Chapter 8. Classification: Ensembles

Meng Jiang

CSE 40647/60647 Data Science Fall 2017

Introduction to Data Mining

Turing Award Recipients

…	…	…
1998	Jim Gray	For seminal contributions to **database and transaction processing research**…
1999	Frederick P. Brooks, Jr.	For landmark contributions to **computer architecture, operating systems, and software engineering**…
2000	Andrew Chi-Chih Yao	**Theory of computation**, pseudorandom number generation, cryptography, and communication complexity…
2001	Ole-Johan Dahl Kristen Nygaard	**Object-oriented programming**, Simula I and Simula 67…
2002	Ronald L. Rivest, Adi Shamir, Leonard M. Adleman	For their ingenious contribution for making **public-key cryptography** useful in practice.
2003	Alan Kay	Contemporary **object-oriented programming languages**…
2004	Vinton G. Cerf Robert E. Kahn	**Internetworking**, including the design and implementation of the Internet's basic communications protocols, TCP/IP…
2005	Peter Naur	**Programming language design**, ALGOL 60, compiler design…
2006	Frances E. Allen	**Optimizing compiler techniques**…

Turing Award Recipients (cont.)

2007	Edmund M. Clarke, E. Allen Emerson and Joseph Sifakis	For their roles in developing model checking into a **highly effective verification technology**, widely adopted in the hardware and software industries…
2008	Barbara Liskov	**Programming language and system design**…
2009	Charles P. Thacker	**Xerox Alto, the 1st modern PC, Ethernet and Tablet PC**….
2010	Leslie G. Valiant	Theory of computation…
2011	Judea Pearl	**Artificial intelligence** through the development of a calculus for **probabilistic and causal reasoning**…
2012	Silvio Micali Shafi Goldwasser	Transformative work that laid the complexity-theoretic foundations for the science of **cryptography**…
2013	Leslie Lamport	Contributed to **distributed and concurrent systems**…
2014	Michael Stonebraker	Concepts underlying **modern database systems**…
2015	Martin E. Hellman Whitfield Diffie	Introduced **public-key cryptography**, the foundation for the most regularly-used security protocols on the Internet…
2016	Tim Berners-Lee	Invented the **World Wide Web** and the **first web browser**…

Marvin Minsky

- Marvin Lee Minsky (August 9, 1927 – January 24, 2016) was an American cognitive scientist concerned largely with research of **artificial intelligence** (AI), co-founder of the Massachusetts Institute of Technology's AI laboratory, and author of several texts concerning AI and philosophy.
- Awards
  - **Turing Award (1969)**
  - Japan Prize (1990)
  - IJCAI Award for Research Excellence (1991)
  - Benjamin Franklin Medal (2001)
  - Computer History Museum Fellow (2006)
  - BBVA Foundation Frontiers of Knowledge Award (2013)

Logical vs. Analogical
or
Symbolic vs. Connectionist
or
Neat vs. Scruffy

https://web.media.mit.edu/~minsky/papers/SymbolicVs.Connectionist.html

Marvin Minsky

In Artificial Intelligence at MIT, Expanding Frontiers, Patrick H. Winston (Ed.), Vol.1, MIT Press, 1990. Reprinted in AI Magazine, Summer 1991.

Ensembles

"To solve really hard problems, we'll have to use *several different representations*…

It is time to *stop arguing* over *which* type of pattern-classification technique *is best*…

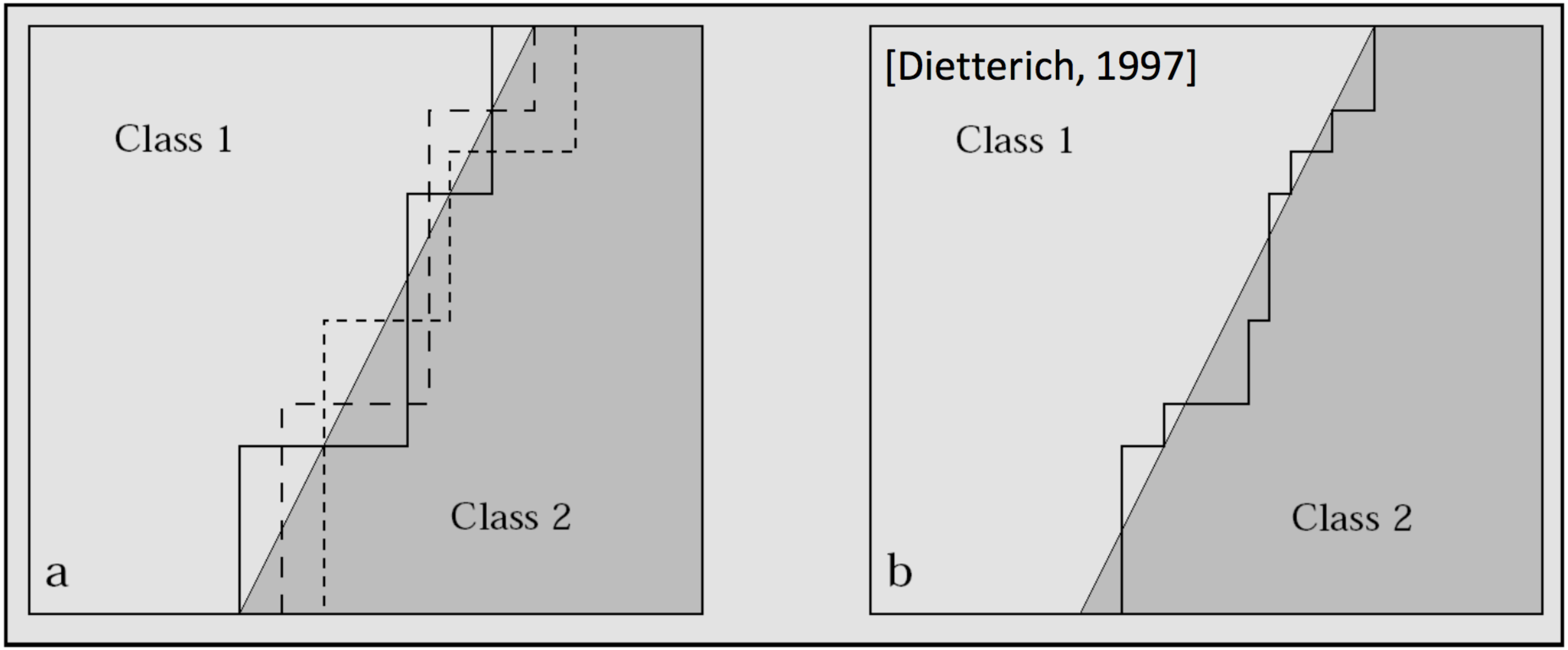
Instead we should work at a higher level of organization and discover how to build *managerial systems* to exploit the *different virtues* and evade the *different limitations* of each of these ways of comparing things." [**Minsky, 1991**]

Ensembles (cont.)

- An ensemble is a set of classifiers that learn a target function, and their *individual predictions are combined* (weighted or unweighted) to classify new examples
  - Each classifier should be *more accurate than by chance*, and independent of one another.
  - Usually *more accurate than a single classifier*.
- Ensembles generally improve the *generalization performance* of a set of classifiers on a domain.

Ensemble Methods

- Ensemble methods
  - Use a *combination of models* to ***increase accuracy***
  - Combine a series of *k* learned models, M_1, M_2, …, M_k, with the aim of creating an *improved model* M*



Ensemble Methods (cont.)

- Popular ensemble methods
  - **Bagging:** averaging the prediction over a collection of classifiers
  - **Boosting:** weighted vote with a collection of classifiers
    - **AdaBoost (Adaptive Boosting):** adaptive in the sense that subsequent weak learners are tweaked in favor of those instances misclassified by previous classifiers.

Bagging

- Training
  - Given a data set D of *d* instances, a classifier model M_i is learned for a training set D_i of *d* instances that is *sampled with replacement* from D (i = 1…k)
  - As a result of the *sampling-with-replacement* procedure, each classifier is trained on approximately *63.2%* of the training examples
  - For a dataset with *d* instances, each instance has a probability of $1 - (1 - 1/d)^d$ of being selected at least once in the *d* samples.
    - For $d \rightarrow \infty$, this number converges to $(1 - 1/e)$ or 0.632 [Bauer and Kohavi, 1999]

Bagging (cont.)

- Classification: classify an unknown sample **X**
  - Each classifier M_i returns its class prediction
  - The bagged classifier M* *counts the votes* and assigns the class with the *most votes* to **X**
- Accuracy: Proved improved accuracy in prediction
  - *Often significantly better* than a single classifier derived from D
- *Example:* (Course Project) Entity type recognition
  - Features: Triggers in contextual words
  - Each context setting as a classifier: 1/2/3 contextual words. The probability that the technical term is a "method" ("problem", "dataset", "metric", etc.).
  - Majority voting

  http://www.meng-jiang.com/teaching/TypingDemo.zip

Boosting

- Training
  - *Weights* are assigned to each training instance
  - A series of k classifiers is *iteratively* learned
  - After a classifier M_i is learned, the *weights* are updated to allow the subsequent classifier, M_{i+1}, to pay more attention to the *training* instances that were *misclassified* by M_i
- Classification
  - The final M* *combines the votes* of each individual classifier, where the *weight* of each classifier's vote is a function of its *accuracy* on classifying training instances
- Comparing with Bagging: Boosting tends to have *greater accuracy*, but it risks *overfitting* the model to misclassified data

AdaBoost (Adaptive Boosting)

- Given a set of *d* class-labeled instances, $(\mathbf{X_1}, y_1), \ldots, (\mathbf{X_d}, y_d)$
- Initially, all the *weights* of instances are set the same (1/*d*)
- Generate *k* classifiers in *k* rounds. At round *i*,
  - Instances from D are *sampled with replacement* to form a training set D_i of the same size
  - Each instance's chance of being selected is based on its *weight*
  - A classification model M_i is derived from D_i
  - Its *error rate* is calculated using D_i as a "test set"
  - If an instance is misclassified, its *weight* is increased, otherwise it is decreased

AdaBoost (cont.)

- *Error rate: err($\mathbf{X}_j$)* is the misclassification error of instance $\mathbf{X}_j$. Classifier M_i's error rate is the sum of the weights of the misclassified instances:

$$error(M_i) = \sum_{j}^{d} w_j \times err(\mathbf{X_j})$$

- The *weight* of classifier M_i's vote is

$$\log \frac{1 - error(M_i)}{error(M_i)}$$

References

- C. Apte and S. Weiss. Data mining with decision trees and decision rules. Future Generation Computer Systems, 13, 1997
- P. K. Chan and S. J. Stolfo. Learning arbiter and combiner trees from partitioned data for scaling machine learning. KDD'95
- A. J. Dobson. An Introduction to Generalized Linear Models. Chapman & Hall, 1990.
- R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification, 2ed. John Wiley, 2001
- U. M. Fayyad. Branching on attribute values in decision tree generation. AAAI'94.
- Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. J. Computer and System Sciences, 1997.
- J. Gehrke, R. Ramakrishnan, and V. Ganti. Rainforest: A framework for fast decision tree construction of large datasets. VLDB'98.
- J. Gehrke, V. Gant, R. Ramakrishnan, and W.-Y. Loh, BOAT -- Optimistic Decision Tree Construction. SIGMOD'99.
- T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag, 2001.
- T.-S. Lim, W.-Y. Loh, and Y.-S. Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. Machine Learning, 2000

References (cont.)

- J. Magidson. The Chaid approach to segmentation modeling: Chi-squared automatic interaction detection. In R. P. Bagozzi, editor, Advanced Methods of Marketing Research, Blackwell Business, 1994
- M. Mehta, R. Agrawal, and J. Rissanen. SLIQ : A fast scalable classifier for data mining. EDBT'96
- T. M. Mitchell. Machine Learning. McGraw Hill, 1997
- S. K. Murthy, Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey, Data Mining and Knowledge Discovery 2(4): 345-389, 1998
- J. R. Quinlan. Induction of decision trees. Machine Learning, 1:81-106, 1986.
- J. R. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.
- J. R. Quinlan. Bagging, boosting, and c4.5. AAAI'96.
- R. Rastogi and K. Shim. **Public: A decision tree classifier that integrates building and pruning**. VLDB'98
- J. Shafer, R. Agrawal, and M. Mehta. **SPRINT : A scalable parallel classifier for data mining**. VLDB'96
- J. W. Shavlik and T. G. Dietterich. **Readings in Machine Learning**. Morgan Kaufmann, 1990
- P. Tan, M. Steinbach, and V. Kumar. **Introduction to Data Mining**. Addison Wesley, 2005
- S. M. Weiss and C. A. Kulikowski. **Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems**. Morgan Kaufman, 1991
- S. M. Weiss and N. Indurkhya. **Predictive Data Mining**. Morgan Kaufmann, 1997
- I. H. Witten and E. Frank. **Data Mining: Practical Machine Learning Tools and Techniques**, 2ed. Morgan Kaufmann, 2005