Inspire…Educate…Transform.

Advanced Machine Learning

Boosting and Stacking

Manish Gupta

Slide adapted from
Jing Gao’s SDM 2010 tutorial “On the Power of Ensemble: Supervised and Unsupervised Methods Reconciled”
https://project.dke.maastrichtuniversity.nl/datamining/2013-Slides/lecture-07.ppt

Outline

- Why ensemble learning?
- Supervised learning
  - Methods for Independently Constructing Ensembles
    - Majority Vote, Bagging and Random Forest
  - **Methods for Coordinated Construction of Ensembles**
    - **Boosting, Stacking**

Methods for Coordinated Construction of Ensembles

The key idea is to learn complementary classifiers so that instance classification is realized by taking an weighted sum of the classifiers. This idea is used in two methods:

- Boosting
- Stacking.

Boosting

- Also uses voting/averaging but models are weighted according to their performance
- Iterative procedure: new models are influenced by performance of previously built ones
  - New model is encouraged to become expert for instances classified incorrectly by earlier models
  - Intuitive justification: models should be experts that complement each other

Boosting Example

- Example
  – Record 4 is hard to classify
  – Its weight is increased, therefore it is more likely to be chosen again in subsequent rounds

Original Data	1	2	3	4	5	6	7	8	9	10
Boosting (Round 1)	7	3	2	8	7	9	4	10	6	3
Boosting (Round 2)	5	4	9	4	2	5	1	7	4	2
Boosting (Round 3)	4	4	8	10	4	5	4	6	3	4

AdaBoost.M1

classifier generation

```
Assign equal weight to each training instance.
For each of t iterations:
  Learn a classifier from weighted dataset.
  Compute error e of classifier on weighted dataset.
  If e equal to zero, or e greater or equal to 0.5:
    Terminate classifier generation.
  For each instance in dataset:
    If instance classified correctly by classifier:
       Multiply weight of instance by e / (1 - e).
  Normalize weight of all instances.
```

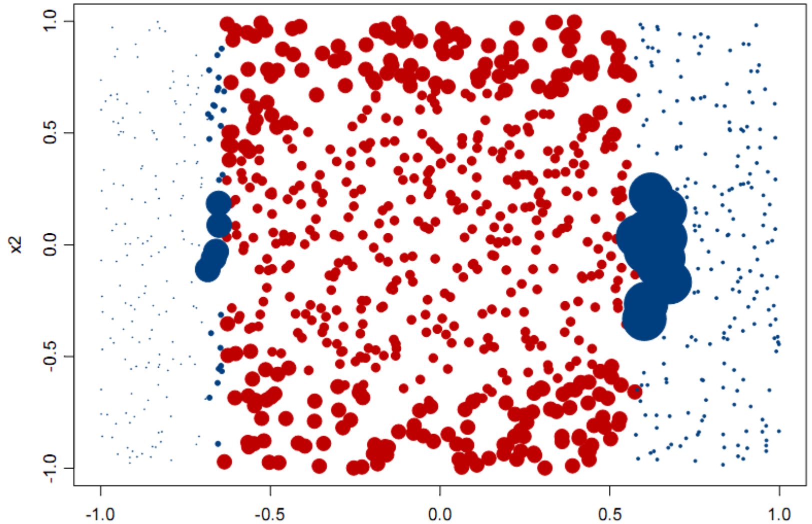
classification

```
Assign weight of zero to all classes.
For each of the t classifiers:
  Add -log(e / (1 - e)) to weight of class predicted
  by the classifier.
Return class with highest weight.
```

CSE 7305c

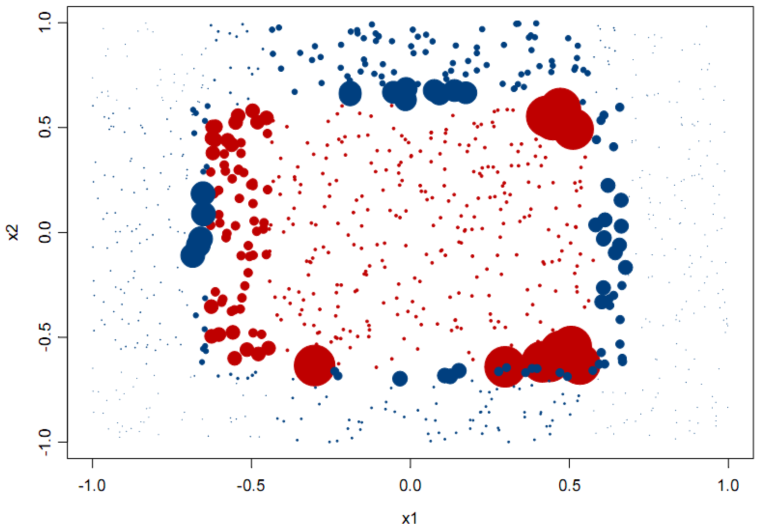
Remarks on Boosting

- Boosting can be applied without weights using re-sampling with probability determined by weights;
- Boosting decreases exponentially the training error in the number of iterations;
- Boosting works well if base classifiers are not too complex and their error doesn't become too large too quickly!
- Boosting reduces the bias component of the error of simple classifiers!

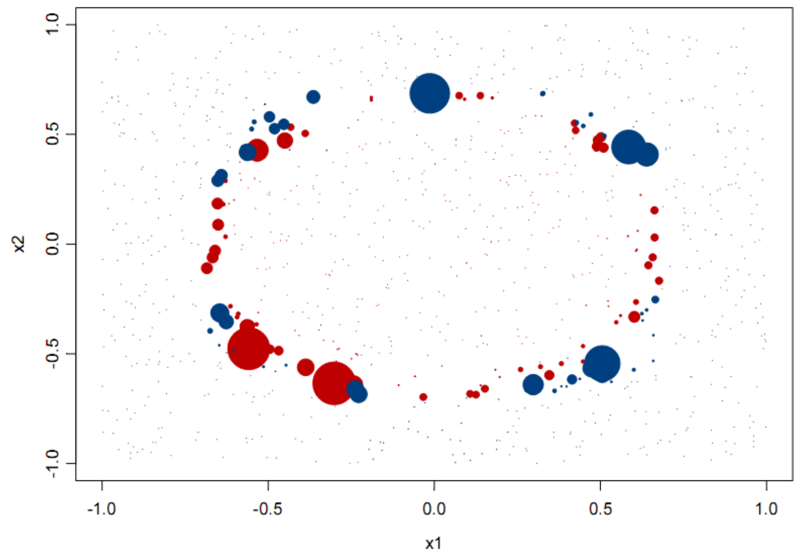


Classifications (colors) and Weights (size) after *1 iteration* Of AdaBoost

3 iterations

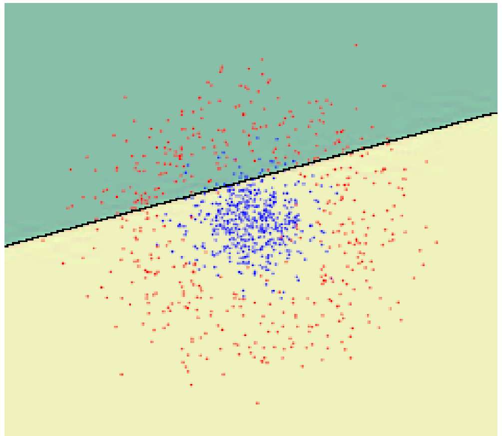


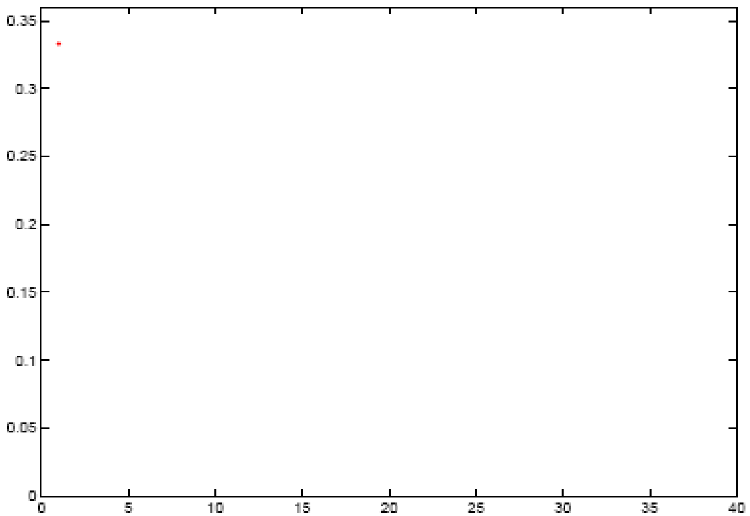
20 iterations



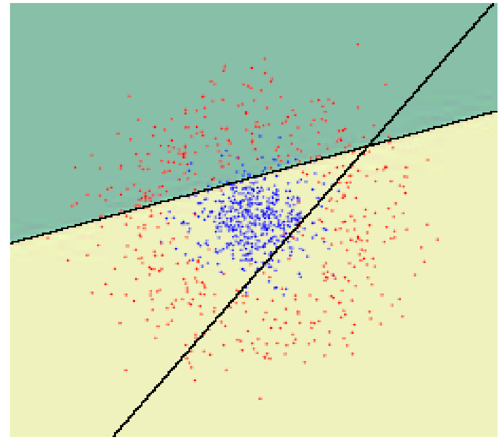
from Elder, John. From Trees to Forests and Rule Sets - A Unified Overview of Ensemble Methods. 2007.

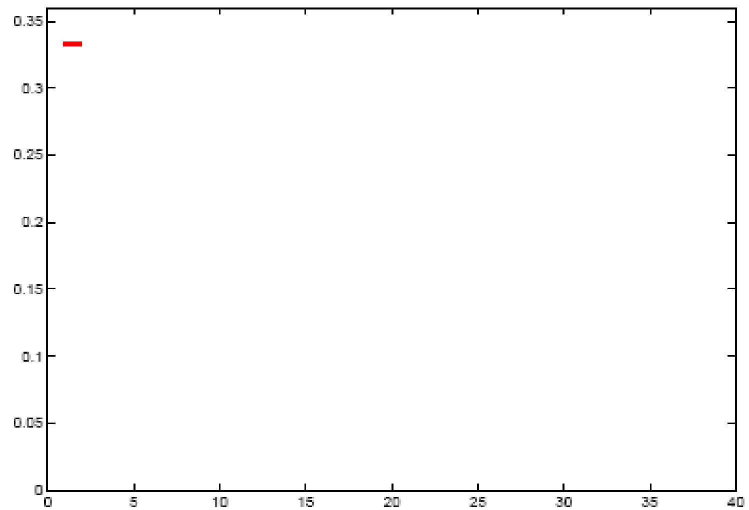
$t = 1$



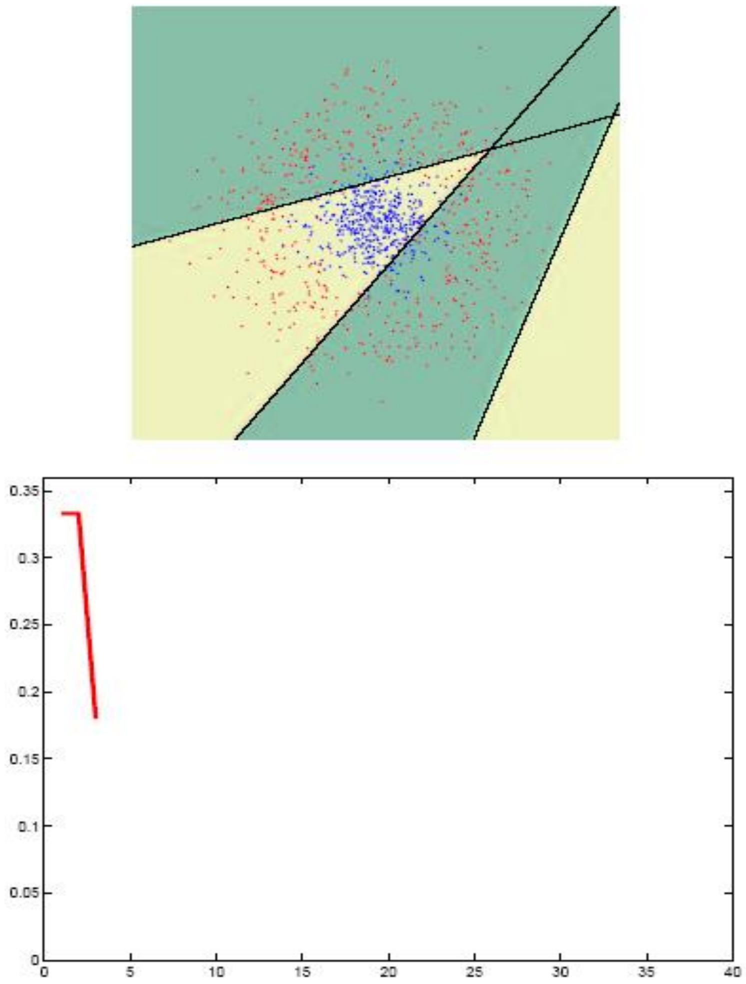

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0 5 10 15 20 25 30 35 40

$t = 2$

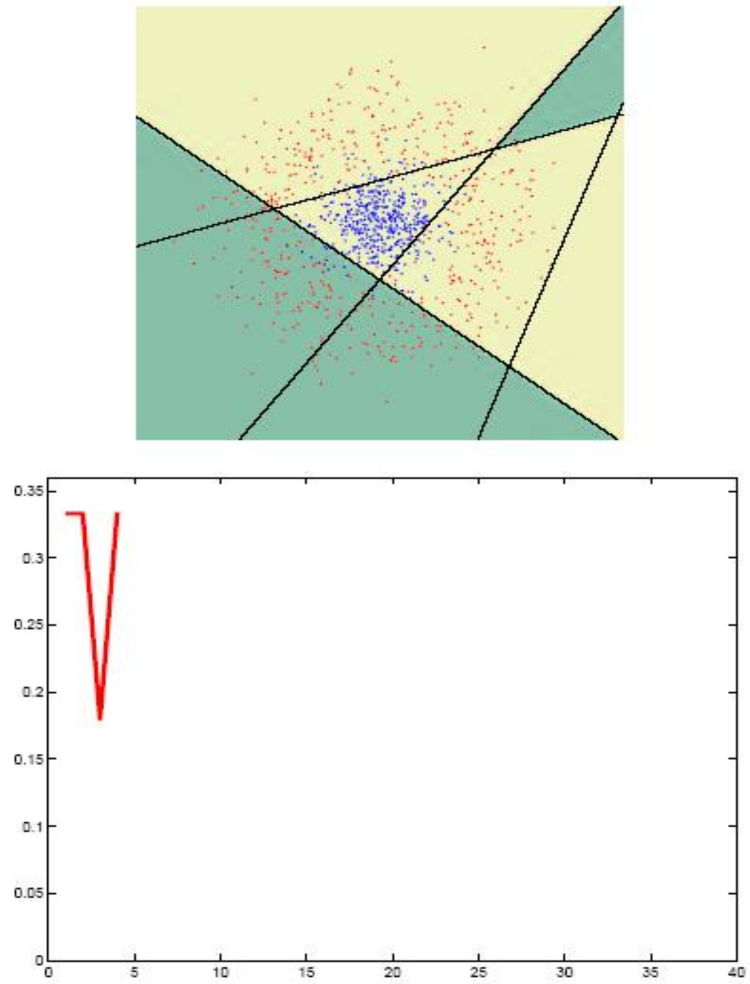



0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0 5 10 15 20 25 30 35 40

$t = 3$

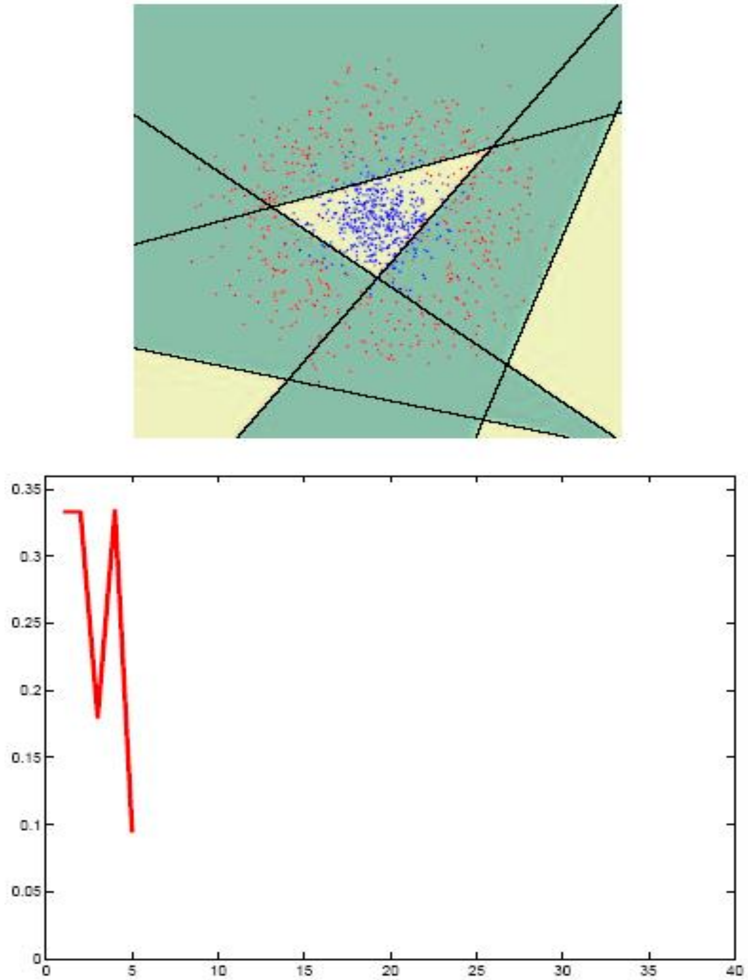


$t = 4$

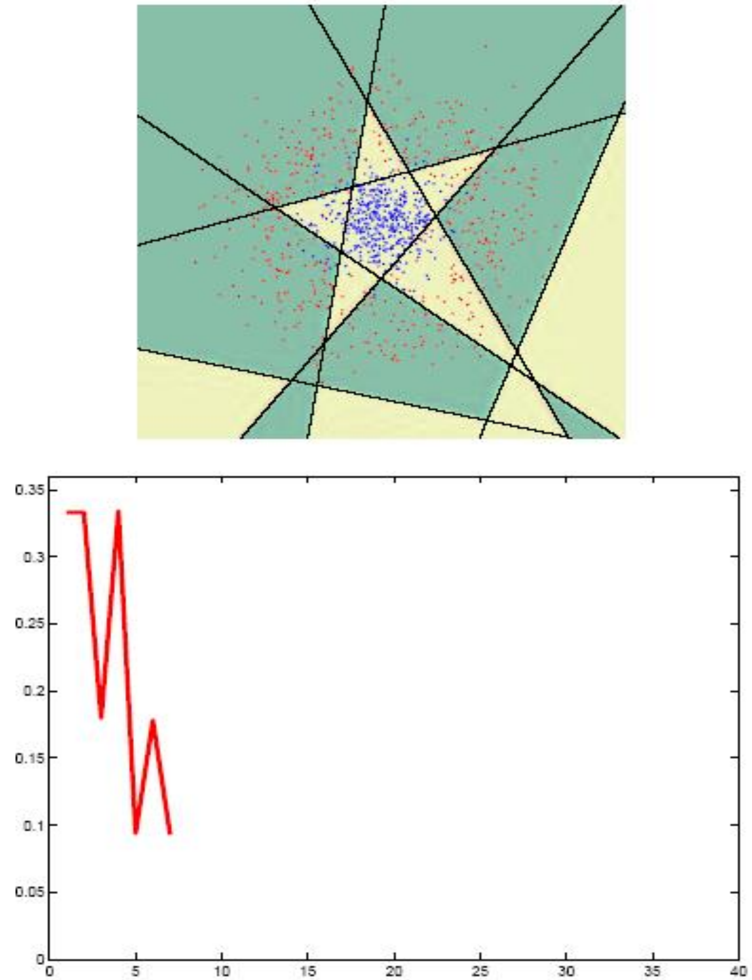


CSE 7305c

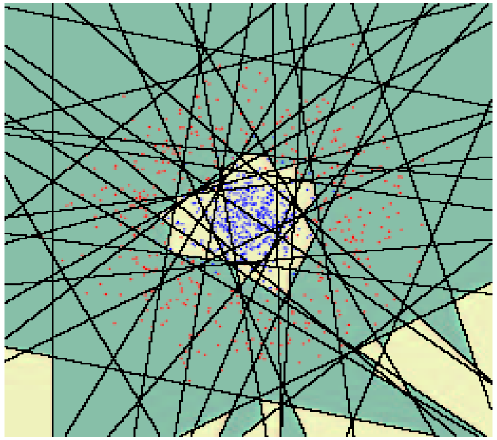
$t = 5$

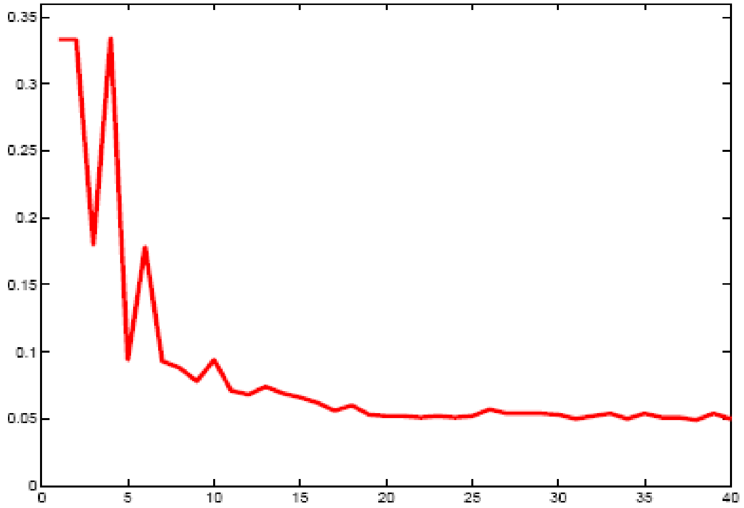

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
5
10
15
20
25
30
35
40

$t = 7$

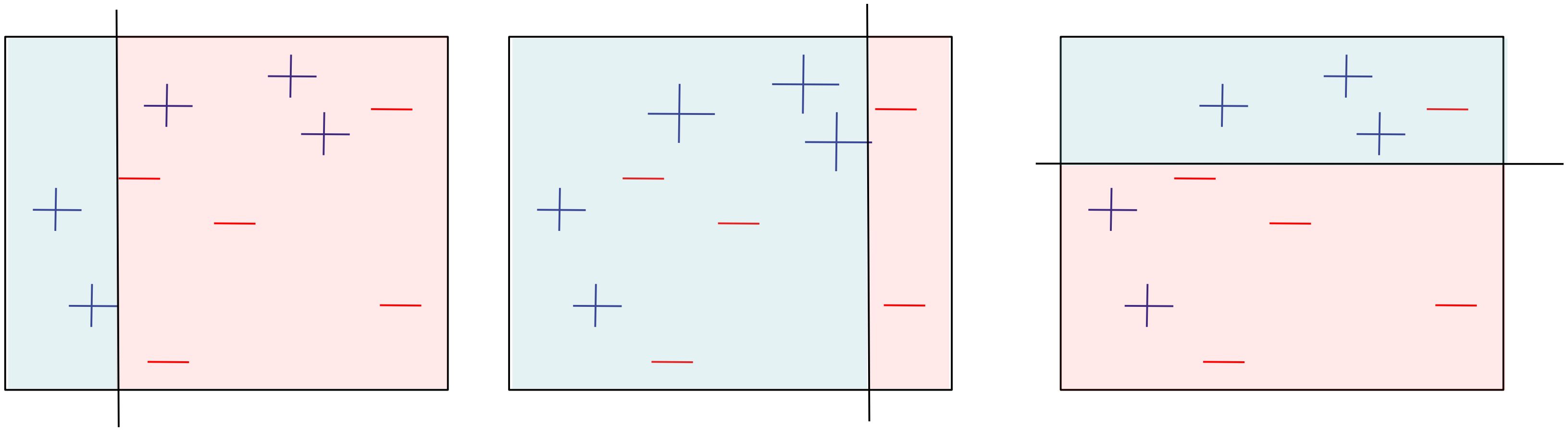

0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
5
10
15
20
25
30
35
40

$$t = 40$$

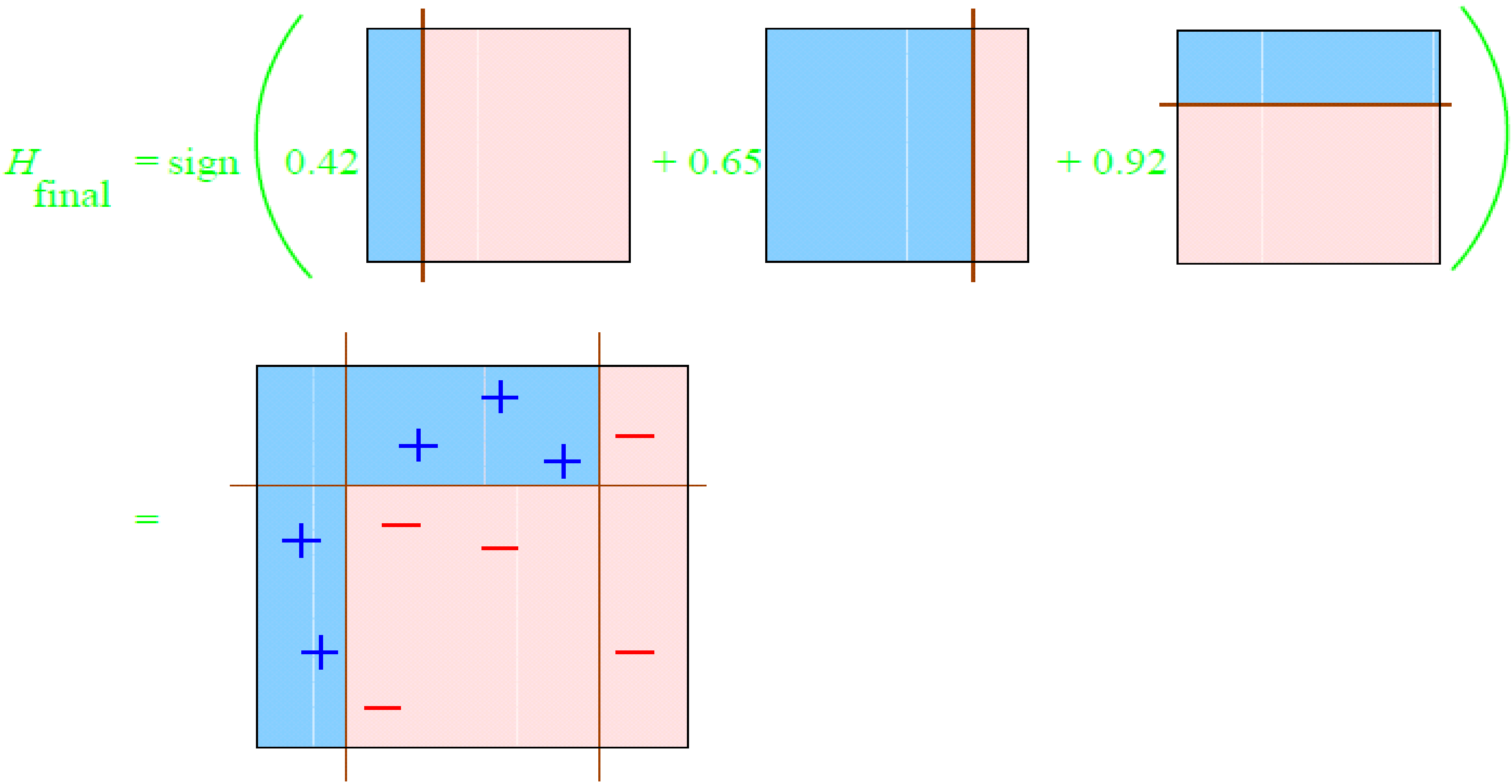




Another Toy Example



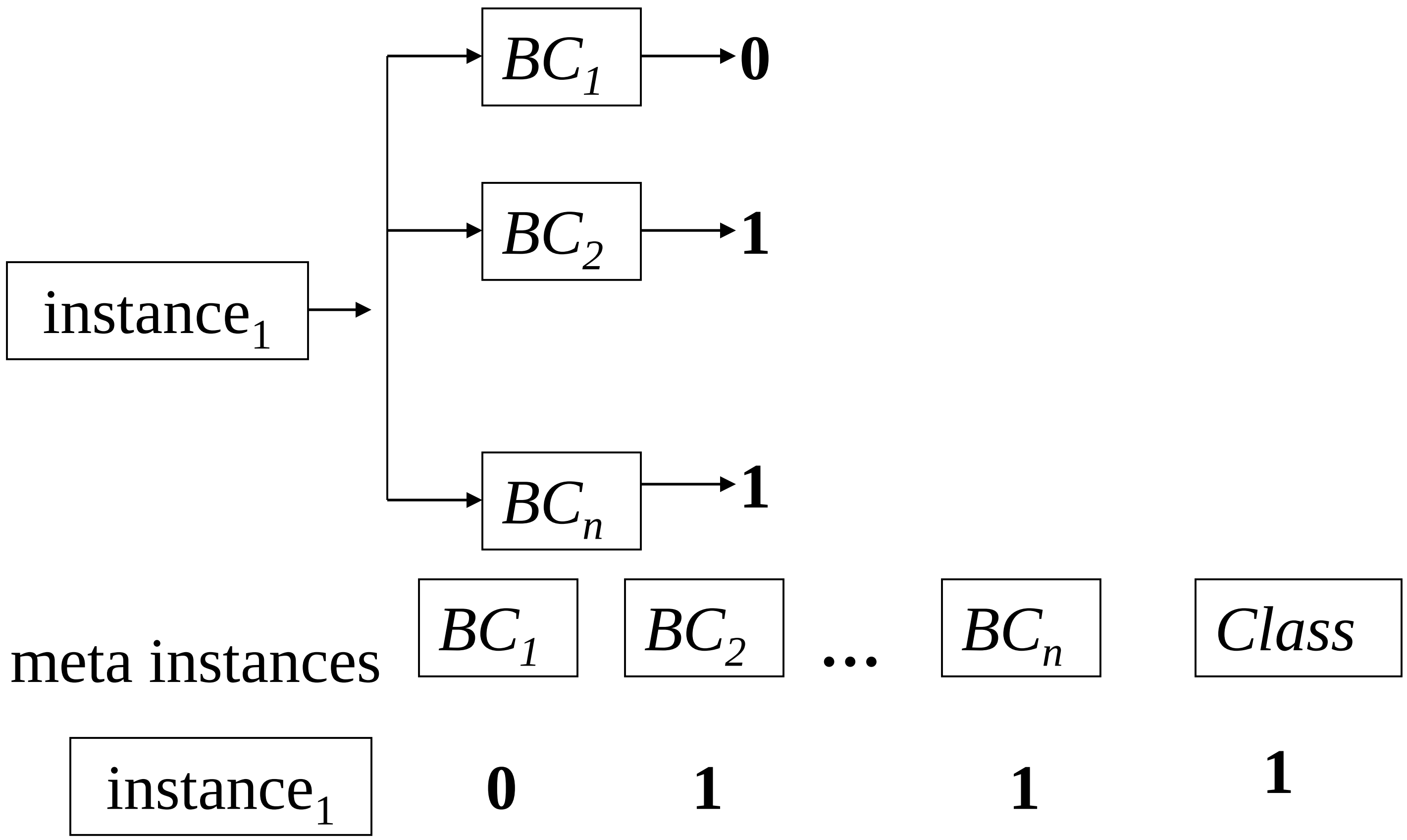
Final Classifier



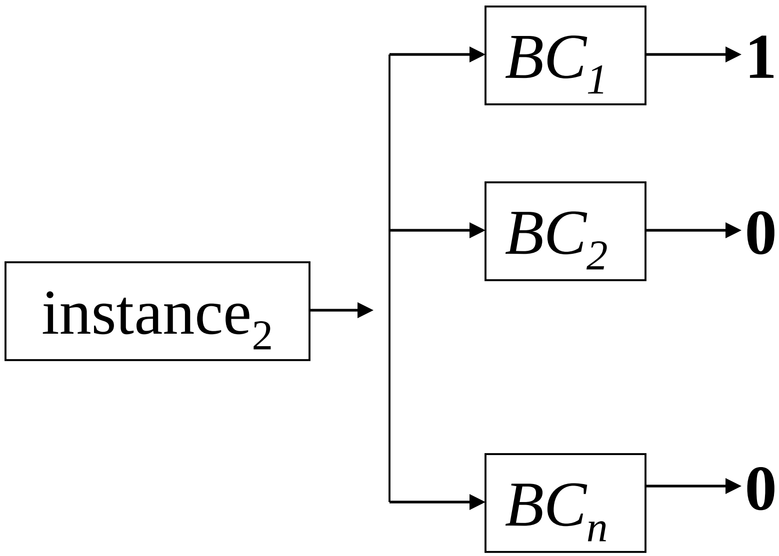
Stacking

- Uses *meta learner* instead of voting to combine predictions of base learners
  - Predictions of base learners (*level-0 models*) are used as input for meta learner (*level-1 model)*
- Base learners usually different learning schemes
- Hard to analyze theoretically: "black magic"

Stacking


$instance_1$
BC_1
0
BC_2
1
BC_n
1
meta instances
BC_1
BC_2
...
BC_n
Class
$instance_1$
0
1
1
1

Stacking



meta instances	BC_1	BC_2	…	BC_n	*Class*
instance$_1$	**0**	**1**		**1**	**1**
instance$_2$	**1**	**0**		**0**	**0**

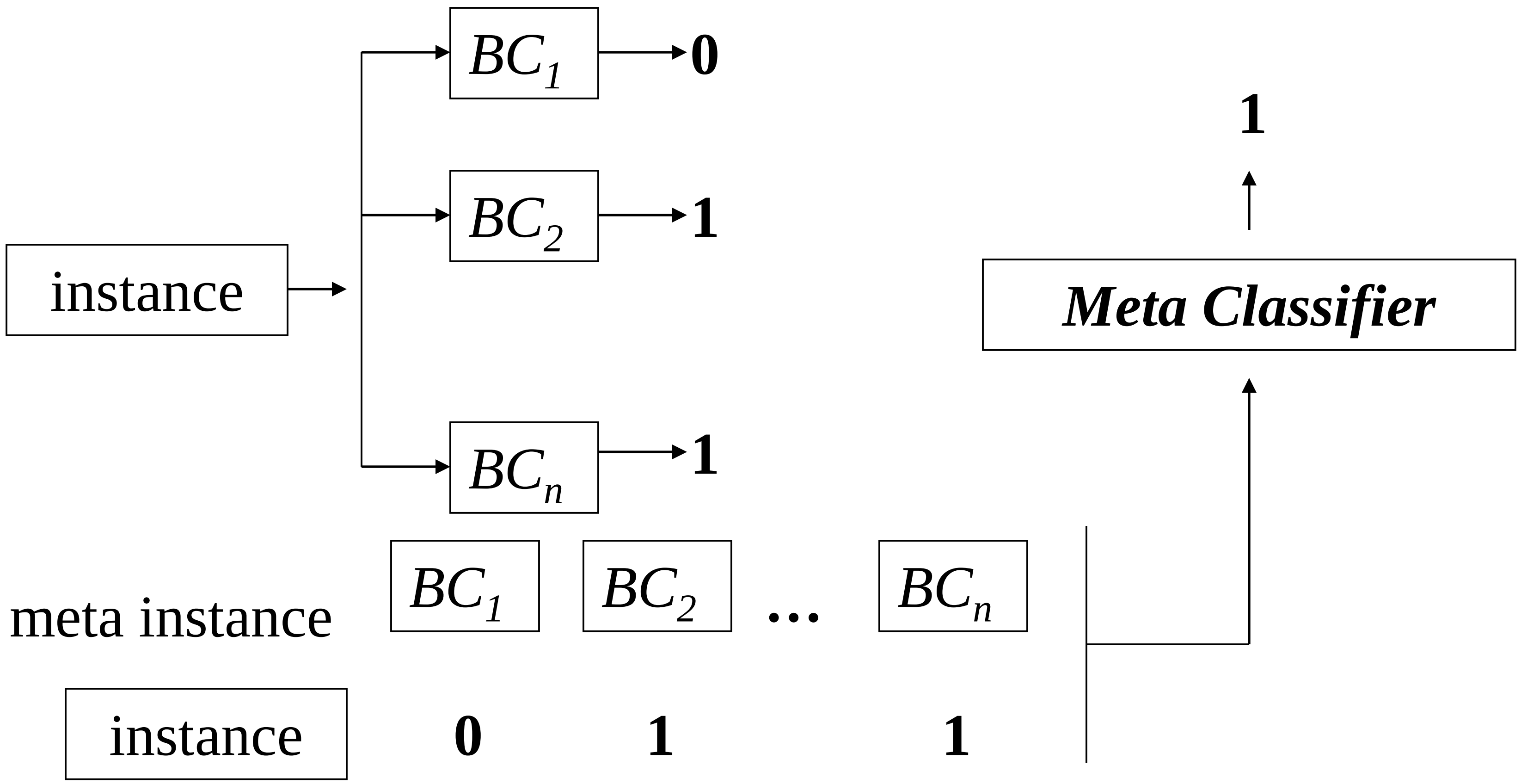
Stacking

Meta Classifier

$\Uparrow$

meta instances	BC_1	BC_2	…	BC_n	*Class*
instance_1	**0**	**1**		**1**	**1**
instance_2	**1**	**0**		**0**	**0**

Stacking



BC1
0
BC2
1
instance
BCn
1
1
Meta Classifier
meta instance
BC1
BC2
...
BCn
instance
0
1
1

More on stacking

- If base learners can output probabilities it's better to use those as input to meta learner
- Which algorithm to use to generate meta learner?
  - In principle, any learning scheme can be applied
  - David Wolpert: "relatively global, smooth" model
    - Base learners do most of the work
    - Reduces risk of overfitting

Outline

- Why ensemble learning?
- Supervised learning
  - Methods for Independently Constructing Ensembles
    - Majority Vote, Bagging and Random Forest
  - Methods for Coordinated Construction of Ensembles
    - Boosting, Stacking
- **Gradient Boosted Machines**

Gradient Boosting for Regression

Let's play a game...
You are given $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$, and the task is to fit a model $F(x)$ to minimize square loss.
Suppose your friend wants to help you and gives you a model F.
You check his model and find the model is good but not perfect.
There are some mistakes: $F(x_1) = 0.8$, while $y_1 = 0.9$, and $F(x_2) = 1.4$ while $y_2 = 1.3$... How can you improve this model?

Rule of the game:

- You are not allowed to remove anything from F or change any parameter in F.
- You can add an additional model (regression tree) h to F, so the new prediction will be $F(x) + h(x)$.

Gradient Boosting for Regression

Simple solution:

You wish to improve the model such that

$$
\begin{aligned}
F(x_1) + h(x_1) &= y_1 \\
F(x_2) + h(x_2) &= y_2 \\
&\dots \\
F(x_n) + h(x_n) &= y_n
\end{aligned}
$$

Or, equivalently, you wish

$$
\begin{aligned}
h(x_1) &= y_1 - F(x_1) \\
h(x_2) &= y_2 - F(x_2) \\
&\dots \\
h(x_n) &= y_n - F(x_n)
\end{aligned}
$$

Just fit a regression tree h to data

$(x_1, y_1 - F(x_1)), (x_2, y_2 - F(x_2)), \dots, (x_n, y_n - F(x_n))$

Gradient Boosting for Regression

$y_i - F(x_i)$ are called **residuals**. These are the parts that existing model F cannot do well.
The role of h is to compensate the shortcoming of existing model F.

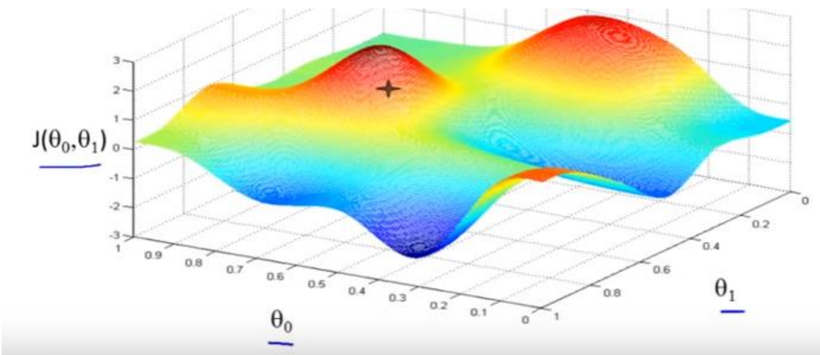
If the new model $F + h$ is still not satisfactory, we can add another regression tree...

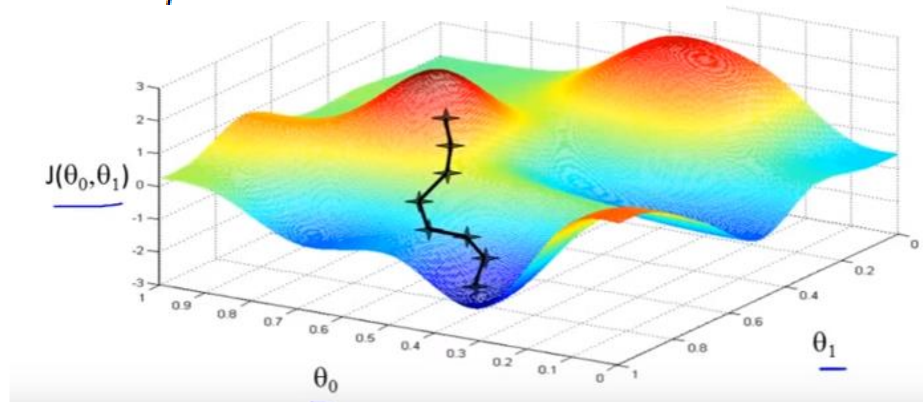
How is this related to gradient descent?

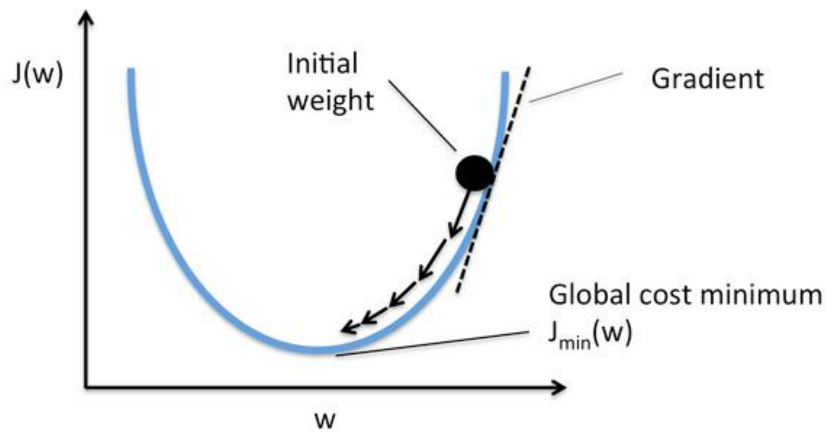
Gradient Descent

Minimize a function by moving in the opposite direction of the gradient.

$$\theta_i := \theta_i - \rho \frac{\partial J}{\partial \theta_i}$$







CSE 7305c

Relationship between Gradient descent and GBM

Loss function $L(y, F(x)) = (y - F(x))^2/2$

We want to minimize $J = \sum_i L(y_i, F(x_i))$ by adjusting $F(x_1), F(x_2), ..., F(x_n)$.

$$\frac{\partial J}{\partial F(x_i)} = \frac{\partial \sum_i L(y_i, F(x_i))}{\partial F(x_i)} = \frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} = F(x_i) - y_i$$

So we can interpret residuals as negative gradients.

$$y_i - F(x_i) = -\frac{\partial J}{\partial F(x_i)}$$

CSE 7305c

Relationship between Gradient descent and GBM

$$F(x_i) := F(x_i) + h(x_i)$$
$$F(x_i) := F(x_i) + y_i - F(x_i)$$
$$F(x_i) := F(x_i) - 1\frac{\partial J}{\partial F(x_i)}$$
$$\theta_i := \theta_i - \rho\frac{\partial J}{\partial \theta_i}$$

For regression with **square loss**,

$$\textit{residual} \Leftrightarrow \textit{negative gradient}$$

$$\textit{fit } h \textit{ to residual} \Leftrightarrow \textit{fit } h \textit{ to negative gradient}$$

$$\textit{update } F \textit{ based on residual} \Leftrightarrow \textit{update } F \textit{ based on negative gradient}$$

So we are actually updating our model using **gradient descent**!

It turns out that the concept of **gradients** is more general and useful than the concept of **residuals**.

Regression with square loss

Let us summarize the algorithm we just derived using the concept of gradients. Negative gradient:

$$-g(x_i) = -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} = y_i - F(x_i)$$

start with an initial model, say, $F(x) = \frac{\sum_{i=1}^{n} y_i}{n}$
iterate until converge:
  calculate negative gradients $-g(x_i)$
  fit a regression tree h to negative gradients $-g(x_i)$
  $F := F + \rho h$, where $\rho = 1$

The benefit of formulating this algorithm using gradients is that it allows us to consider other loss functions and derive the corresponding algorithms in the same way.

CSE 7305c

Loss Functions for Regression functions

Why do we need to consider other loss functions? Isn't square loss good enough?

Square loss is:

- ✓ Easy to deal with mathematically
- × Not robust to outliers

  Outliers are heavily punished because the error is squared.

  Example:

y_i	0.5	1.2	2	5*
$F(x_i)$	0.6	1.4	1.5	1.7
$L = (y - F)^2/2$	0.005	0.02	0.125	5.445

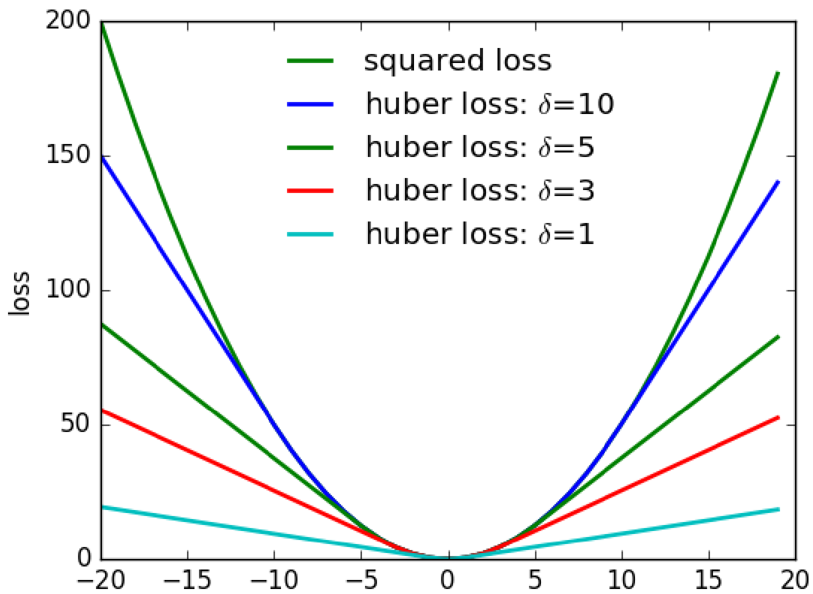
  Consequence?

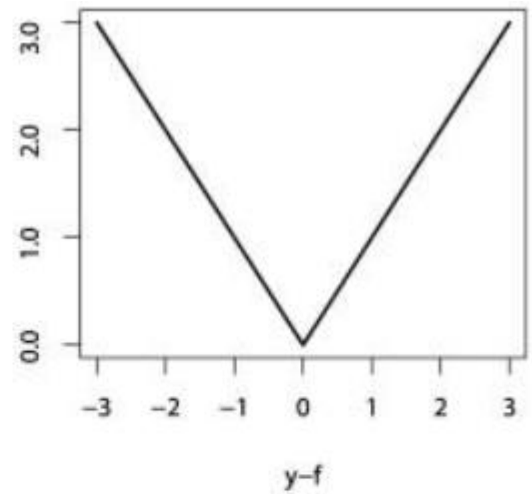
  Pay too much attention to outliers. Try hard to incorporate outliers into the model. Degrade the overall performance.

Loss Functions for Regression functions

- Absolute loss (more robust to outliers)

$$L(y, F) = |y - F|$$





absolute loss function

		1.4	1.5	1.7
		0.02	0.125	5.445
Absolute loss	0.1	0.2	0.5	3.3
Huber loss($\delta = 0.5$)	0.005	0.02	0.125	1.525

Regression with other Loss Functions

Regression with Absolute Loss

Negative gradient:

$$-g(x_i) = -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} = sign(y_i - F(x_i))$$

Regression with Huber Loss

Negative gradient:

$$\begin{aligned} -g(x_i) &= -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \\ &= \begin{cases} y_i - F(x_i) & |y_i - F(x_i)| \le \delta \\ \delta sign(y_i - F(x_i)) & |y_i - F(x_i)| > \delta \end{cases} \end{aligned}$$

In general,

$$negative\ gradients \not\Leftrightarrow residuals$$

We should follow negative gradients rather than residuals. Why?

CSE 7305c

Outline

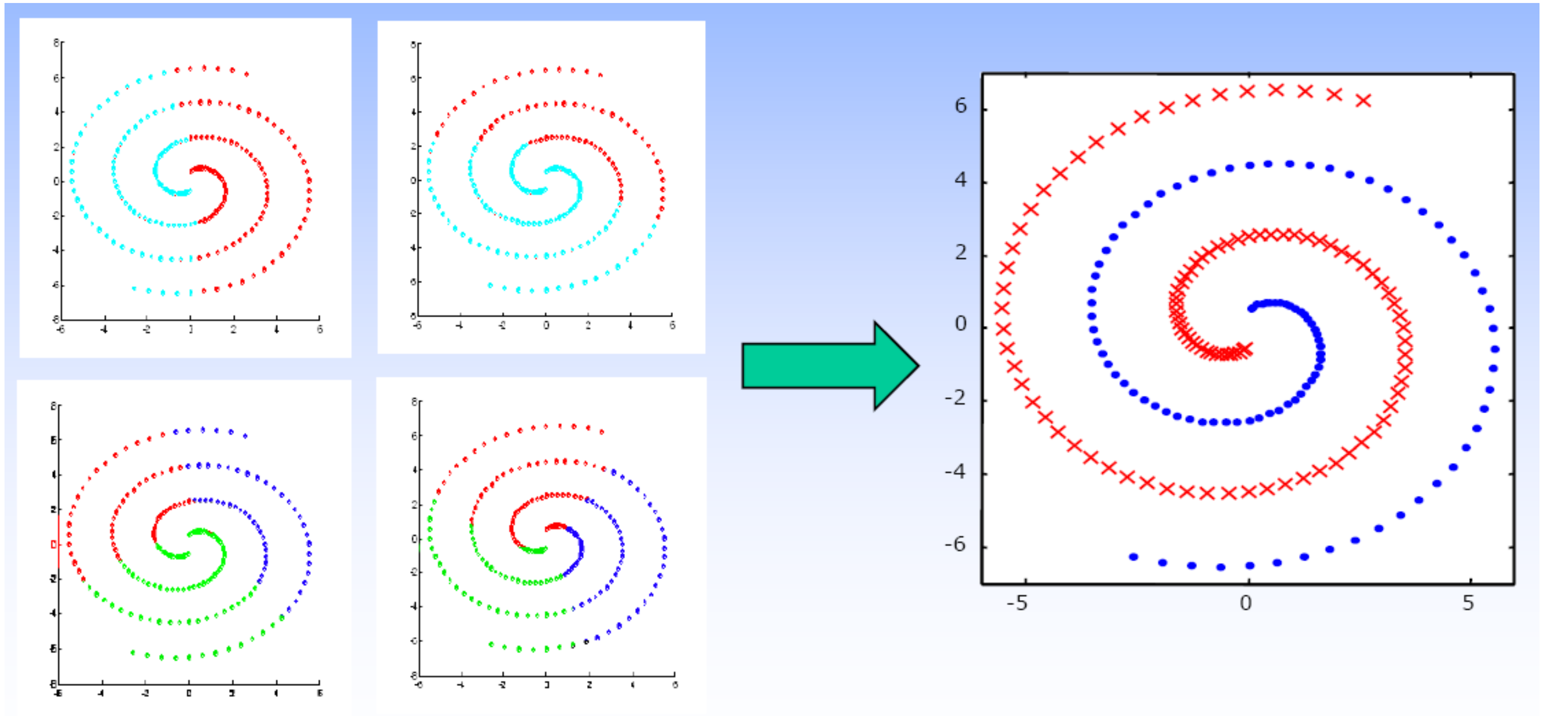
- Why ensemble learning?
- Supervised learning
  - Methods for Independently Constructing Ensembles
    - Majority Vote, Bagging and Random Forest
  - Methods for Coordinated Construction of Ensembles
    - Boosting, Stacking
- Gradient Boosted Machines
- **Unsupervised learning**
  - **Clustering ensembles**

Clustering Ensemble

- Problem
  - Given an unlabeled data set $D=\{x_1,x_2,\ldots,x_n\}$
  - An ensemble approach computes:
    - A set of clustering solutions $\{C_1,C_2,\ldots,C_k\}$, each of which maps data to a cluster: $f_j(x)=m$
    - A unified clustering solutions f^* which combines base clustering solutions by their consensus
- Challenges
  - The correspondence between the clusters in different clustering solutions is unknown
  - Unsupervised
  - Combinatorial optimization problem-NP-complete

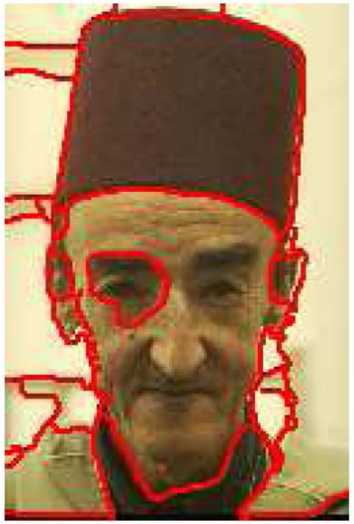
Motivations

- Goal
  - Combine "weak" clusterings to a better one

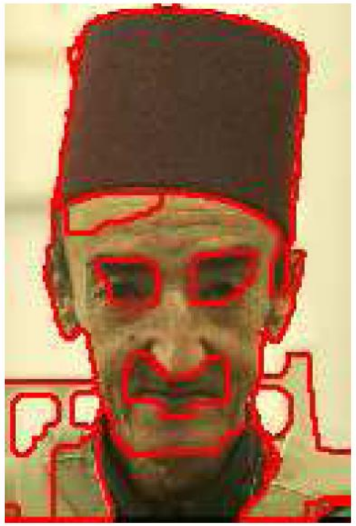


CSE 7305c

(a)



(b)



(c)

(d)

ensemble

CSE 7305c

Take-aways

- Ensembles learning is very useful in obtaining improved models.
- We discussed ensemble learning from two perspectives
  - Supervised ensembles
  - Unsupervised ensembles

Tutorial on Ensemble of Classifiers

- *Survey of Boosting from an Optimization Perspective*. Manfred K. Warmuth and S.V.N. Vishwanathan. ICML'09, Montreal, Canada, June 2009.
- *Theory and Applications of Boosting*. Robert Schapire. NIPS'07, Vancouver, Canada, December 2007.
- *From Trees to Forests and Rule Sets--A Unified Overview of Ensemble Methods*. Giovanni Seni and John Elder. KDD'07, San Jose, CA, August 2007.

References

- [AUL08] M. Amini, N. Usunier, and F. Laviolette. A transductive bound for the voted classifier with an application to semi-supervised learning. In Advances in Neural Information Processing Systems 21, 2008.
- [BBY04] M. Balcan, A. Blum, and K. Yang. Co-training and expansion: Towards bridging theory and practice. In Advances in Neural Information Processing Systems 17, 2004.
- [BBM07] A. Banerjee, S. Basu, and S. Merugu. Multi-way clustering on relation graphs. In Proc. 2007 SIAM Int. Conf. Data Mining (SDM'07), 2007.
- [BaKo04] E. Bauer and R. Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. Machine Learning, 36:105-139, 2004.
- [BEM05] R. Bekkerman, R. El-Yaniv, and A. McCallum. Multi-way distributional clustering via pairwise interactions. In Proc. 2005 Int. Conf. Machine Learning (ICML'05), pages 41-48, 2005.
- [BDH05] P. N. Bennett, S. T. Dumais, and E. Horvitz. The combination of text classifiers using reliability indicators. Information Retrieval, 8(1):67-100, 2005.
- [BiSc04] S. Bickel and T. Scheffer. Multi-view clustering. In Proc. 2004 Int. Conf. Data Mining (ICDM'04), pages 19-26, 2004.
- [BlMi98] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. Proceedings of the Workshop on Computational Learning Theory, pages 92-100, 1998.
- [BGS+08] P. Brazdil, C. Giraud-Carrier, C. Soares, and R. Vilalta. Metalearning: Applications to Data Mining. Springer, 2008.
- [BBS05] Ulf Brefeld, Christoph Büscher, and Tobias Scheffer. Multi-view discriminative sequential learning. In Proc. European Conf. Machine Learning (ECML'05), pages 60-71, 2005.
- [Breiman96] L. Breiman. Bagging predictors. Machine Learning, 26:123-140, 1996.
- [Breiman01] L. Breiman. Random forests. Machine Learning, 45(1):5-32, 2001.
- [Caruana97] R. Caruana. Multitask learning. Machine Learning, 28(1):41-75, 1997.

CSE 7305c

References

- [CoSi99] M. Collins and Y. Singer. Unsupervised models for named entity classification. In Proc. 1999 Conf. Empirical Methods in Natural Language Processing (EMNLP'99), 1999.
- [CKW08] K. Crammer, M. Kearns, and J. Wortman. Learning from multiple sources. Journal of Machine Learning Research, 9:1757-1774, 2008.
- [DYX+07] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu. Boosting for transfer learning. In Proc. 2007 Int. Conf. Machine Learning (ICML'07), pages 193-200, 2007.
- [DLM01] S. Dasgupta, M. Littman, and D. McAllester. PAC Generalization Bounds for Co-training. In Advances in Neural Information Processing Systems 14, 2001.
- [DaFa06] I. Davidson and W. Fan. When efficient model averaging out-performs boosting and bagging. In Proc. 2006 European Conf. Principles and Practice of Knowledge Discovery in Databases (PKDD'06), pages 478-486, 2006.
- [DMM03] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretic co-clustering. In Proc. 2003 ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD'03), pages 89-98, 2003.
- [Dietterich00] T. Dietterich. Ensemble methods in machine learning. In Proc. 2000 Int. Workshop Multiple Classifier Systems, pages 1-15, 2000.
- [DWH01] E. Dimitriadou, A. Weingessel, and K. Homik. Voting-merging: an ensemble method for clustering. In Proc. 2001 Int. Conf. Artificial Neural Networks (ICANN'01), pages 217-224, 2001.
- [DoAl09] C. Domeniconi and M. Al-Razgan. Weighted cluster ensembles: Methods and analysis. ACM Transactions on Knowledge Discovery from Data (TKDD), 2(4):1-40, 2009.
- [Domingos00] P. Domingos. Bayesian averaging of classifiers and the overfitting problem. In Proc. 2000 Int. Conf. Machine Learning (ICML'00), pages 223-230, 2000.
- [DHS01] R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification. John Wiley & Sons, second edition, 2001.

References

- [DzZe02] S. Dzeroski and B. Zenko. Is combining classifiers better than selecting the best one. In Proc. 2002 Int. Conf. Machine Learning (ICML'02), pages 123-130, 2002.
- [DuFr03] S. Dudoit and J. Fridlyand. Bagging to improve the accuracy of a clustering procedure. Bioinformatics, 19(9): 1090-1099, 2003.
- [FaDa07] W. Fan and I. Davidson. On sample selection bias and its efficient correction via model averaging and unlabeled examples. In Proc. 2007 SIAM Int. Conf. Data Mining (SDM'07), 2007.
- [FGM+05] W. Fan, E. Greengrass, J. McCloskey, P. S. Yu, and K. Drummey. Effective estimation of posterior probabilities: Explaining the accuracy of randomized decision tree approaches. In Proc. 2005 Int. Conf. Data Mining (ICDM'05), pages 154-161, 2005.
- [FHM+05] J. Farquhar, D. Hardoon, H. Meng, J. Shawe-taylor, and S. Szedmak. Two view learning: SVM-2K, theory and practice. In Advances in Neural Information Processing Systems 18, 2005.
- [FeBr04] X. Z. Fern and C. E. Brodley. Solving cluster ensemble problems by bipartite graph partitioning. In Proc. 2004 Int. Conf. Machine Learning (ICML'04), pages 281-288, 2004.
- [FeLi08] X. Z. Fern and W. Lin. Cluster ensemble selection. In Proc. 2008 SIAM Int. Conf. Data Mining (SDM'08), 2008.
- [FiSk03] V. Filkov and S. Skiena. Integrating microarray data by consensus clustering. In Proc. 2003 Int. Conf. Tools with Artificial Intelligence, pages 418-426, 2003.
- [FrJa02] A. Fred and A. Jain. Data Clustering using evidence accumulation. In Proc. 2002 Int. Conf. Pattern Recognition (ICPR'02), 2002.
- [FrSc97] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1):119-139, 1997.

References

- [FrPo08] J. H. Friedman and B. E. Popescu. Predictive learning via rule ensembles. Annals of Applied Statistics, 3(2):916-954, 2008.
- [GGB+08] K. Ganchev, J. Graca, J. Blitzer, and B. Taskar. Multi-view learning over structured and non-identical outputs. In Proc. 2008 Conf. Uncertainty in Artificial Intelligence (UAI'08), pages 204-211, 2008.
- [GFH07] J. Gao, W. Fan, and J. Han. On appropriate assumptions to mine data streams: Analysis and practice. In Proc. 2007 Int. Conf. Data Mining (ICDM'07), pages 143-152, 2007.
- [GFJ+08] J. Gao, W. Fan, J. Jiang, and J. Han. Knowledge transfer via multiple model local structure mapping. In Proc. 2008 ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD'08), pages 283-291, 2008.
- [GFS+09] J. Gao, W. Fan, Y. Sun, and J. Han. Heterogeneous source consensus learning via decision propagation and negotiation. In Proc. 2009 ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD'09), pages 339-347, 2009.
- [GLF+09] J. Gao, F. Liang, W. Fan, Y. Sun, and J. Han. Graph-based consensus maximization among multiple supervised and unsupervised models. In Advances in Neural Information Processing Systems 22, 2009.
- [GSI+09] R. Ghaemi, M. Sulaiman, H. Ibrahim, and N. Mutspha. A survey: clustering ensembles techniques. World Academy of Science, Engineering and Technology 50, 2009.
- [GeTa07] L. Getoor and B. Taskar. Introduction to statistical relational learning. MIT Press, 2007.
- [GMT07] A. Gionis, H. Mannila, and P. Tsaparas. Clustering aggregation. ACM Transactions on Knowledge Discovery from Data (TKDD), 1(1), 2007.
- [GVB04] C. Giraud-Carrier, R. Vilalta, and P. Brazdil. Introduction to the special issue on meta-learning. Machine Learning, 54(3):187-193, 2004.

References

- [GoFi08] A. Goder and V. Filkov. Consensus clustering algorithms: comparison and refinement. In Proc. 2008 Workshop on Algorithm Engineering and Experiments (ALENEX'08), pages 109-117, 2008.
- [GoZh00] S. Goldman and Y. Zhou. Enhancing supervised learning with unlabeled data. In Proc. 2000 Int. Conf. Machine Learning (ICML'00), pages 327-334, 2000.
- [HKT06] S. T. Hadjitodorov, L. I. Kuncheva, and L. P. Todorova. Moderate diversity for better cluster ensembles. Information Fusion, 7(3):264-275, 2006.
- [HaKa06] J. Han and M. Kamber. Data mining: concepts and techniques. Morgan Kaufmann, second edition, 2006.
- [HTF09] T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, second edition, 2009.
- [HMR+99] J. Hoeting, D. Madigan, A. Raftery, and C. Volinsky. Bayesian model averaging: a tutorial. Statistical Science, 14:382-417, 1999.
- [JJN+91] R. Jacobs, M. Jordan, S. Nowlan, and G. Hinton. Adaptive mixtures of local experts. Neural Computation, 3(1):79-87, 1991.
- [KoMa] J. Kolter and M. Maloof. Using additive expert ensembles to cope with concept drift. In Proc. 2005 Int. Conf. Machine Learning (ICML'05), pages 449-456, 2005.
- [KuWh03] L. I. Kuncheva and C. J. Whitaker. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine Learning, 51(2):181-207, 2003.
- [Leskes05] B. Leskes. The Value of Agreement, a New Boosting Algorithm. In 2005 Proc. Conf. Learning Theory (COLT'05), pages 95-110, 2005.
- [LiDi08] T. Li and C. Ding. Weighted consensus clustering. In Proc. 2008 SIAM Int. Conf. Data Mining (SDM'08), 2008.

CSE 7305c

References

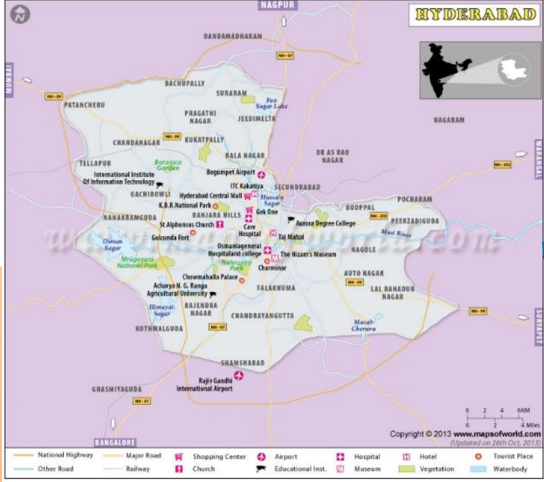
- [LDJ07] T. Li, C. Ding, and M. Jordan. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In Proc. 2007 Int. Conf. Data Mining (ICDM'07), pages 577-582, 2007.
- [LiOg05] T. Li and M. Ogihara. Semisupervised learning from different information sources. Knowledge and Information Systems, 7(3):289-309, 2005.
- [LiYa06] C. X. Ling and Q. Yang. Discovering classification from data of multiple sources. Data Mining and Knowledge Discovery, 12(2-3):181-201, 2006.
- [LZY05] B. Long, Z. Zhang, and P. S. Yu. Combining multiple clusterings by soft correspondence. In Proc. 2005 Int. Conf. Data Mining (ICDM'05), pages 282-289, 2005.
- [LZX+08] P. Luo, F. Zhuang, H. Xiong, Y. Xiong, and Q. He. Transfer learning from multiple source domains via consensus regularization. In Proc. 2008 Int. Conf. Information and Knowledge Management (CIKM'08), pages 103-112, 2008.
- [MTP04] B. Minaei-Bidgoli, A. Topchy, and W. Punch: A comparison of resampling methods for clustering ensembles. In Proc. 2004 Int. Conf. Artificial Intelligence (ICAI'04), pages 939-945, 2004.
- [NiGh00] K. Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In Proc. 2000 Int. Conf. Information and Knowledge Management (CIKM'00), pages 86-93, 2000.
- [OkVa08] O. Okun and G. Valentini. Supervised and Unsupervised Ensemble Methods and their Applications. Springer, 2008.
- [Polikar06] R. Polikar. Ensemble based systems in decision making. IEEE Circuits and Systems Magazine, 6(3):21-45, 2006.
- [PrSc08] C. Preisach and L. Schmidt-Thieme. Ensembles of relational classifiers. Knowledge and Information Systems, 14(3):249-272, 2008.

References

- [PTJ05] W. Punch, A. Topchy, and A. K. Jain. Clustering ensembles: Models of consensus and weak partitions. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(12):1866-1881, 2005.
- [PuGh08] K. Punera and J. Ghosh. Consensus based ensembles of soft clusterings. Applied Artificial Intelligence, 22(7-8): 780-810, 2008.
- [RoKa07] D. M. Roy and L. P. Kaelbling. Efficient bayesian task-level transfer learning. In Proc. 2007 Int. Joint Conf. Artificial Intelligence (IJCAI'07), pages 2599-2604, 2007.
- [SNB05] V. Sindhwani, P. Niyogi, and M. Belkin. A co-regularization approach to semi-supervised learning with multiple views. In Proc. 2005 ICML workshop on Learning with Multiple Views, 2005.
- [SMP+07] V. Singh, L. Mukherjee, J. Peng, and J. Xu. Ensemble clustering using semidefinite programming. In Advances in Neural Information Processing Systems 20, 2007.
- [StGh03] A. Strehl and J. Ghosh. Cluster ensembles --a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research, 3:583-617, 2003.
- [TLJ+04] A. Topchy, M. Law, A. Jain, and A. Fred. Analysis of consensus partition in cluster ensemble. In Proc. 2004 Int. Conf. Data Mining (ICDM'04), pages 225-232, 2004.
- [TuGh96] K. Tumer and J. Ghosh. Analysis of decision boundaries in linearly combined neural classifiers. Pattern Recognition, 29, 1996.
- [ViDr02] R. Vilalta and Y. Drissi. A perspective view and survey of meta-learning. Artificial Intelligence Review, 18(2):77-95, 2002.
- [WFY+03] H. Wang, W. Fan, P. Yu, and J. Han. Mining concept-drifting data streams using ensemble classifiers. In Proc. 2003 ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD'03), pages 226-235, 2003.
- [WSB09] H. Wang, H. Shan, and A. Banerjee. Bayesian cluster ensembles. In Proc. 2009 SIAM Int. Conf. Data Mining (SDM'09), 2009.

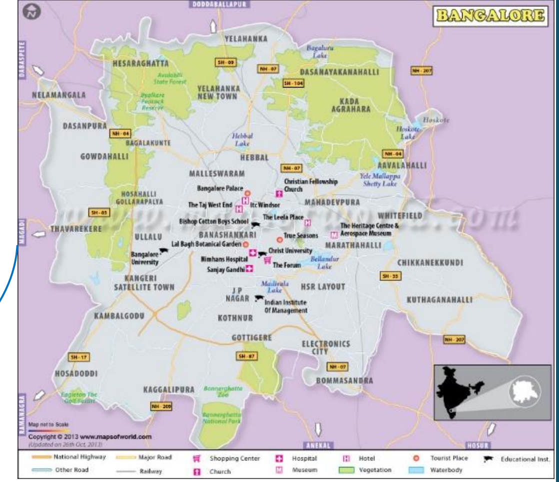
References

- [Wolpert92] D. H. Wolpert. Stacked generalization. Neural Networks, 5:241-259, 1992.
- [WWL09] F. Wang, X. Wang, and T. Li.Generalized Cluster aggregation. In Proc. 2009 Int. Joint Conf. Artificial Intelligence (IJCAI'09), pages 1279-1284, 2009.
- [ZGY05] J. Zhang, Z. Ghahramani, and Y. Yang. Learning multiple related tasks using latent independent component. In Advances in Neural Information Processing Systems 18, 2005.
- [ZFY+06] K. Zhang, W. Fan, X. Yuan, I. Davidson, and X. Li. Forecasting skewed biased stochastic ozone days: Analyses and solutions. In Proc. 2006 Int. Conf. Data Mining (ICDM'06), pages 753-764, 2006.
- [ZZY07] Z. Zhou, D. Zhan, and Q. Yang. Semi-Supervised Learning with Very Few Labeled Training Examples. In Proc. 2007 Conf. Artificial Intelligence (AAAI'07), pages 675-680, 2007.



INSOFE

Inspire...Educate...Transform.



HYDERABAD

Office and Classrooms

Plot 63/A, Floors 1&2, Road # 13, Film Nagar,
Jubilee Hills, Hyderabad - 500 033
+91-9701685511 (Individuals)
+91-9618483483 (Corporates)

BENGALURU

Office

Incubex, #728, Grace Platina, 4th Floor, CMH Road,
Indira Nagar, 1st Stage, Bengaluru – 560038
+91-9502334561 (Individuals)
+91-9502799088 (Corporates)

Classroom

KnowledgeHut Solutions Pvt. Ltd., Reliable Plaza,
Jakkasandra Main Road, Teacher's Colony, 14th Main
Road, Sector – 5, HSR Layout, Bengaluru - 560102

Social Media

Web: http://www.insofe.edu.in

Facebook: https://www.facebook.com/insofe

Twitter: https://twitter.com/Insofeedu

YouTube: http://www.youtube.com/InsofeVideos

SlideShare: http://www.slideshare.net/INSOFE

LinkedIn: http://www.linkedin.com/company/international-school-of-engineering