 2011 – Dealing with Noise in Defect Prediction
- Kocaguneli 2011 – Exploiting the Essential Assumptions of ABE Estimation
- Kocaguneli 2010 – When to use data from other projects for effort estimation
- Experiment: Independent Variable Mutation
- Experiment: Bias/Variance

In the Beginning

- Chang 1974, realized that few instances matter
- His experimental results...

RECOGNITION RATES FOR LIVER DISEASE DATA

Classifiers	Training Set (514 cases) Recognition Rate (%)	Training Set (514 cases) Error Rate (%)	Test Set (120 cases) Recognition Rate (%)	Test Set (120 cases) Error Rate (%)
The Nearest Neighbor Classifier Using 514 Initial Prototypes	100	0	92.5	7.5
The Nearest Neighbor Classifier Using 34 Final Prototypes	100	0	91.7	8.3

Previous Research = Less is More!

- Chang 1974 – Finding Prototypes for Nearest Neighbor Classifiers
- **Kim 2011 – Dealing with Noise in Defect Prediction**
- Kocaguneli 2011 – Exploiting the Essential Assumptions of ABE Estimation
- Kocaguneli 2010 – When to use data from other projects for effort estimation
- Experiment: Independent Variable Mutation
- Experiment: Bias/Variance

Noise Reduction is Important

- Kim 2011, noise affects results in defect prediction
- Therefore eliminating noise improves results

Table 4.The defect prediction performance (F-measure) after identifying and removing noisy instances (SWT)

Remove Noises ?	Noise Rate	Bayes Net	Naïve Bayes	SVM	Bagging
No	15%	0.781	0.305	0.594	0.841
	30%	0.777	0.308	0.339	0.781
	45%	0.249	0.374	0.353	0.350
Yes	15%	0.793	0.429	0.797	0.838
	30%	0.802	0.364	0.706	0.803
	45%	0.762	0.418	0.235	0.505

Previous Research = Less is More!

- Chang 1974 – Finding Prototypes for Nearest Neighbor Classifiers
- Kim 2011 – Dealing with Noise in Defect Prediction
- **Kocaguneli 2011 – Exploiting the Essential Assumptions of ABE Estimation**
- Kocaguneli 2010 – When to use data from other projects for effort estimation
- Experiment: Independent Variable Mutation
- Experiment: Bias/Variance

TEAK

- TEAK → Test Essential Assumption Knowledge
- TEAK's design
  - Select a prediction system.
  - Identify the predictor’s essential assumption(s).
  - Recognize when those assumption(s) are violated.
  - Remove those situations.
  - Execute the modified prediction system.
- Conclusion – only few instances matter.

TEAK Results

20 * LEAVE-ONE-OUT

	TEAK	LR	NNet	Best(K)	k=1	k=16	k=2	k=4	k=8
MRE									
Cocomo81	▲								
Cocomo81e	▲								
Cocomo81o	▲								
Nasa93		▲							
Nasa93c2		▲							
Nasa93c5	▲								
Desharnais		▲							
Sdr	▲								
ISBSG-Banking	▲								
Count	6	3	0	0	0	0	0	0	0
Pred(25)									
Cocomo81	▲								
Cocomo81e			▲						
Cocomo81o	▲								
Nasa93		▲							
Nasa93c2		▲							
Nasa93c5	▲								
Desharnais		▲							
Sdr	▲								
ISBSG-Banking	▲								
Count	5	3	1	0	0	0	0	0	0
AR									
Cocomo81	▲								
Cocomo81e	▲								
Cocomo81o	▲								
Nasa93		▲							
Nasa93c2		▲							
Nasa93c5	▲								
Desharnais		▲							
Sdr	▲								
ISBSG-Banking	▲								
Count	6	3	0	0	0	0	0	0	0

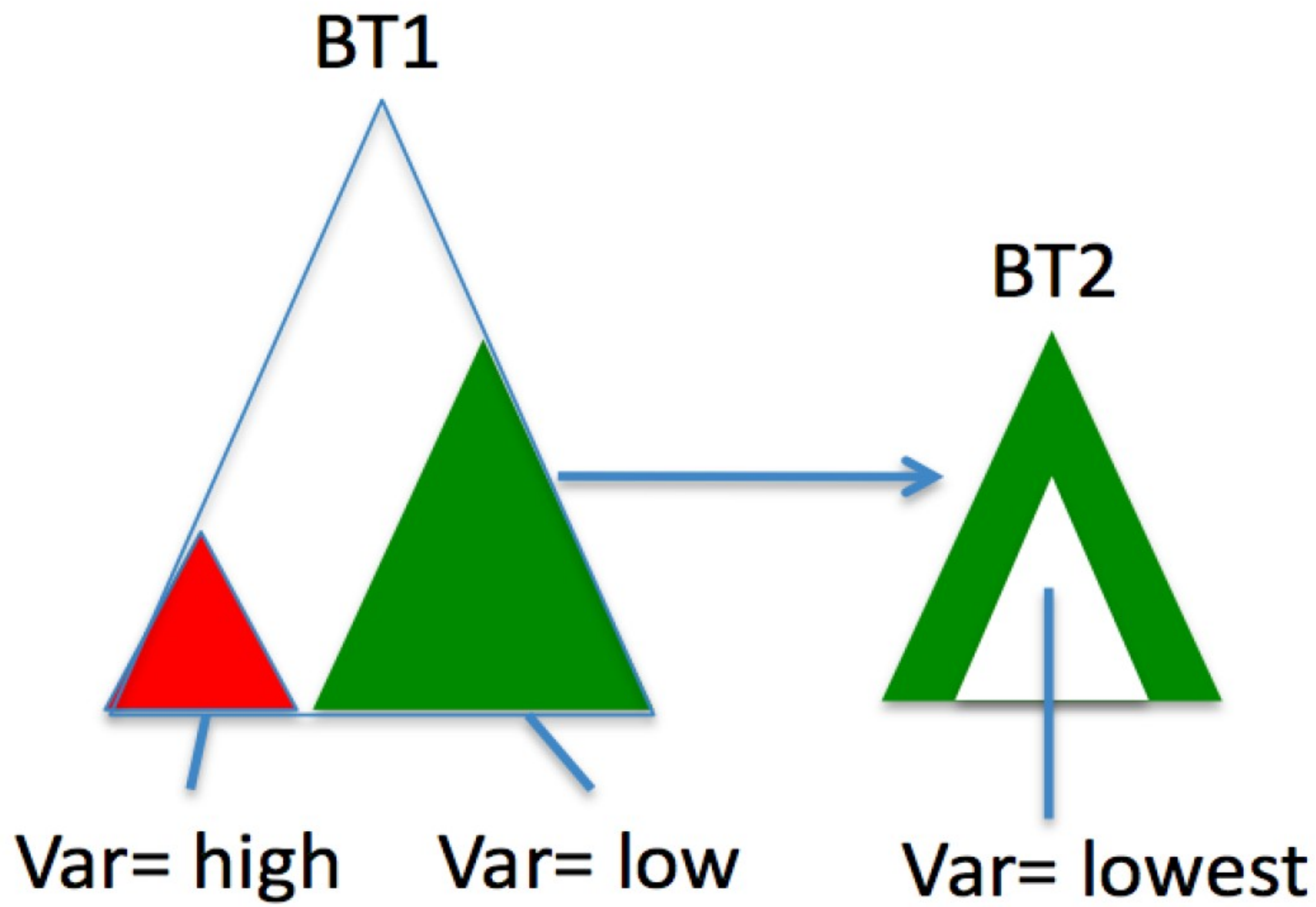
Previous Research = Less is More!

- Chang 1974 – Finding Prototypes for Nearest Neighbor Classifiers
- Kim 2011 – Dealing with Noise in Defect Prediction
- Kocaguneli 2011 – Exploiting the Essential Assumptions of ABE Estimation
- **Kocaguneli 2010 – When to use data from other projects for effort estimation**
- Experiment: Independent Variable Mutation
- Experiment: Bias/Variance

Cross Company

- Acceptable to use cross data sources once a **relevancy filter** is used
- Relevancy filter selects small subset relevant to current test case
- Removes training instances that create noise in the estimation process
- In theory, this leaves data that adheres to the principal of locality.

Using TEAK as a Relevancy Filter



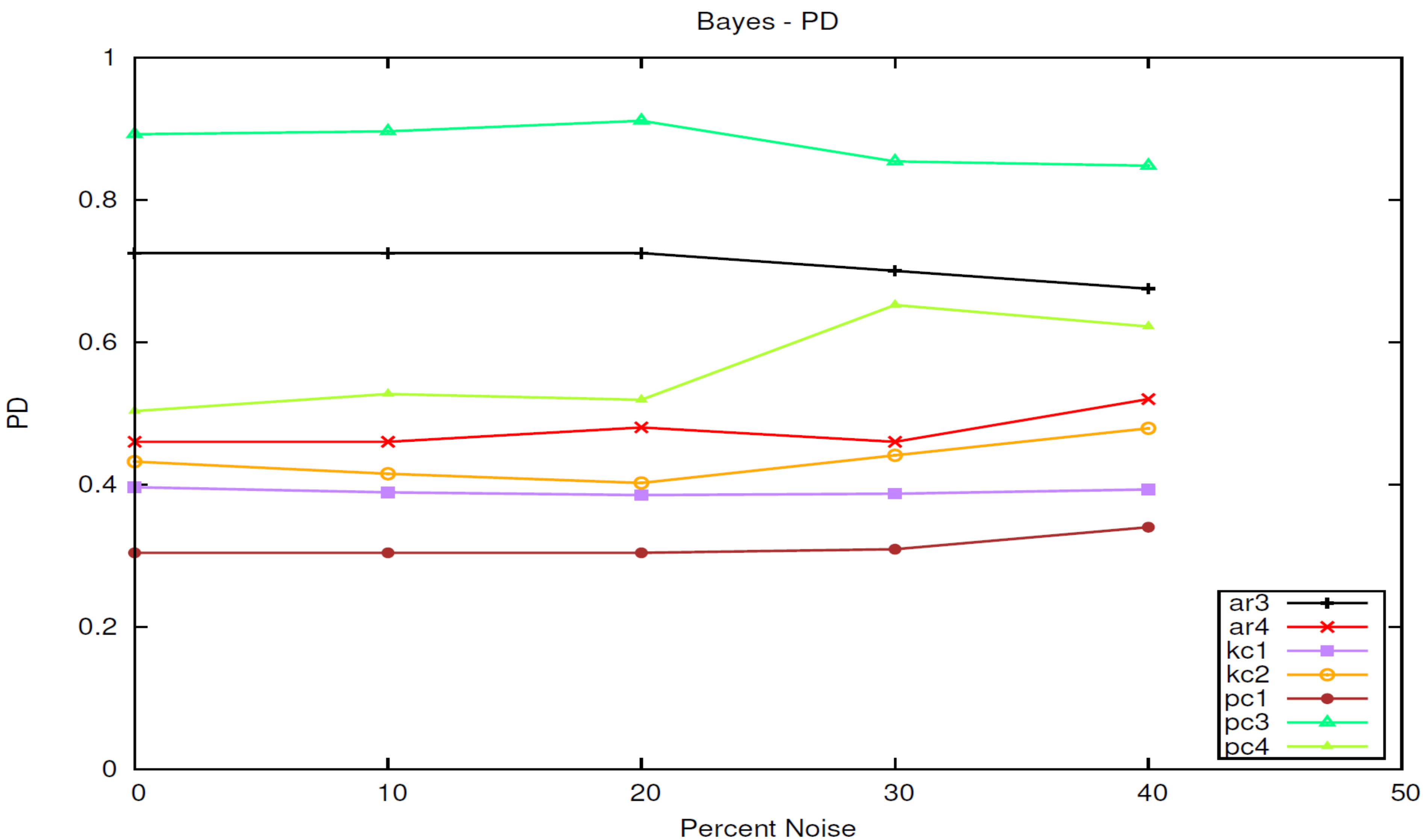
Result

Dataset	Method	Win	Tie	Loss
Coc81o	within	13	7	0
Coc81e and Coc81s	cross	0	7	13
Coc81e	within	1	19	0
Coc81o and Coc81s	cross	0	19	1
Coc81s	within	0	20	0
Coc81o and Coc81e	cross	0	20	0

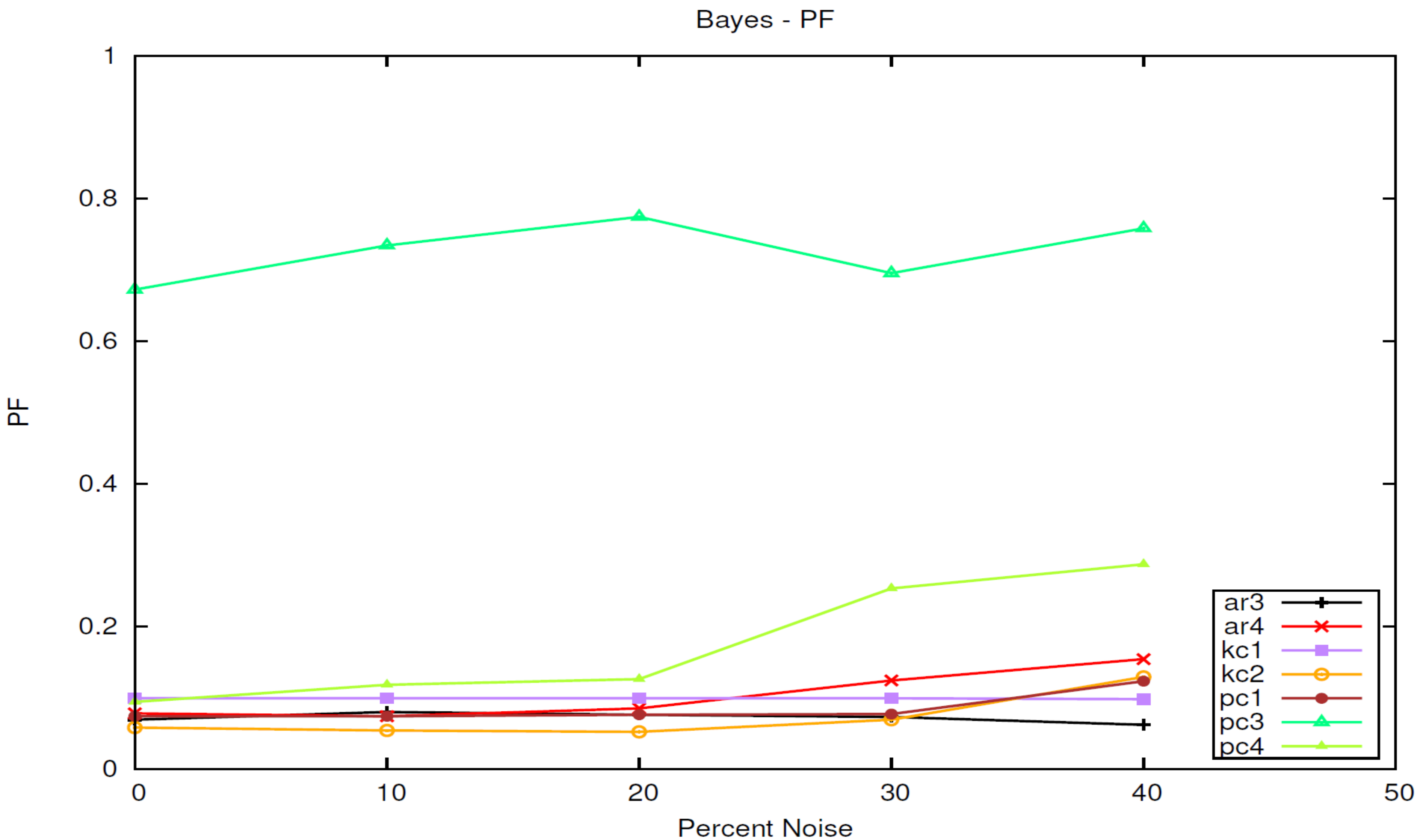
Previous Research = Less is More!

- Chang 1974 – Finding Prototypes for Nearest Neighbor Classifiers
- Kim 2011 – Dealing with Noise in Defect Prediction
- Kocaguneli 2011 – Exploiting the Essential Assumptions of ABE Estimation
- Kocaguneli 2010 – When to use data from other projects for effort estimation
- **Experiment: Independent Variable Mutation**
- Experiment: Bias/Variance

Independent Variable Mutation



Independent Variable Mutation



Previous Research = Less is More!

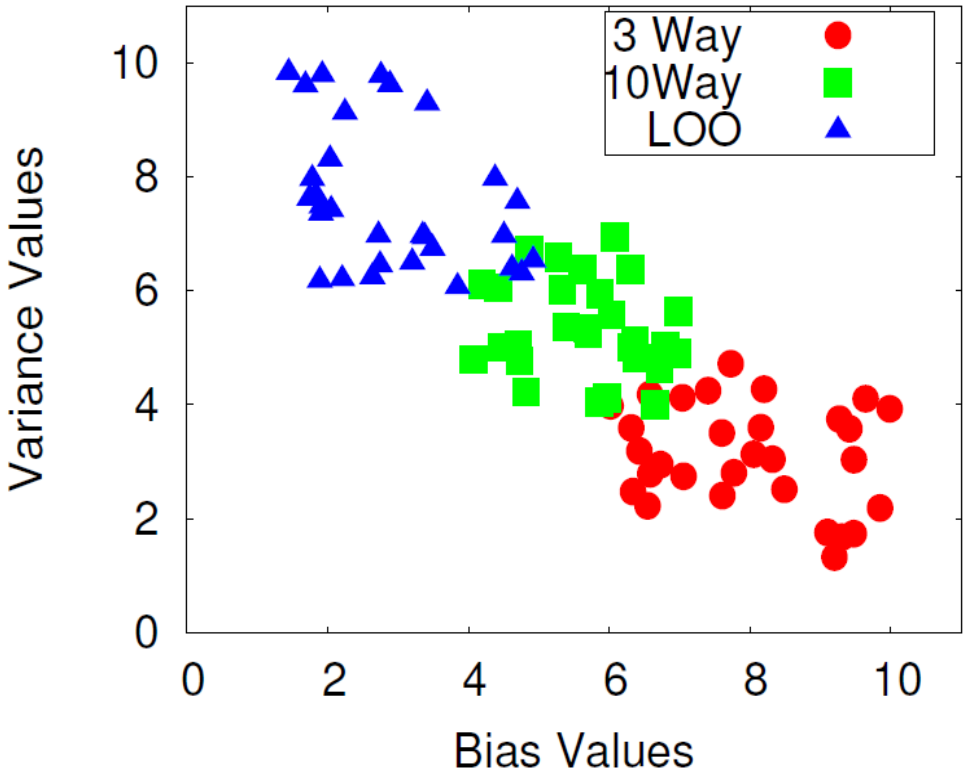
- Chang 1974 – Finding Prototypes for Nearest Neighbor Classifiers
- Kim 2011 – Dealing with Noise in Defect Prediction
- Kocaguneli 2011 – Exploiting the Essential Assumptions of ABE Estimation
- Kocaguneli 2010 – When to use data from other projects for effort estimation
- Experiment: Independent Variable Mutation
- **Experiment: Bias/Variance**

Bias/Variance

- Observations
  - According to theory higher number of smaller test sets, increase the variance and decrease the bias.
  - Extensive study showed that the theory does not hold for effort estimation datasets.
- Conclusion – only few instances matter!

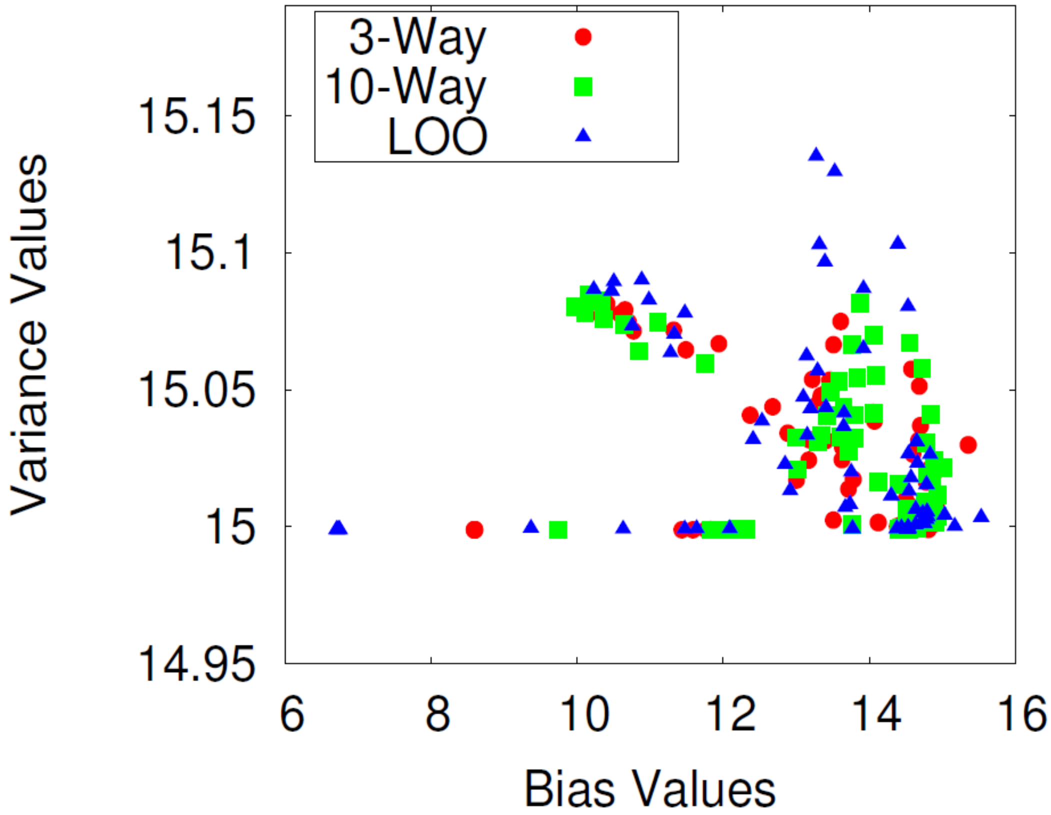
Bias/Variance

- A simple simulation for the “expected” case of B&V relation to testing strategies.



Bias/Variance

- B&V values for cocomo81.



Outline

- Previous research = Less is More
- **Why? - The Answer lies in the E(k) matrix**
- Now - we exploit instance space

Effort Estimation and Active Learning

- Investigation of software effort dataset characteristics
- First application of active learning on software effort estimation
- Active-learning guidance system based on dataset characteristics
  - Reduction in data collection effort

The E(k) Matrix

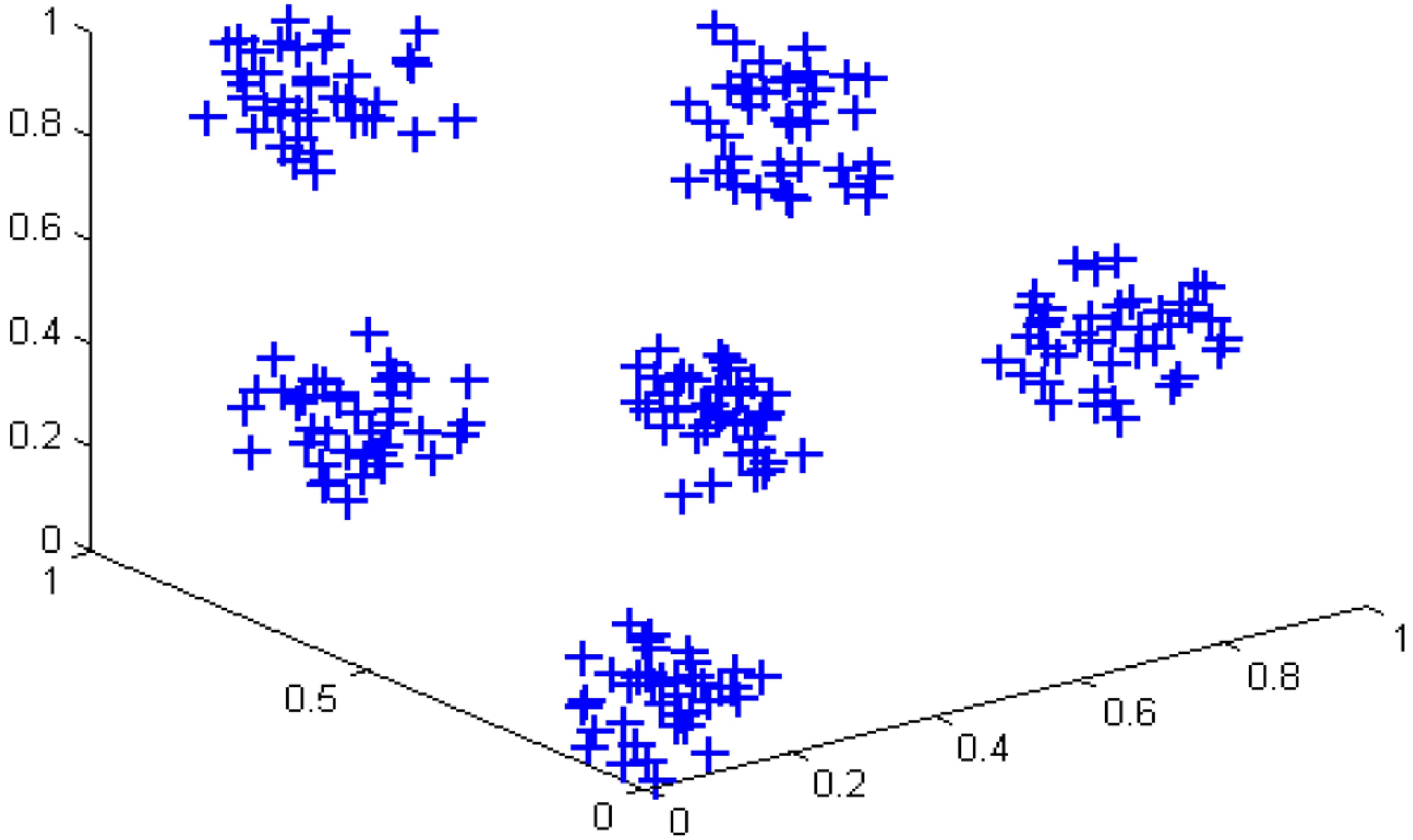
- Everyones k-th nearest matrix
  - The story...

We were interested in the effect of injecting noise to the datasets in the context of ABE.

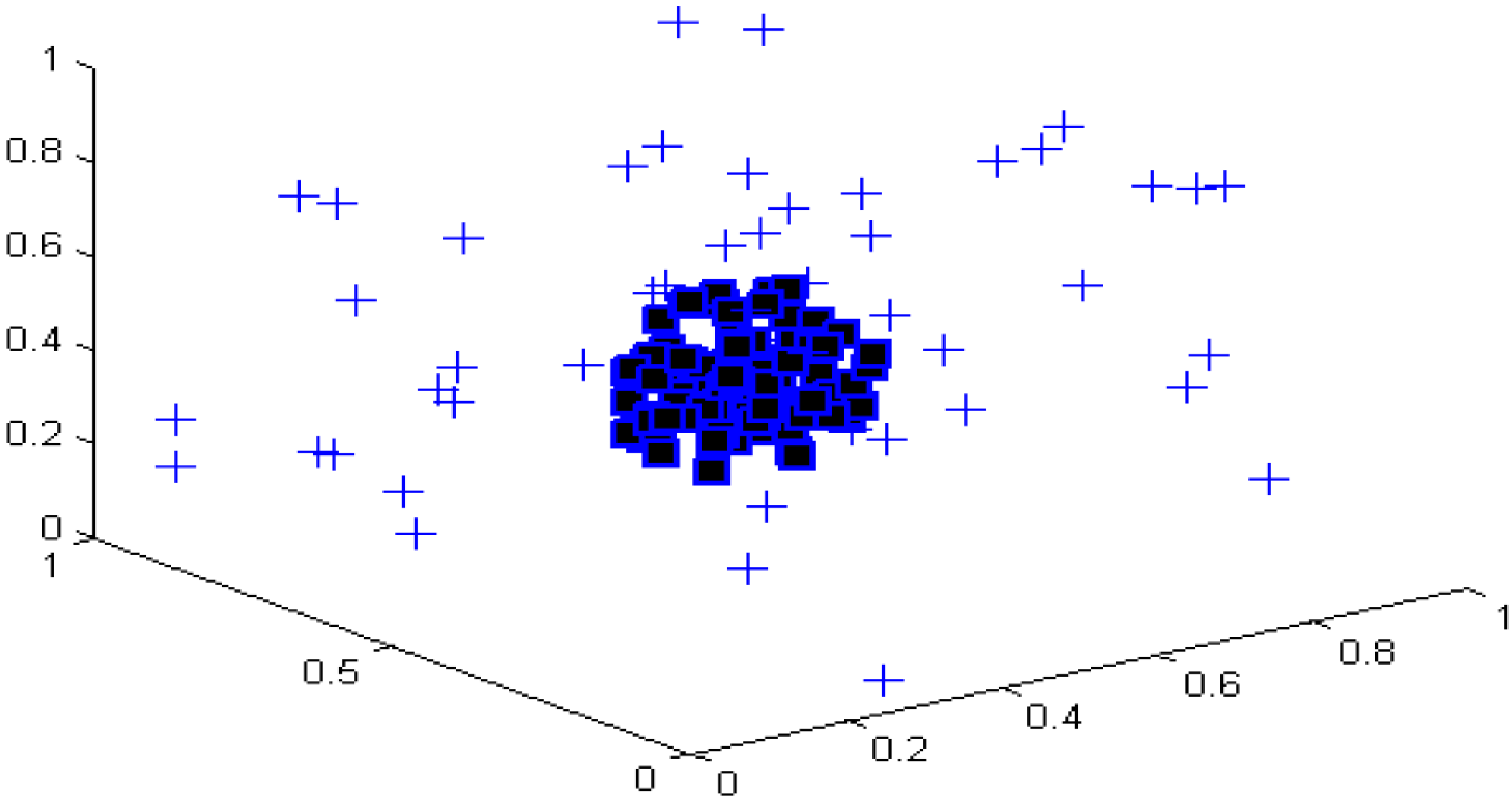
When noise was injected the ABE performances before and after noise injection were statistically the same.

Why? - maybe datasets had a different topology than predicted

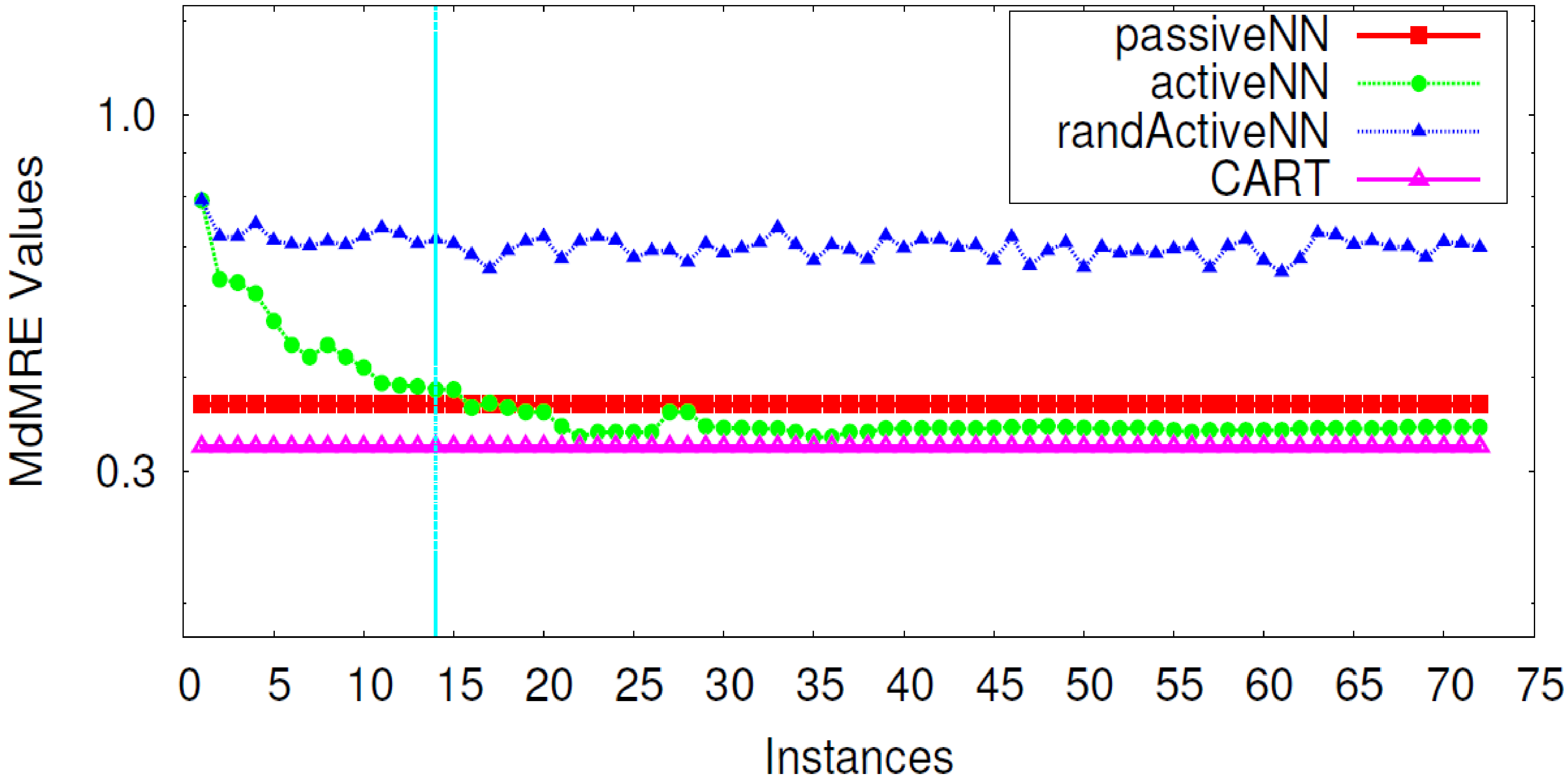
Expected Topology



Actual Topology



Result



passiveNN
activeNN
randActiveNN
CART
MdMRE Values
1.0
0.3
0
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
Instances

Outline

- Previous research = Less is More
- Why? - The Answer lies in the E(k) matrix
- **Now - we exploit instance space**

Exploiting Instance Space

- E(k) and guidance system
  - Find popularity of each instance
  - Use expert to label % of most popular
- CLIFF
  - Select instances based on best ranked attribute values
  - Immunizes against noise

E(k) Matrix and Guidance System

- Simple example
-

Project	KLOC	Effort
P_1	20	3
P_2	10	4
P_3	40	7

E(k) Matrix and Guidance System

- Step 1: Build distance matrix
-

	P_1	P_2	P_3
P_1	0	0.34	0.66
P_2	0.34	0	1
P_3	0.66	1	0

E(k) Matrix and Guidance System

- Step 2: Create E(k) Matrix
-

	P_1	P_2	P_3
P_1	na	1	2
P_2	1	na	2
P_3	1	2	na

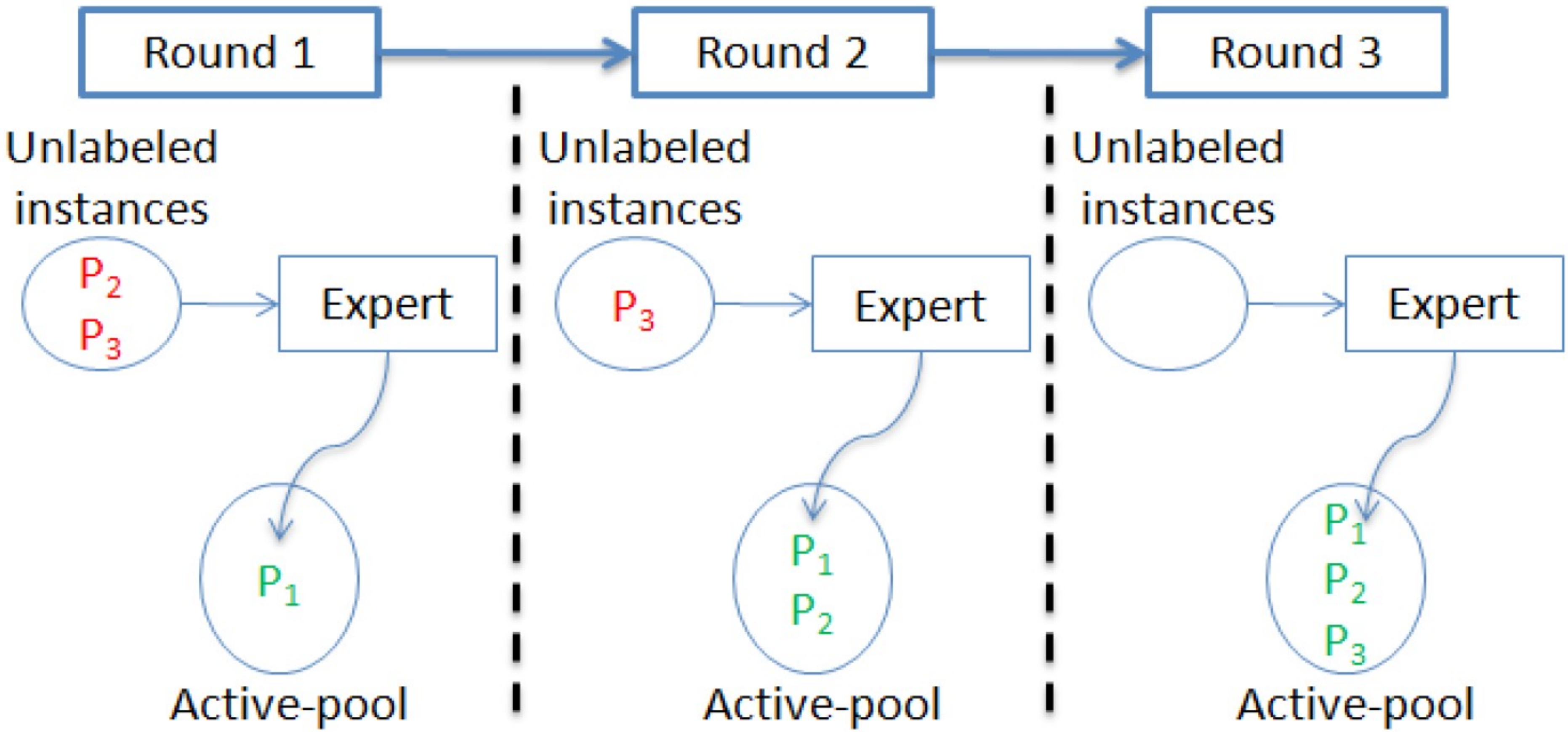
E(k) Matrix and Guidance System

- Step 3: Calculate Popularity Index

$$
\begin{array}{rr|ccc}
 & & P_1 & P_2 & P_3 \\
\hline
 & P_1 & 0 & 1 & 0 \\
 & P_2 & 1 & 0 & 0 \\
+ & P_3 & 1 & 0 & 0 \\
\hline
 & Popularity: & 2 & 1 & 0
\end{array}
$$

E(k) Matrix and Guidance System

- Visualization of Process



Exploiting Instance Space

- E(k) and guidance system
  - Find popularity of each instance
  - Use expert to label % of most popular
- CLIFF
  - Select instances based on best ranked attribute values
  - Immunizes against noise

CLIFF – Immunizes Against Noise

- Simple example
-

#	forecast	temp	humidty	windy	play
1.	sunny	hot	high	FALSE	no
2.	sunny	hot	high	TRUE	no
3.	overcast	hot	high	FALSE	yes
4.	rainy	mild	high	FALSE	yes
5.	rainy	cool	normal	FALSE	yes
6.	rainy	cool	normal	TRUE	no
7.	overcast	cool	normal	TRUE	yes
8.	sunny	mild	high	FALSE	no
9.	sunny	cool	normal	FALSE	yes
10.	rainy	mild	normal	FALSE	yes
11.	sunny	mild	normal	TRUE	yes
12.	overcast	mild	high	TRUE	yes
13.	overcast	hot	normal	FALSE	yes
14.	rainy	mild	high	TRUE	no

CLIFF – Immunizer Against Noise

- Step 1: Get Criteria

{{forecast, rainy} {temp, mild} {humidity, high} {windy, FALSE}}

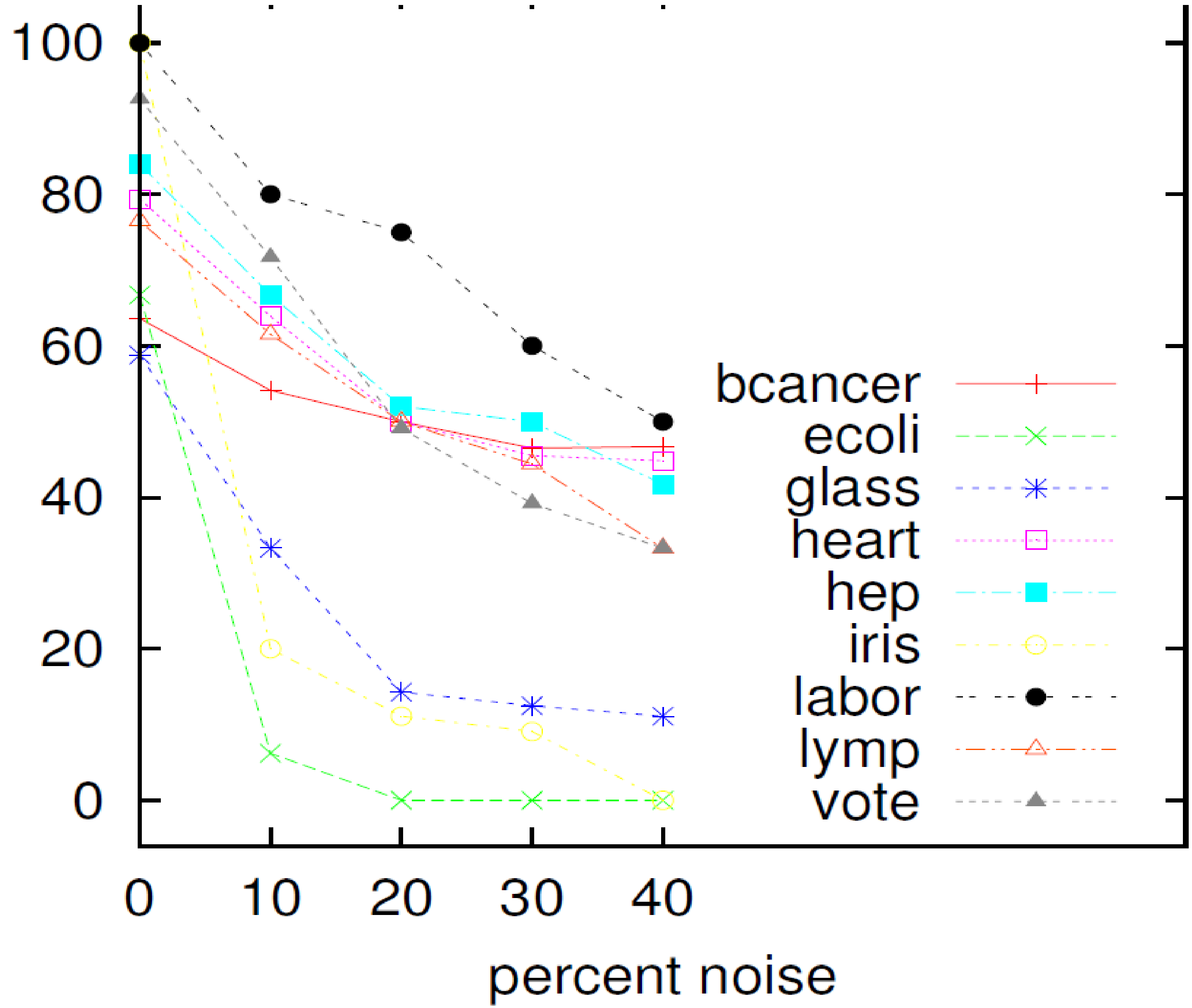
- Step 2: Apply Criteria

4.	rainy	mild	high	FALSE	yes
5.	rainy	cool	normal	FALSE	yes
10.	rainy	mild	normal	FALSE	yes

4.	rainy	mild	high	FALSE	yes
10.	rainy	mild	normal	FALSE	yes

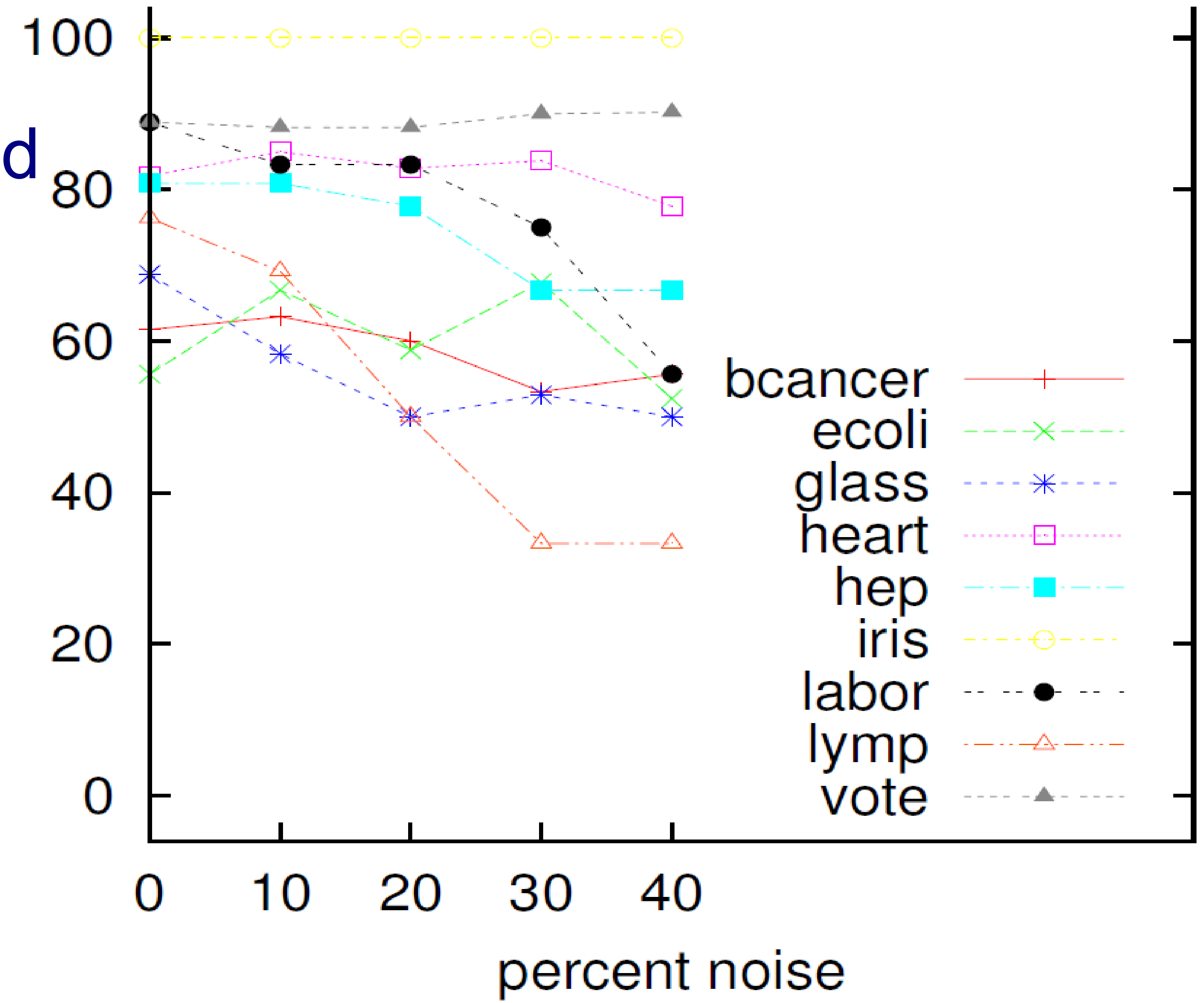
CLIFF vs KNN

- KNN pd



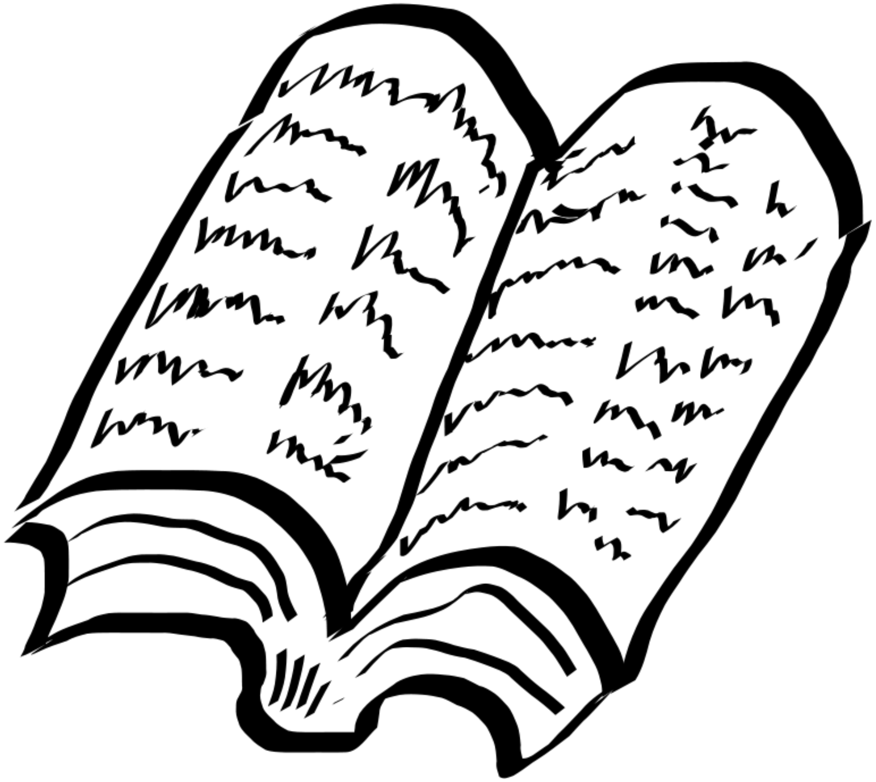
CLIFF vs KNN

- CLIFF pd



Conclusions

- Since few instances matter...
  - Instead of adding to the list of algorithms



Let's pay attention to the data

Questions?

References

- Slide 4

(Chang 1974)
Chang, C L. “Finding Prototypes for Nearest Neighbor Classifiers.” IEEE Trans on Computers C.11 (1974) : 1179-1185.

(Kim 2011)
Kim, S., Zhang, H., Wu, R., & Gong, L. (2011). Dealing with Noise in Defect Prediction. Changes.

(Kocaguneli 2011)
Ekrem Kocaguneli, Tim Menzies, Ayse Bener, Jacky W. Keung, "Exploiting the Essential Assumptions of Analogy-Based Effort Estimation," IEEE Transactions on Software Engineering, 02 Mar. 2011. IEEE computer Society Digital Library. IEEE Computer Society, < http://doi.ieeecomputersociety.org/10.1109/TSE.2011.27>

(Kocaguneli 2010)
Ekrem Kocaguneli, Gregory Gay, Tim Menzies, Ye Yang, and Jacky W. Keung. 2010. When to use data from other projects for effort estimation. In Proceedings of the IEEE/ACM international conference on Automated software engineering (ASE '10). ACM, New York, NY, USA, 321-324. DOI=10.1145/1858996.1859061 http://doi.acm.org/10.1145/1858996.1859061