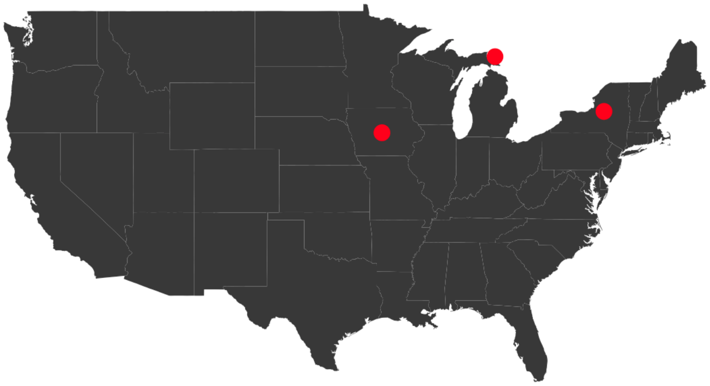
A data driven stopping criterion for evolutionary instance selection

Walter Bennette

September 06, 2016

About me

- **2014 - Present:** Air Force Research Laboratory Information Directorate
- **2009 - 2014:** Iowa State University, Industrial Engineering MS and PhD
- **2005 - 2009:** Lake Superior State University, Mathematics BS



Instance selection

What

- A pre-processing technique for instance-based classification
- Only "necessary" instances are maintained

Why

- Memory
- Prediction time

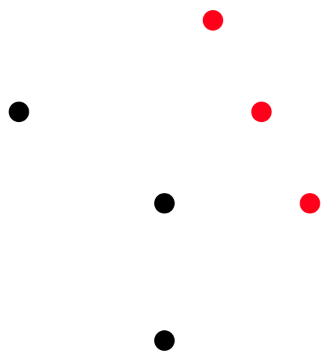
How

- Filters
- Wrappers
  - An evolutionary algorithm with an arbitrary stopping criterion

Instance-based classification

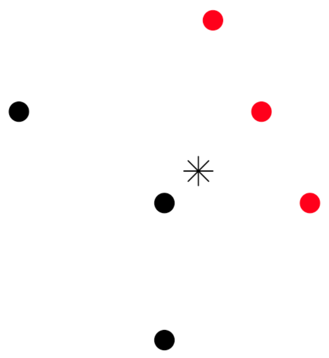
Instance-based classification

Given this data...



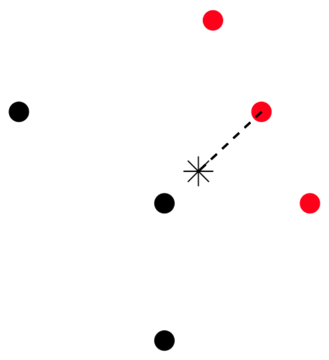
Instance-based classification

What would we label a new point?



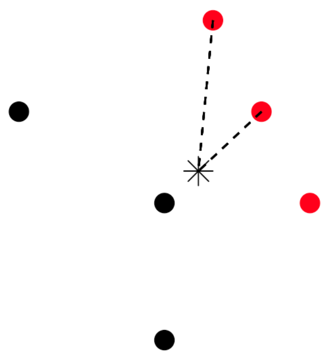
Instance-based classification

It should be the same as its closest neighbors.



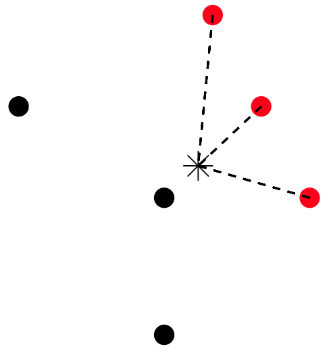
Instance-based classification

It should be the same as its closest neighbors.



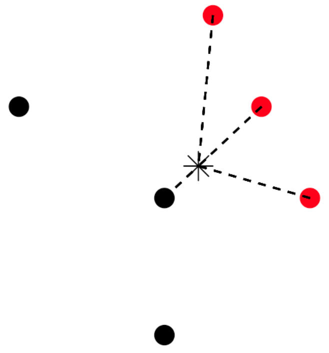
Instance-based classification

It should be the same as its closest neighbors.



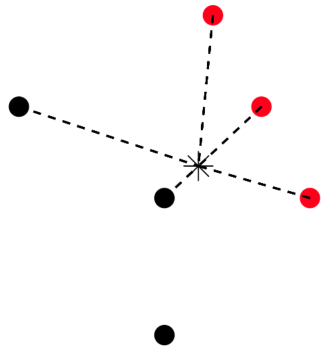
Instance-based classification

It should be the same as its closest neighbors.



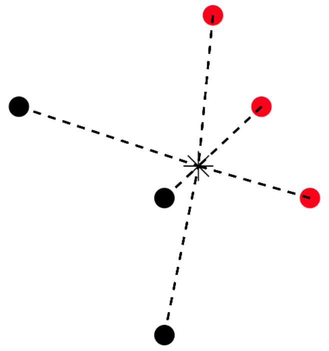
Instance-based classification

It should be the same as its closest neighbors.

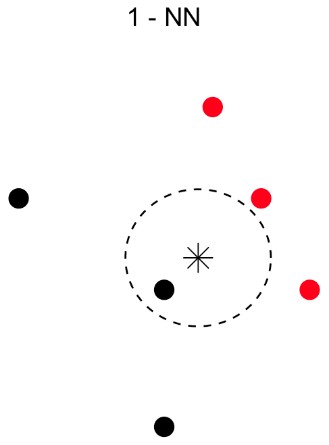


Instance-based classification

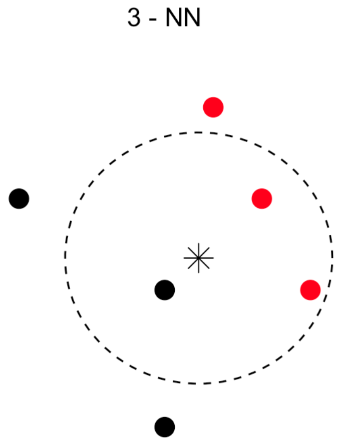
It should be the same as its closest neighbors.



Instance-based classification



Instance-based classification

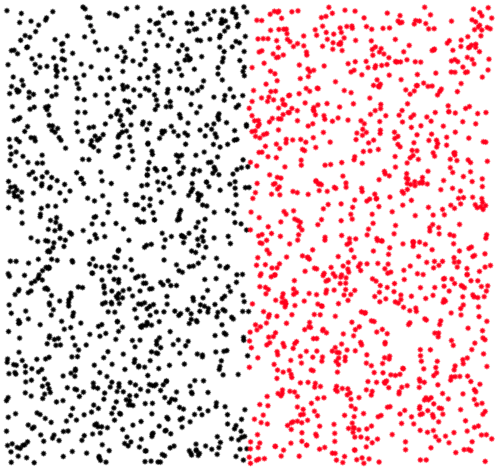


Instance-based classification

- Load forecasting assistant for power company
    - Hourly load forecast
    - Utilize weather and seasonal variables
    - Growing number of data sources and observations
    - Increased control

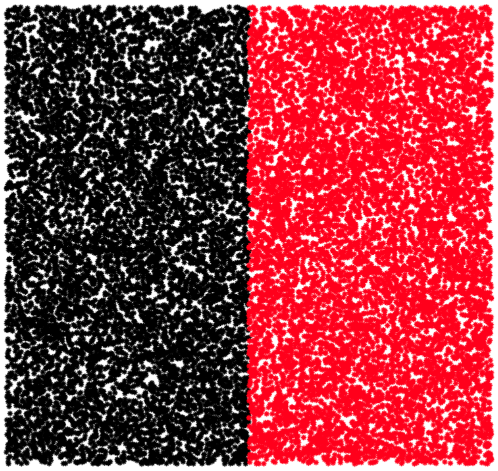
Instance-based classification

What if there is a large amount of data?



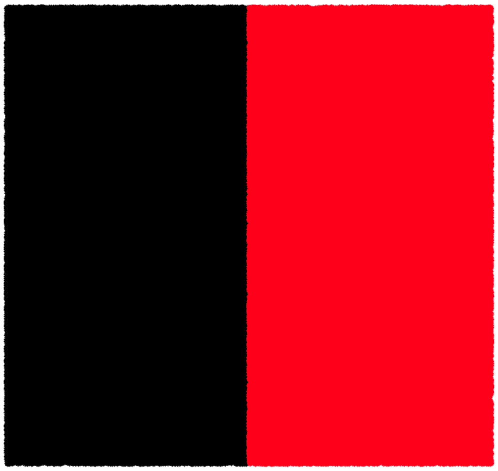
Instance-based classification

What if there is a huge amount of data?



Instance-based classification

What if there is a serious amount of data?



Instance selection

Instance selection

Retain only the instances "necessary" to achieve adequate classification rates

- Reduce storage requirements
- Reduce prediction time

Edited Nearest Neighbors (ENN)

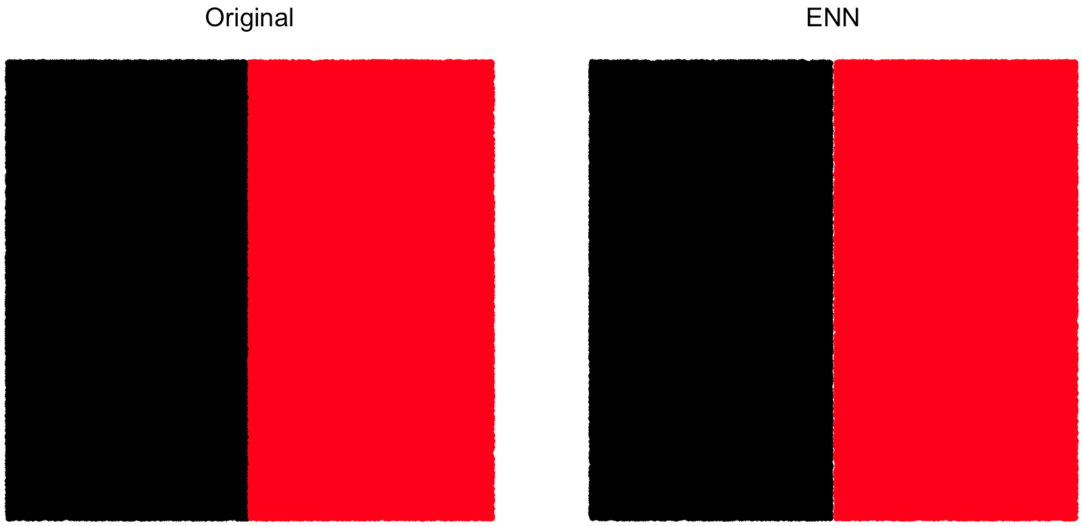
Formulation:

- An instance is removed from the training data if its does not agree with the majority of its k nearest neighbors

Effect:

- Makes decision boundaries smoother
- Doesn't remove much data

Edited Neares Neighbors (ENN)



DROP3

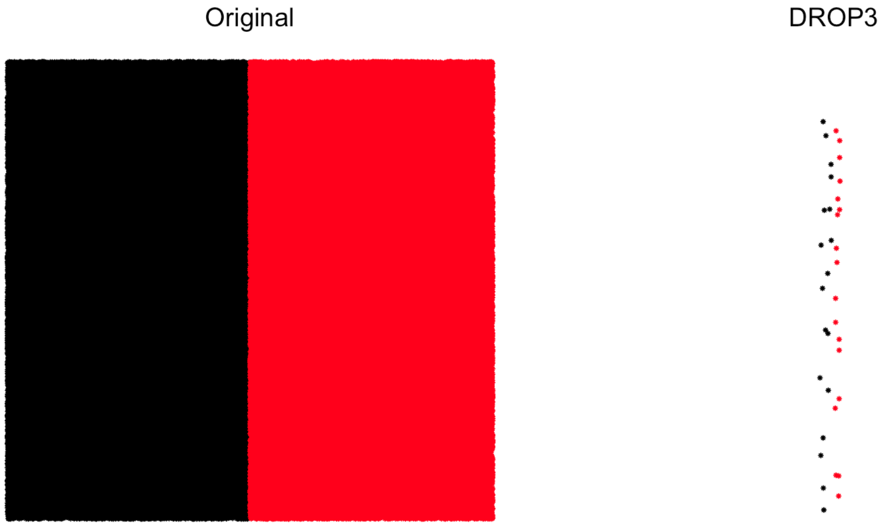
Formulation:

- Iterative procedure that compares accuracy of neighborhoods with and without members

Effect:

- Removes much more data than ENN
- Maintains acceptable accuracy

DROP3



Original

DROP3

Evolutionary Instance Selection

Evolutionary Instance Selection

- Search for best subset of training data
- $Fitness = \alpha * classAccuracy + (1 - \alpha) * percReduction$
- Each instance is a gene
  - One, keep instance
  - Zero, discard instance

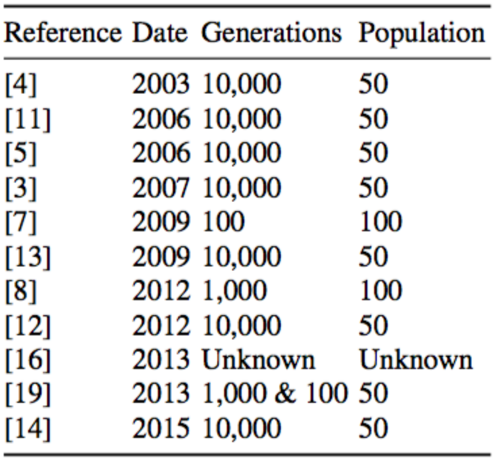
Evolutionary Instance Selection

- Cano, Herrera, and Lozano (2003), tested families of evolutionary algorithms
- Determined "cross generational elitist selection, heterogeneous recombination and cataclysmic mutation" (CHC) was most effective
- Widely adapted in instance selection literature
- Some of the best results for data reduction and classification accuracy (García, Luengo, and Herrera 2015)

CHC

1. Create a parent population of size N and set threshold to $\frac{|training\ data|}{4}$
2. Generate a child population from parents
   - Select two previously unconsidered parents
   - If Hamming distance is greater than threshold perform half uniform crossover (HUX) to generate two children
3. Hold a competition to determine new parent population
   - If no children enter parent population reduce threshold by one
   - If threshold falls below zero perform cataclysmic re-population
     - Reset threshold and discard all chromosomes except most fit
     - Use mutation to generate $N - 1$ new chromosomes
4. Return to Step 2 until stopping criterion is met

Current CHC stopping critierion

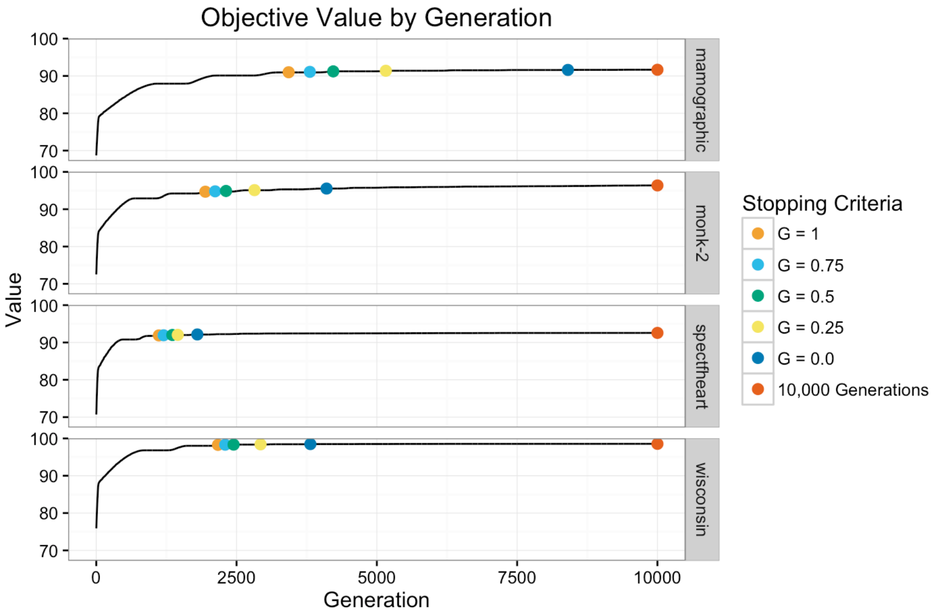


Reference	Date	Generations	Population
[4]	2003	10,000	50
[11]	2006	10,000	50
[5]	2006	10,000	50
[3]	2007	10,000	50
[7]	2009	100	100
[13]	2009	10,000	50
[8]	2012	1,000	100
[12]	2012	10,000	50
[16]	2013	Unknown	Unknown
[19]	2013	1,000 & 100	50
[14]	2015	10,000	50

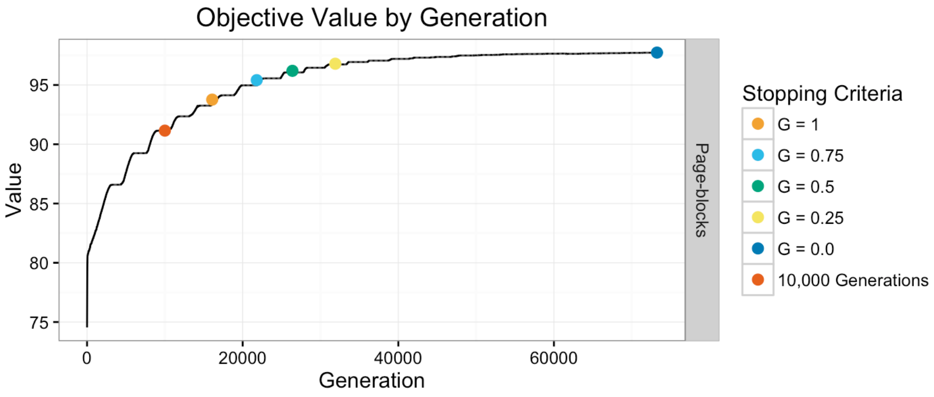
Data driven CHC stopping critierion

- At a cataclysmic re-population, compare the best individual to the best individual from the last cataclysmic re-population
- If the improvement in fitness is less than or equal to some G, stop

Data driven example

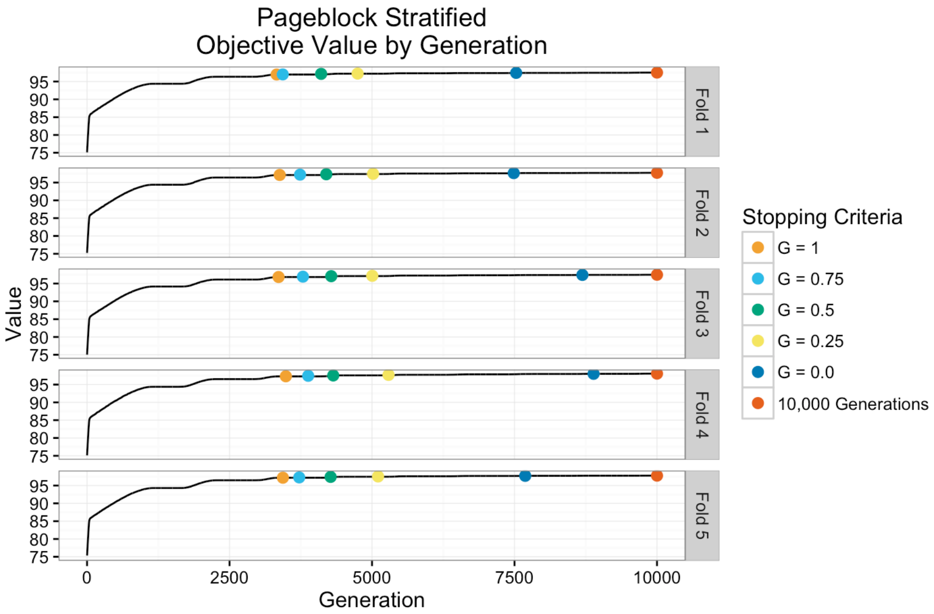


Data driven example



- CHC has difficulty converging when there are many instances
- Most recommend a stratified scaling approach

Data driven example



Experiment

Using 3-NN:

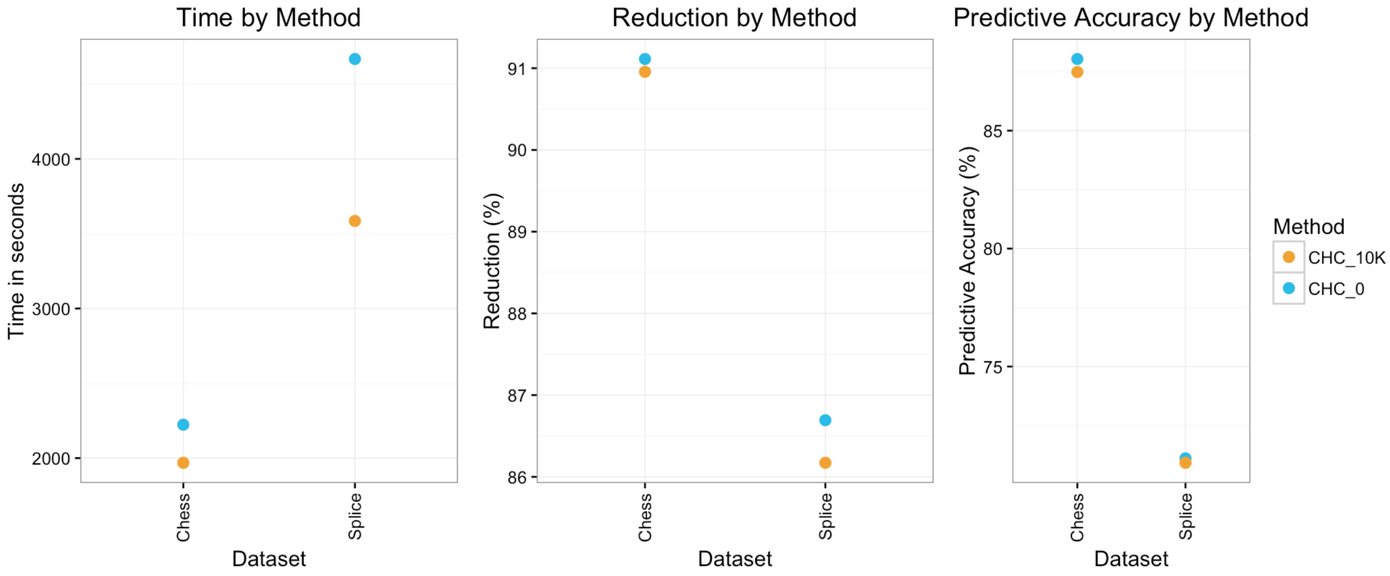
- Compare CHC_10K and CHC_0 (10k generations versus data driven for $G = 0$)
- 10-fold cross validation applied three times
- Record accuracy, reduction, and computation time
- 30 "small" datasets (100 - 1,000 instances)
- 21 "medium" datasets, using stratification (1,001 - 12,960 instances)
- Wilcoxin Signed Ranked test for differences in accuracy, reduction, and time

Results

Size	Method	Accuracy	Reduction	Time
Small	CHC_10K	77.3	91.1	119
Small	CHC_0	77.3	90.6	64
Medium	CHC_10K	75.4	90.9	1631
Medium	CHC_0	75.6	90.8	1415

- No significant difference in accuracy
- Significant (but small) difference in reduction
- Significant difference in time

Unexpected results



Take away one

- A set number of generations is not the correct way to terminate CHC
- CHC_0 is a criterion
- Additional criterion can be created

A word on the competittion

Size	Method	Accuracy	Reduction	Time
Small	3-NN	78.6	NA	NA
Small	DROP3	76.1	90.7	1
Small	CHC_0	77.3	90.6	64
Medium	3-NN	78.7	NA	NA
Medium	DROP3	73.7	92.8	17
Medium	CHC_0	75.6	90.8	1415

- On average, DROP3 achieves greater reduction
- On average, CHC_0 achieves better accuracy
- DROP3 is very fast

Take away two

- Practitioners need to keep their application in mind
- Use DROP3 when instance selection needs to be applied quickly
- Use CHC when accuracy is a priority

Questions

Walter Bennette
walter.bennette.1@us.af.mil
wdbennette@gmail.com

References

Cano, J.R., F. Herrera, and M. Lozano. 2003. “Using evolutionary algorithms as instance selection for data reduction in KDD: an experimental study.” 7 (6): 561–75.

García, Salvador, Julian Luengo, and Francisco Herrera. 2015. . Vol. 72. Intelligent Systems Reference Library. Cham: Springer International Publishing. http://link.springer.com/10.1007/978-3-319-10247-4 http://www.scopus.com/inward/record.url?eid=2-s2.0-84906871736{\&}partnerID=tZOtx3y1.