

# UVA CS 4501 - 001 / 6501 – 007

## Introduction to Machine Learning and Data Mining

### Lecture 19: Decision Tree / Random Forest / Ensemble

Yanjun Qi / Jane, , PhD

University of Virginia  
Department of  
Computer Science

11/6/14

1

Where are we ? →

Five major sections of this course

- Regression (supervised)
- Classification (supervised)
- Unsupervised models
- Learning theory
- Graphical models

11/6/14

2

# Announcements: Rough Plan

E	Th - Oct. 30	<b>Midterm</b>	HW4 due;	.
L19	Tu - Nov. 4	Lecture slides	.	Decision Tree / Ensemble
L20	Th - Nov. 6	Lecture slides	HW5 out (4501- Programming / 6501- Proposal)	Neural Network / Deep Learning
L21	Tu - Nov. 11	Lecture slides	.	PCA, SVD
L22	Th - Nov. 13	Lecture slides	.	K-Means clustering / Hierarchical Clustering
L23	Tu - Nov. 18	Lecture slides	HW6 out;	GMM /EM
L24	Th - Nov. 20	Lecture slides	HW5 due (Sun)	Graphical Model / Bayes Nets
L25	Tu - Nov. 25	Lecture slides	.	Hidden Markov Model
E	Th - Nov. 27	<b>NO CLASS (Thanksgiving Holiday)</b>	.	.
L26	Tu - Dec. 2	<b>Final Review</b>	HW6 due (Wed)	.
E	Th - Dec. 4	<b>Final exam (in class exam)</b>	.	.

11/6/14

3

# Announcements: Rough Plan

- HW5: out this Thursday, due Nov. 23<sup>rd</sup> midnight
  - **4501 – 001:** Programming on image labeling task (similar to HW3)
  - **6501 – 007:** Grant proposal / Business plan
  - **If finishing both, 5 extra credits**
- HW6:
  - similar to HW4
  - 12 samples questions for the preparation for final
  - covering topics after midterm
- Final exam:
  - In class @ **Thornton Hall E316**
  - Dec 4<sup>th</sup> 3:35pm-4:45pm (70 mins)

11/6/14

4

## Announcements: Rough Plan

- HW3 Grades + Solution / Will be posted in Collab this weekend
- HW4 Grades / Will be posted in Collab this weekend
- Midterm Grades + Solution / Will be posted in Collab this weekend

## 6501 – 007 HW5: Project Proposal/ Grant Proposal / Business Plan

- The following sections are expected in the proposal
  - Minimum 8 pages (latex and word template will be provided):
  - Abstract / Summary (300 words limit)
  - Introduction of the target task
  - Previous solution for the target task
  - Why you are the right person for implementing this plan ?
  - Why machine learning methods are good candidates for the target task
  - Method you propose / Please provide a diagram of your proposed system
  - Experimental design / details of the data / where to get / data statistics / preliminary results ? Preliminary prototype ?
  - References (not included in the page counts)
  - Budget plan / Personnel plan are not required

# Where are we ? →

## Five major sections of this course

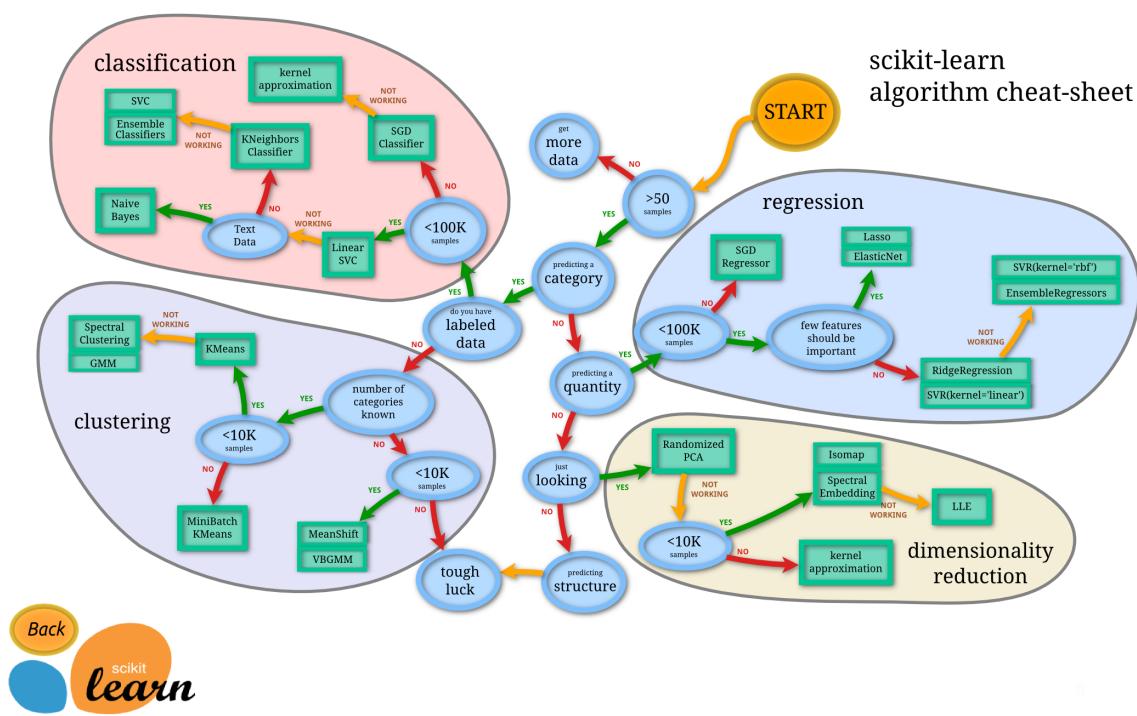
- Regression (supervised)
- Classification (supervised)
- Unsupervised models
- Learning theory
- Graphical models

11/6/14

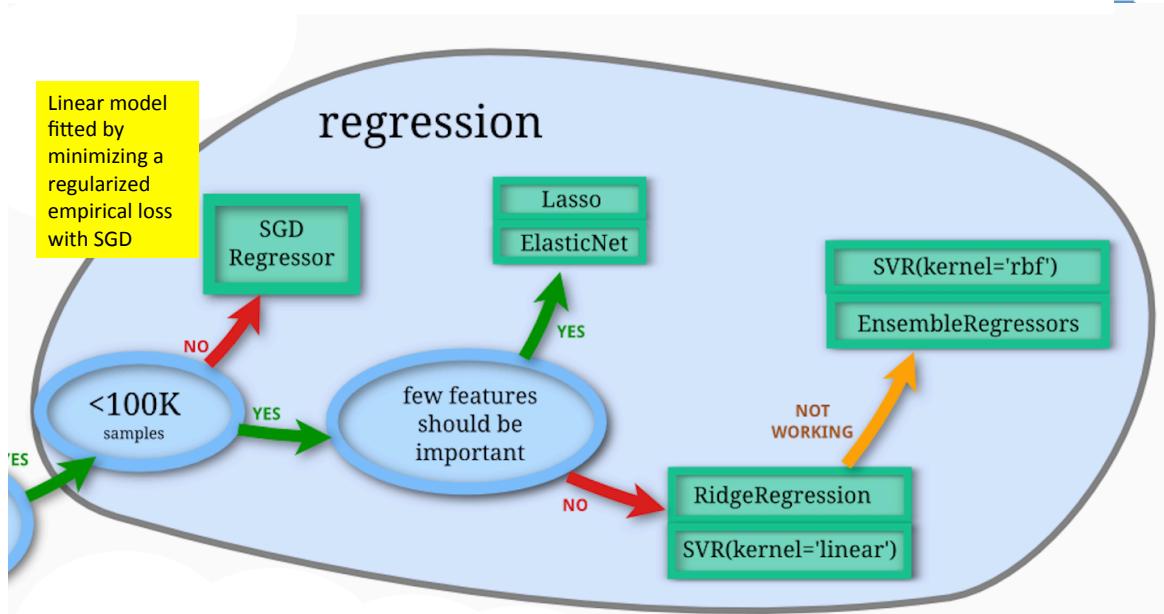
7

[http://scikit-learn.org/stable/tutorial/machine\\_learning\\_map/](http://scikit-learn.org/stable/tutorial/machine_learning_map/)

Yanjun Qi / UVA CS 4501-01-6501-07

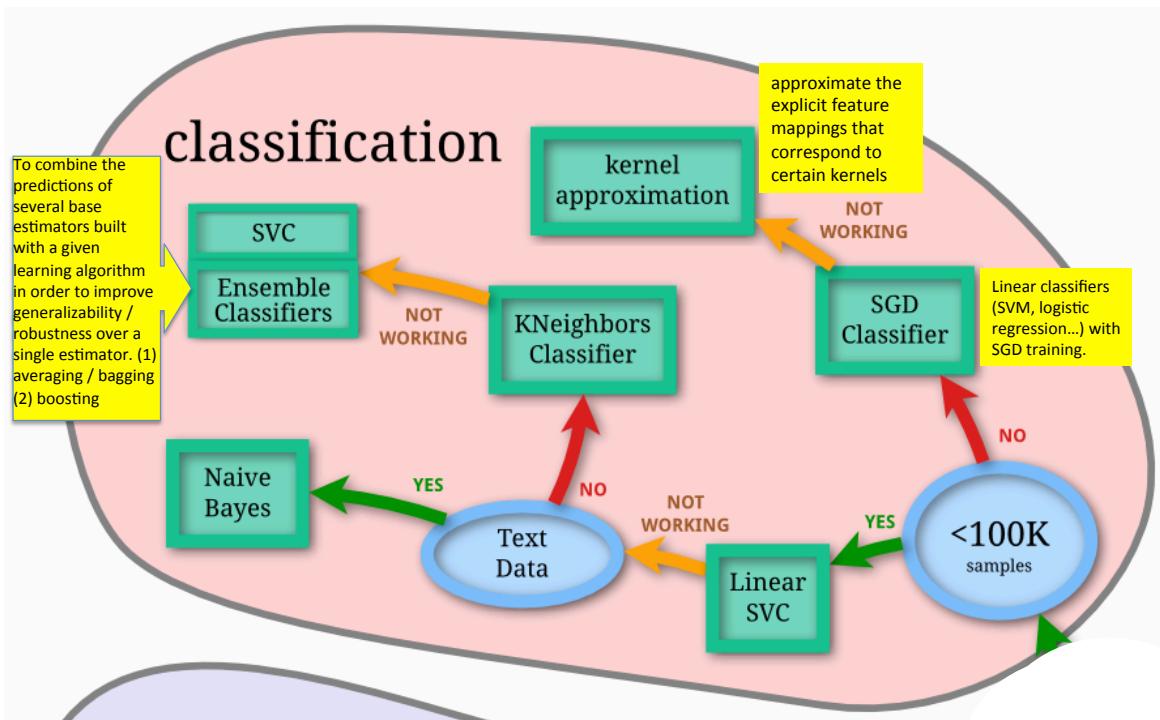


# Scikit-learn : Regression

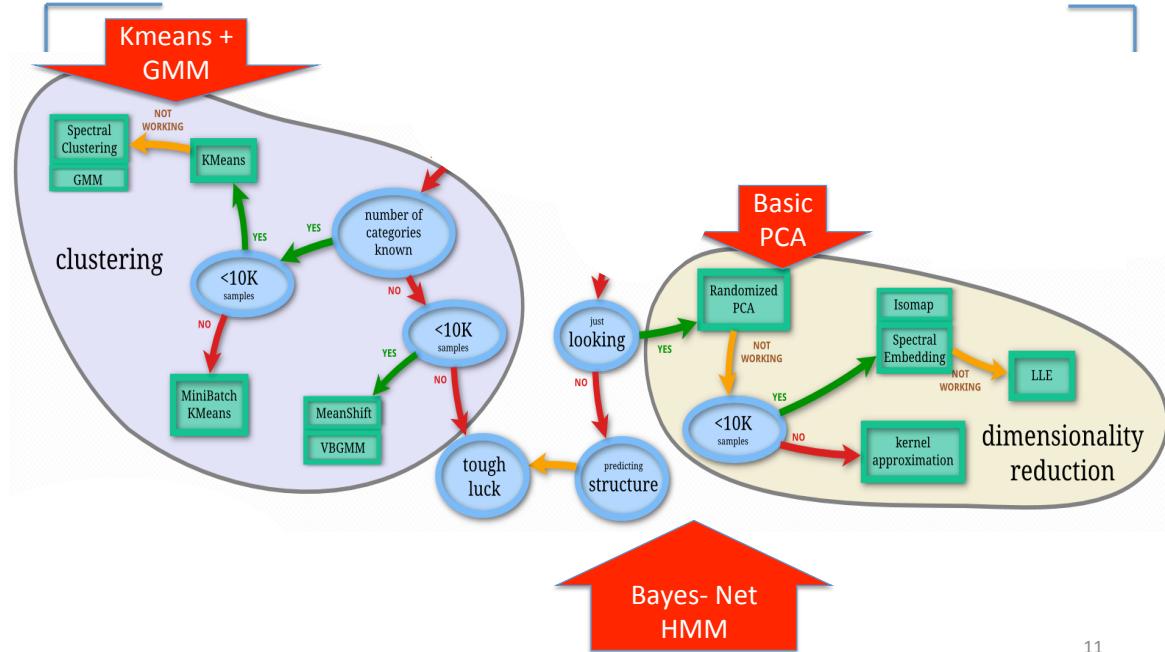


9

# Scikit-learn : Classification



# next after classification ?



11

## Today

- Decision Tree (DT):
- Tree representation
- Brief information theory
- Learning decision trees
- Bagging
- Random forests: Ensemble of DT
- More about ensemble

# A study comparing Classifiers

## An Empirical Comparison of Supervised Learning Algorithms

Rich Caruana

Alexandru Niculescu-Mizil

Department of Computer Science, Cornell University, Ithaca, NY 14853 USA

CARUANA@CS.CORNELL.EDU

ALEXN@CS.CORNELL.EDU

### Abstract

A number of supervised learning methods have been introduced in the last decade. Unfortunately, the last comprehensive empirical evaluation of supervised learning was the Statlog Project in the early 90's. We present a large-scale empirical comparison between ten supervised learning methods: SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. We also examine the effect that calibrating the models via Platt Scaling and Isotonic Regression has on their performance. An important aspect of our study is

This paper presents results of a large-scale empirical comparison of ten supervised learning algorithms using eight performance criteria. We evaluate the performance of SVMs, neural nets, logistic regression, naive bayes, memory-based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps on eleven binary classification problems using a variety of performance metrics: accuracy, F-score, Lift, ROC Area, average precision, precision/recall break-even point, squared error, and cross-entropy. For each algorithm we examine common variations, and thoroughly explore the space of parameters. For example, we compare ten decision tree styles, neural nets of many sizes, SVMs with many kernels, etc.

Because some of the performance metrics we examine

11/6/14

Proceedings of the 23rd International Conference on Machine Learning (ICML '06).

13

# A study comparing Classifiers

## → 11 binary classification problems / 8 metrics

Table 2. Normalized scores for each learning algorithm by metric (average over eleven problems)

MODEL	CAL	ACC	FSC	LFT	ROC	APR	BEP	RMS	MXE	MEAN	OPT-SEL
BST-DT	PLT	.843*	.779	.939	.963	.938	.929*	.880	.896	.896	.917
RF	PLT	.872*	.805	.934*	.957	.931	.930	.851	.858	.892	.898
BAG-DT	—	.846	.781	.938*	.962*	.937*	.918	.845	.872	.887*	.899
BST-DT	ISO	.826*	.860*	.929*	.952	.921	.925*	.854	.815	.885	.917*
RF	—	.872	.790	.934*	.957	.931	.930	.829	.830	.884	.890
BAG-DT	PLT	.841	.774	.938*	.962*	.937*	.918	.836	.852	.882	.895
RF	ISO	.861*	.861	.923	.946	.910	.925	.836	.776	.880	.895
BAG-DT	ISO	.826	.843*	.933*	.954	.921	.915	.832	.791	.877	.894
SVM	PLT	.824	.760	.895	.938	.898	.913	.831	.836	.862	.880
ANN	—	.803	.762	.910	.936	.892	.899	.811	.821	.854	.885
SVM	ISO	.813	.836*	.892	.925	.882	.911	.814	.744	.852	.882
ANN	PLT	.815	.748	.910	.936	.892	.899	.783	.785	.846	.875
ANN	ISO	.803	.836	.908	.924	.876	.891	.777	.718	.842	.884
BST-DT	—	.834*	.816	.939	.963	.938	.929*	.598	.605	.828	.851
KNN	PLT	.757	.707	.889	.918	.872	.872	.742	.764	.815	.837
KNN	—	.756	.728	.889	.918	.872	.872	.729	.718	.810	.830
KNN	ISO	.755	.758	.882	.907	.854	.869	.738	.706	.809	.844
BST-STMP	PLT	.724	.651	.876	.908	.853	.845	.716	.754	.791	.808
SVM	—	.817	.804	.895	.938	.899	.913	.514	.467	.781	.810
BST-STMP	ISO	.709	.744	.873	.899	.835	.840	.695	.646	.780	.810
BST-STMP	—	.741	.684	.876	.908	.853	.845	.394	.382	.710	.726
DT	ISO	.648	.654	.818	.838	.756	.778	.590	.589	.709	.774

11/6/14

14

# Where are we ? →

## Three major sections for classification

- We can divide the large variety of classification approaches into roughly three major types

### 1. Discriminative

- directly estimate a decision rule/boundary
- e.g., logistic regression, support vector machine, **decisionTree**

### 2. Generative:

- build a generative statistical model
- e.g., naïve bayes classifier, Bayesian networks

### 3. Instance based classifiers

- Use observation directly (no models)
- e.g. K nearest neighbors

11/6/14

15

## A Dataset for classification

$X_1$	$X_2$	$X_3$	$C$

$$f : [X] \rightarrow [C]$$

Output as Discrete Class Label  
 $C_1, C_2, \dots, C_L$

- Data/points/examples/samples/records: [rows]
- Features/attributes/dimensions/independent variables/covariates/predictors/regressors: [columns, except the last]
- Target/outcome/response/label/dependent variable: special column to be predicted [last column]

11/6/14

16

# Example

- Example: Play Tennis

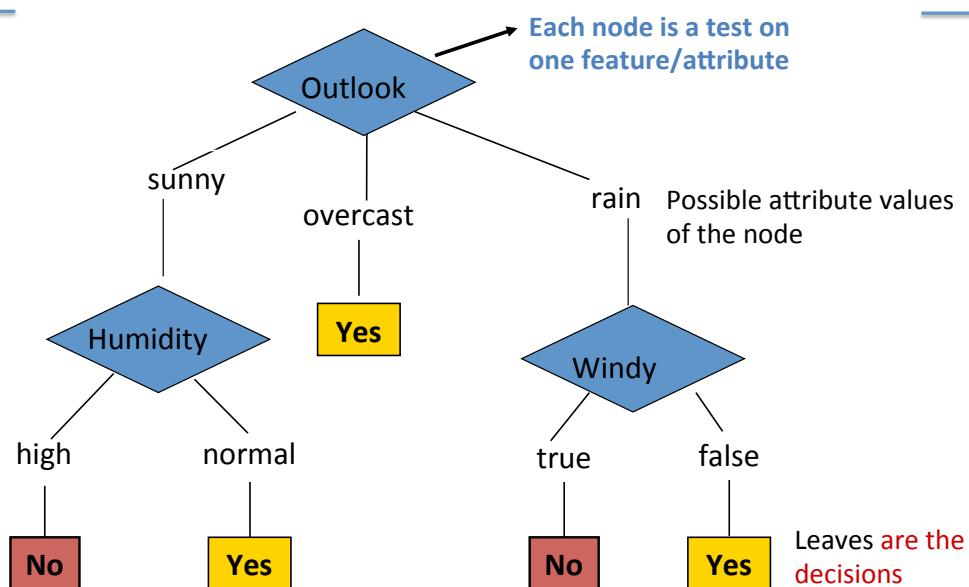
*PlayTennis*: training examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

11/6/14

17

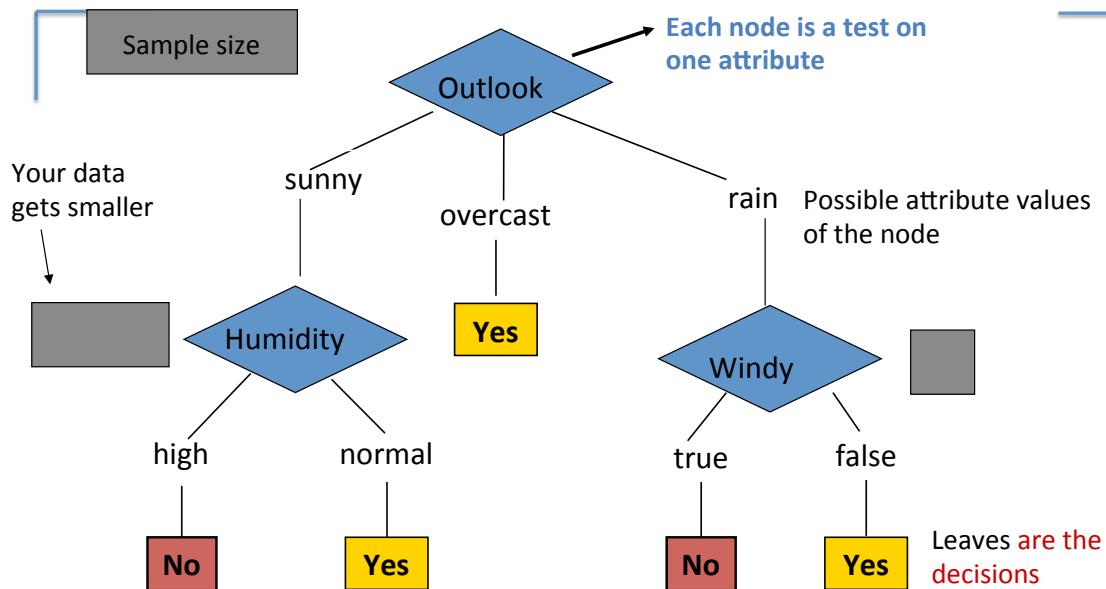
## Anatomy of a decision tree



11/6/14

18

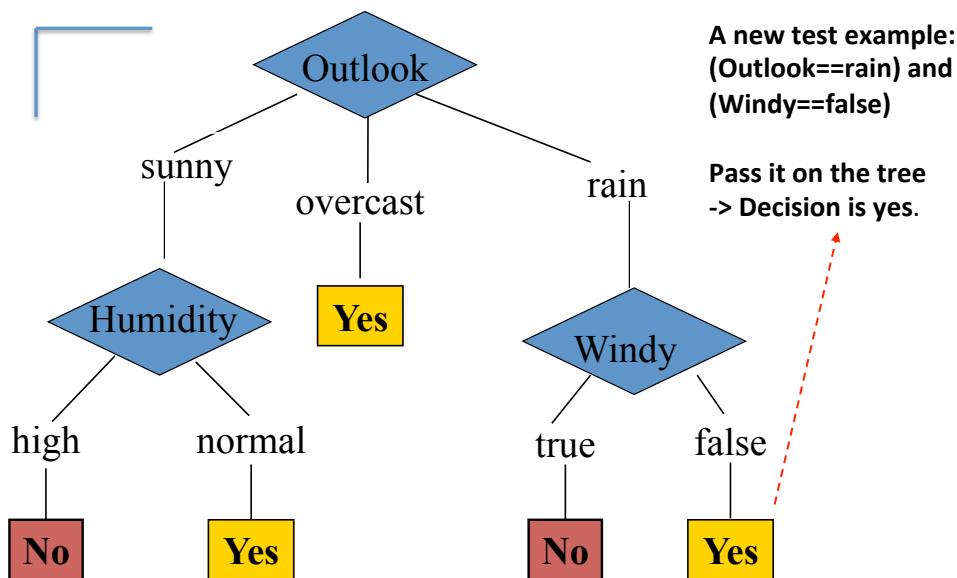
# Anatomy of a decision tree



11/6/14

19

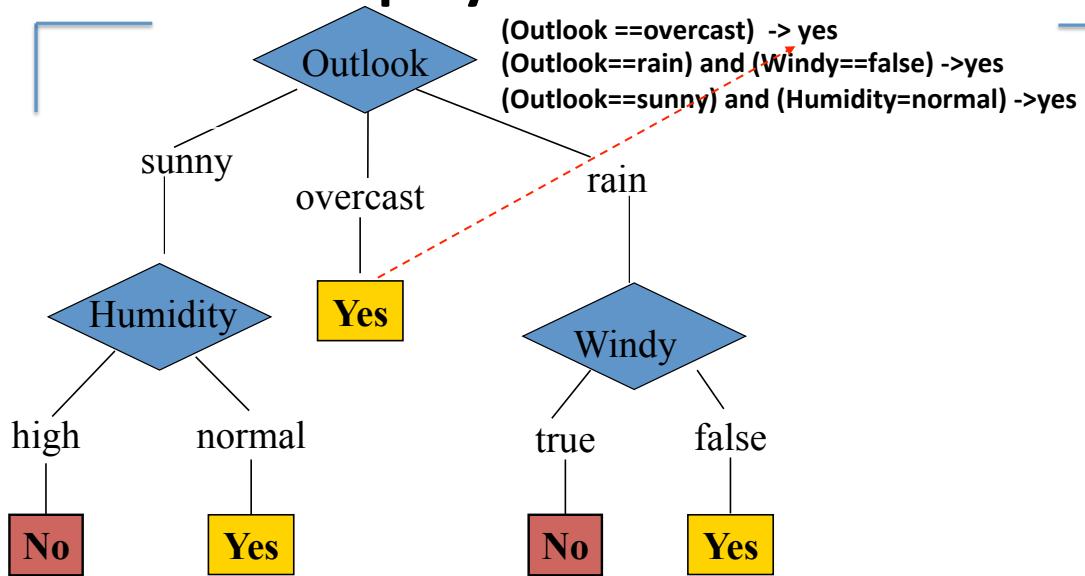
## Apply Model to Test Data: To ‘play tennis’ or not.



11/6/14

20

## Apply Model to Test Data: To ‘play tennis’ or not.



11/6/14

21

## Decision trees

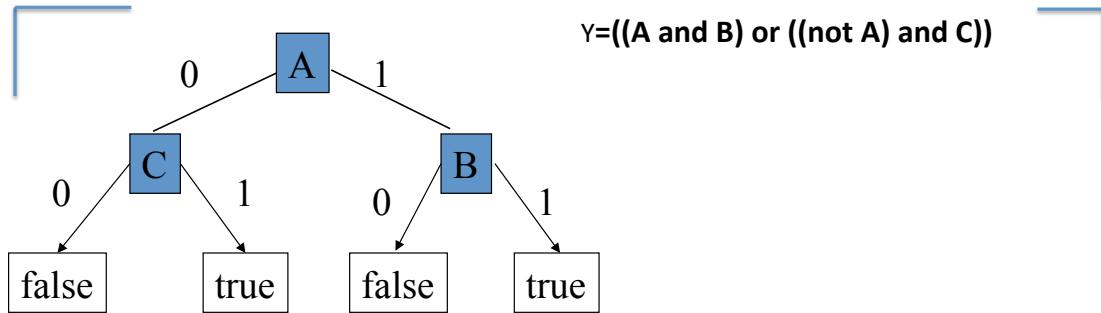
- Decision trees represent a disjunction of conjunctions of constraints on the attribute values of instances.

- (Outlook ==overcast)
- OR
- ((Outlook==rain) and (Windy==false))
- OR
- ((Outlook==sunny) and (Humidity=normal))
- => yes play tennis

11/6/14

22

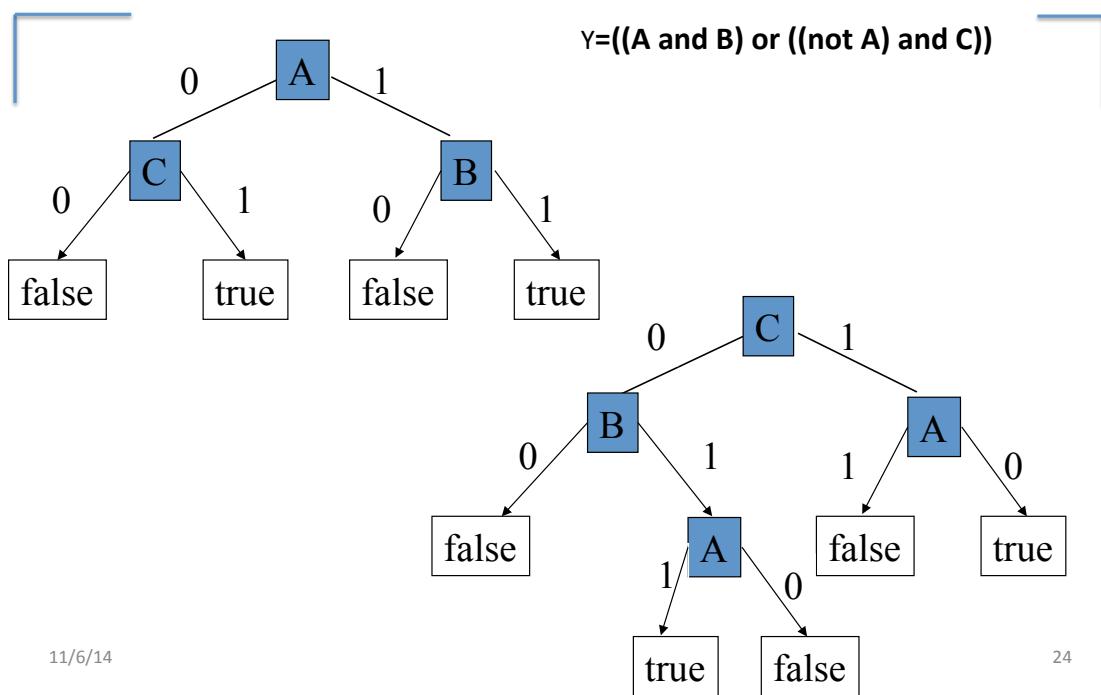
# Representation



11/6/14

23

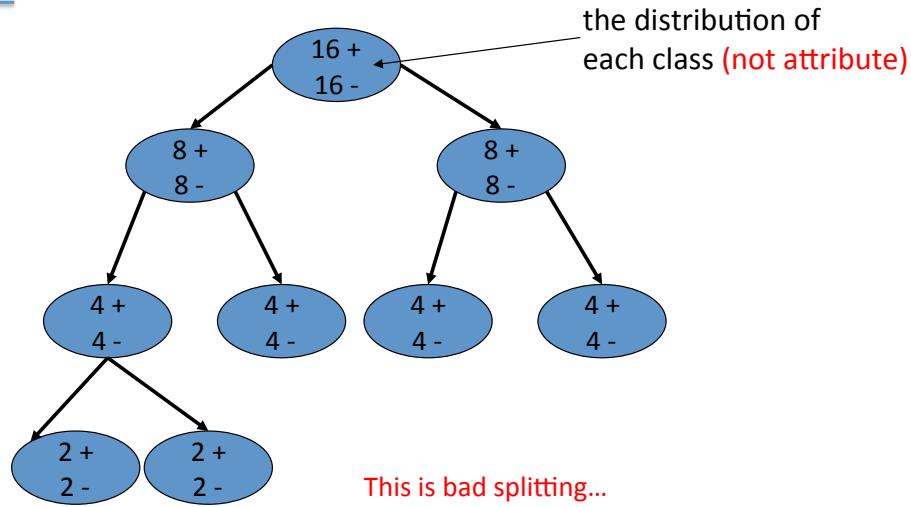
## Same concept / different representation



11/6/14

24

## Which attribute to select for splitting?



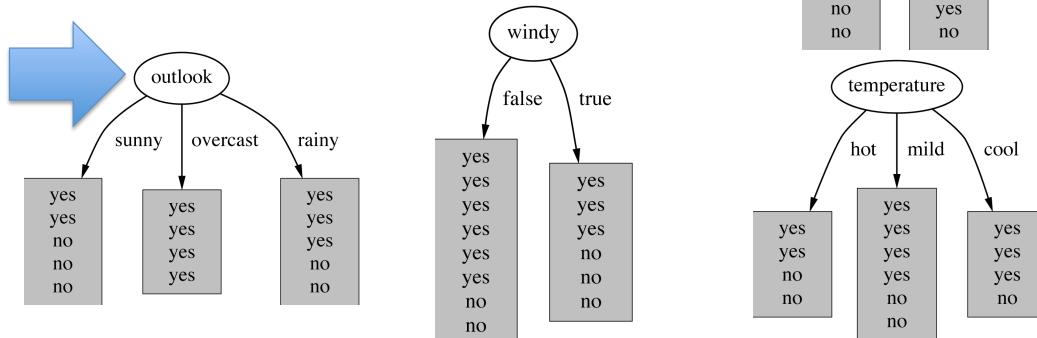
11/6/14

25

## How do we choose which attribute to split ?

Which attribute should be used as the test?

Intuitively, you would prefer the one that *separates* the training examples as much as possible.



11/6/14

26

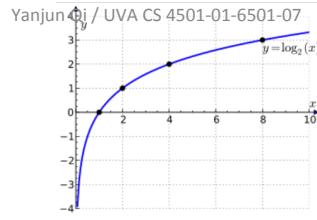
# Today

- Decision Tree (DT):
- Tree representation
- Brief information theory
- Learning decision trees
- Bagging
- Random forests: Ensemble of DT
- More about ensemble

Information gain is one criteria to decide on the attribute for splitting

- Imagine:
  - 1. Someone is about to tell you your own name
  - 2. You are about to observe the outcome of a dice roll
  - 2. You are about to observe the outcome of a coin flip
  - 3. You are about to observe the outcome of a biased coin flip
- Each situation have a different *amount of uncertainty* as to what outcome you will observe.

# Information

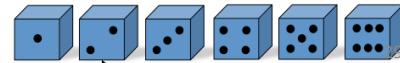


- Information:
- reduction in uncertainty (amount of surprise in the outcome)

$$I(E) = \log_2 \frac{1}{p(x)} = -\log_2 p(x)$$

If the probability of this event happening is small and it happens, the information is large.

- Observing the outcome of a coin flip →  $I = -\log_2 1/2 = 1$
- Observe the outcome of a dice is 6 →  $I = -\log_2 1/6 = 2.58$



11/6/14

Yanjun Qi / UVA CS 4501-01-6501-07

# Entropy

- The *expected amount of information* when observing the output of a random variable X

$$H(X) = E(I(X)) = \sum_i p(x_i)I(x_i) = -\sum_i p(x_i)\log_2 p(x_i)$$

If the X can have 8 outcomes and all are equally likely

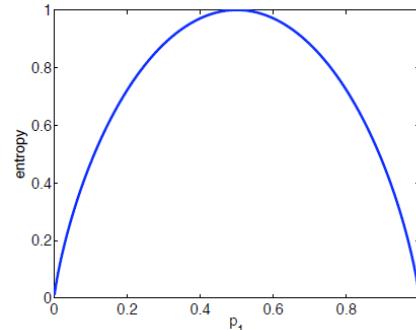
$$H(X) = -\sum_i 1/8 \log_2 1/8 = 3$$

# Entropy

- If there are  $k$  possible outcomes

$$H(X) \leq \log_2 k$$

- Equality holds when all outcomes are equally likely
- The more the probability distribution deviates from uniformity, the lower the entropy



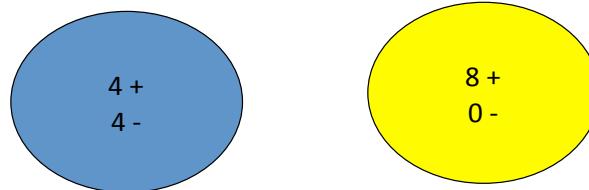
e.g. for a random binary variable

11/6/14

31

## Entropy Lower → better purity

- Entropy measures the purity



The distribution is less uniform  
Entropy is lower  
The node is purer

11/6/14

32

# Information gain

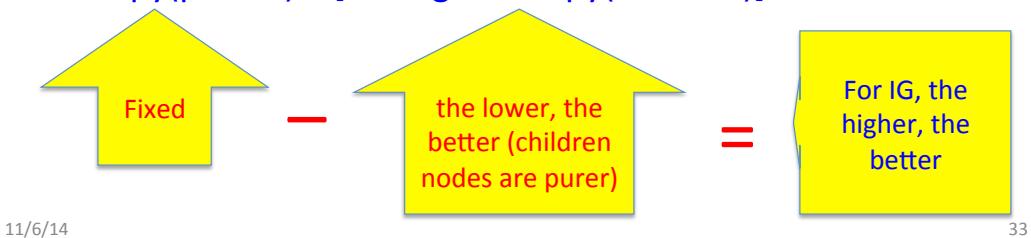
- $IG(X_i, Y) = H(Y) - H(Y|X_i)$

Reduction in uncertainty by knowing a feature  $X_i$

Information gain:

= (information before split) – (information after split)

= entropy(parent) – [average entropy(children)]



# Conditional entropy

$$H(Y) = - \sum_i p(y_i) \log_2 p(y_i)$$

$$H(Y | X) = \sum_j p(x_j) H(Y | X = x_j)$$

$$= - \sum_j p(x_j) \sum_i p(y_i | x_j) \log_2 p(y_i | x_j)$$

# Example

Attributes Labels

X1	X2	Y	Count
T	T	+	2
T	F	+	2
F	T	-	5
F	F	+	1

Which one do we choose

X1 or X2?

$$IG(X1, Y) = H(Y) - H(Y|X1)$$

$$H(Y) = -(5/10) \log(5/10) - 5/10 \log(5/10) = 1$$

$$\begin{aligned} H(Y|X1) &= P(X1=T)H(Y|X1=T) + P(X1=F)H(Y|X1=F) \\ &= 4/10 (1 \log 1 + 0 \log 0) + 6/10 (5/6 \log 5/6 + 1/6 \log 1/6) \\ &= 0.39 \end{aligned}$$

$$\text{Information gain } (X1, Y) = 1 - 0.39 = 0.61$$

11/6/14

35

## Which one do we choose?

X1	X2	Y	Count
T	T	+	2
T	F	+	2
F	T	-	5
F	F	+	1



X1	X2	Y	Count
T	T	+	2
T	F	+	2
F	T	-	5
F	F	+	1

One branch

The other branch

$$\text{Information gain } (X1, Y) = 0.61$$

$$\text{Information gain } (X2, Y) = 0.12$$

Pick the variable which provides  
the most information gain about Y

Pick X1

→ Then recursively choose next  $X_i$  on branches

11/6/14

36

# Decision Trees

- **Caveats:** The number of possible values influences the information gain.
  - The more possible values, the higher the gain (the more likely it is to form small, but pure partitions)
- **Other Purity (diversity) measures**
  - Information Gain
  - Gini (population diversity)  $\sum_{k=1}^K \hat{p}_{mk}(1 - \hat{p}_{mk})$ 
    - where  $p_{mk}$  is proportion of class k at node m
  - Chi-square Test

11/6/14

37

# Overfitting

- You can perfectly fit DT to any training data
- **Instability of Trees**
  - High variance (small changes in training set will result in changes of tree model)
  - Hierarchical structure → Error in top split propagates down
- **Two approaches:**
  - 1. Stop growing the tree when further splitting the data does not yield an improvement
  - 2. Grow a full tree, then prune the tree, by eliminating nodes.

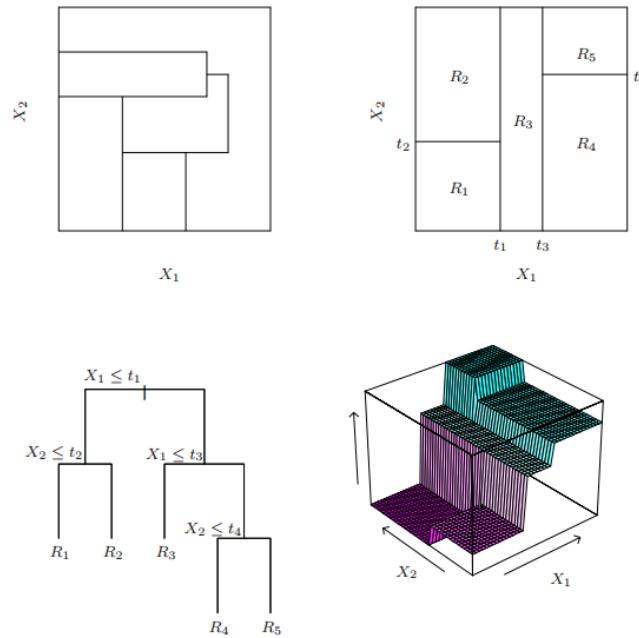
11/6/14

38

From ESL book Ch9 :

### Classification and Regression Trees (CART)

- **Partition feature space into set of rectangles**
- Fit simple model in each partition



**FIGURE 9.2.** Partitions and CART. Top right panel shows a partition of a two-dimensional feature space by recursive binary splitting, as used in CART, applied to some fake data. Top left panel shows a general partition that cannot be obtained from recursive binary splitting. Bottom left panel shows the tree corresponding to the partition in the top right panel, and a perspective plot of the prediction surface appears in the bottom right panel.

Yanjun Qi / UVA CS 4501-01-6501-07

## Summary: Decision trees

- Non-linear classifier
- Easy to use
- **Easy to interpret**
- Susceptible to overfitting but can be avoided.

# Today

- Decision Tree (DT):
  - Tree representation
  - Brief information theory
  - Learning decision trees
  - **Bagging**
  - Random forests: Ensemble of DT
  - More about ensemble

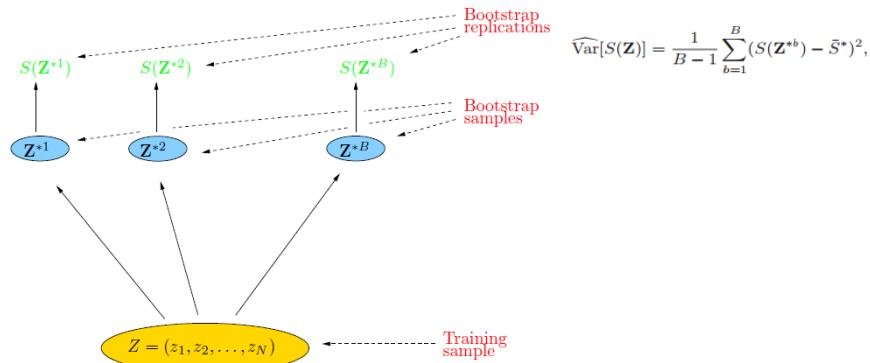
## Bagging

- Bagging or *bootstrap aggregation*
  - a technique for **reducing the variance** of an estimated prediction function.
- For instance, for classification, a **committee of trees**
  - Each tree casts a vote for the predicted class.

# Bootstrap

The basic idea:

randomly draw datasets *with replacement (i.e. allows duplicates)* from the training data, each sample *the same size as the original training set*



11/6/14

43

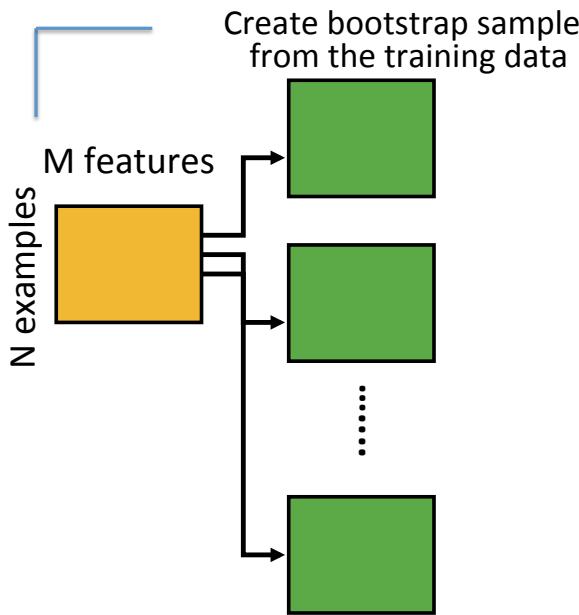
## With vs Without Replacement

- **Bootstrap with replacement** can keep the sampling size the same as the original size for every repeated sampling. The sampled data groups are independent on each other.
- **Bootstrap without replacement** cannot keep the sampling size the same as the original size for every repeated sampling. The sampled data groups are dependent on each other.

11/6/14

44

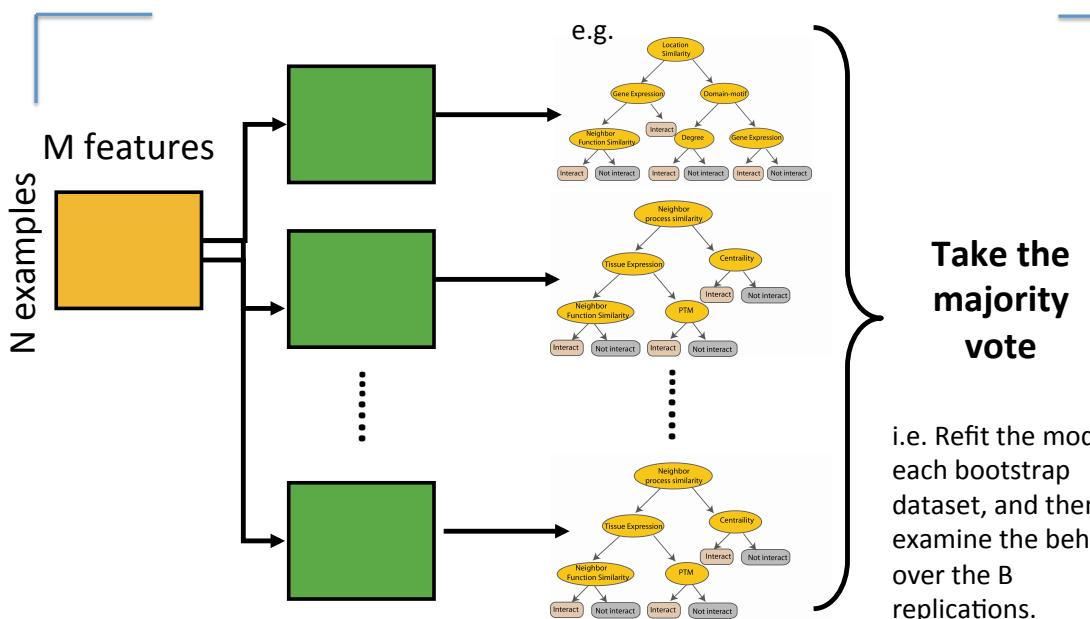
# Bagging



11/6/14

45

# Bagging of DT Classifiers



11/6/14

46

## Bagging for Classification with 0,1 Loss

- Classification with 0, 1 loss
  - Bagging a **good** classifier can make it **better**.
  - Bagging a **bad** classifier can make it **worse**.
  - Can understand the bagging effect in terms of a consensus of independent ***weak learners*** and ***wisdom of crowds***

11/6/14

47

## Peculiarities

- Model Instability is good when bagging
  - The more variable (unstable) the basic model is, the more improvement can potentially be obtained
  - Low-Variability methods (e.g. LDA) improve less than High-Variability methods (e.g. decision trees)
- Load of Redundancy
  - Most predictors do roughly “the same thing”

11/6/14

48

# Bagging : an simulated example

N = 30 training samples,

two classes and p = 5 features,

Each feature N(0, 1) distribution and pairwise correlation .95

Response Y generated according to:

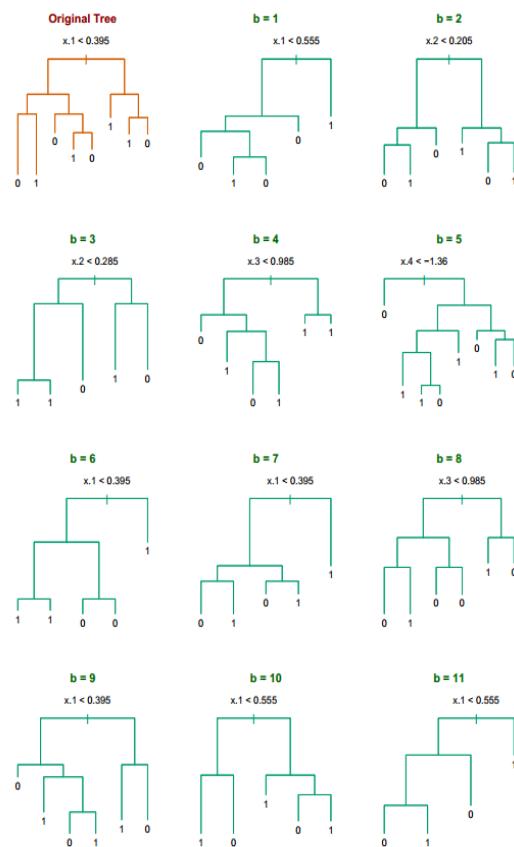
$$\Pr(Y = 1|x_1 \leq 0.5) = 0.2 \quad \Pr(Y = 1|x_1 > 0.5) = 0.8$$

Test sample size of 2000

Fit classification trees to training set and bootstrap samples

B = 200

Notice the bootstrap trees are different than the original tree

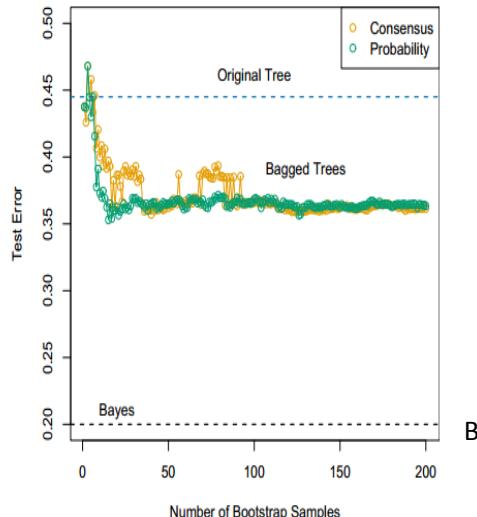


Five features highly correlated with each other

→ No clear difference with picking up which feature to split

→ Small changes in the training set will result in different tree

→ But these trees are actually quite similar for classification



→ For  $B > 30$ , more trees do not improve the bagging results

→ Since the trees correlate highly to each other and give similar classifications

Consensus: Majority vote

Probability: Average distribution at terminal nodes

ESL book / Example 8.7.1

## Bagging

- Slightly increases model space
  - Cannot help where greater enlargement of space is needed
  
- Bagged trees are correlated
  - Use random forest to reduce correlation between trees

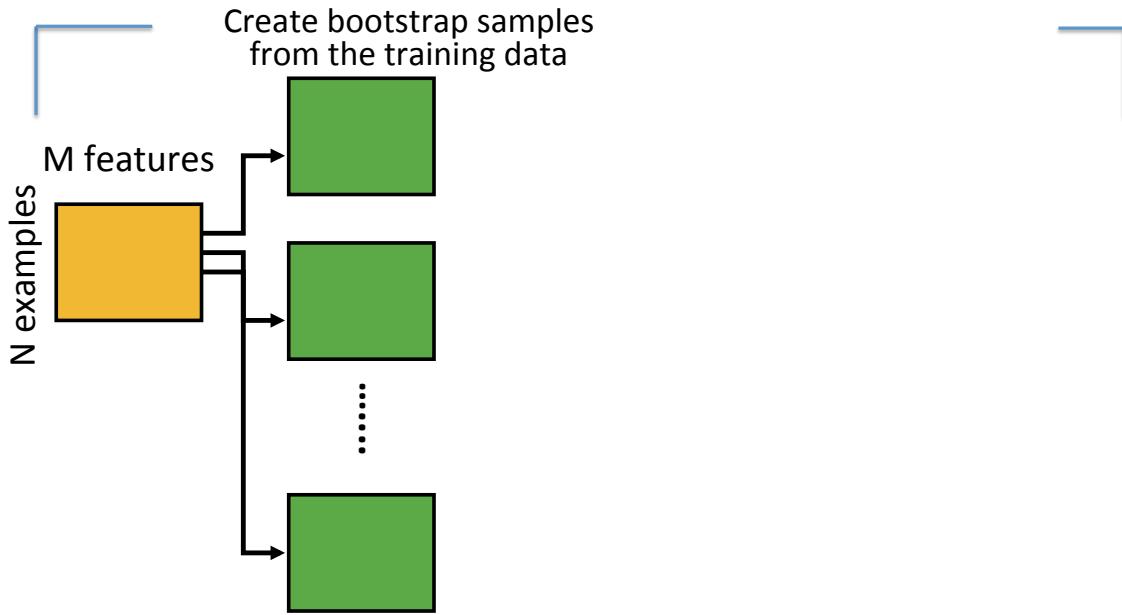
# Today

- Decision Tree (DT):
  - Tree representation
  - Brief information theory
  - Learning decision trees
  - Bagging
  - Random forests: special ensemble of DT
  - More about ensemble

## Random forest classifier

- Random forest classifier,
  - an extension to bagging
  - which *uses de-correlated trees*.

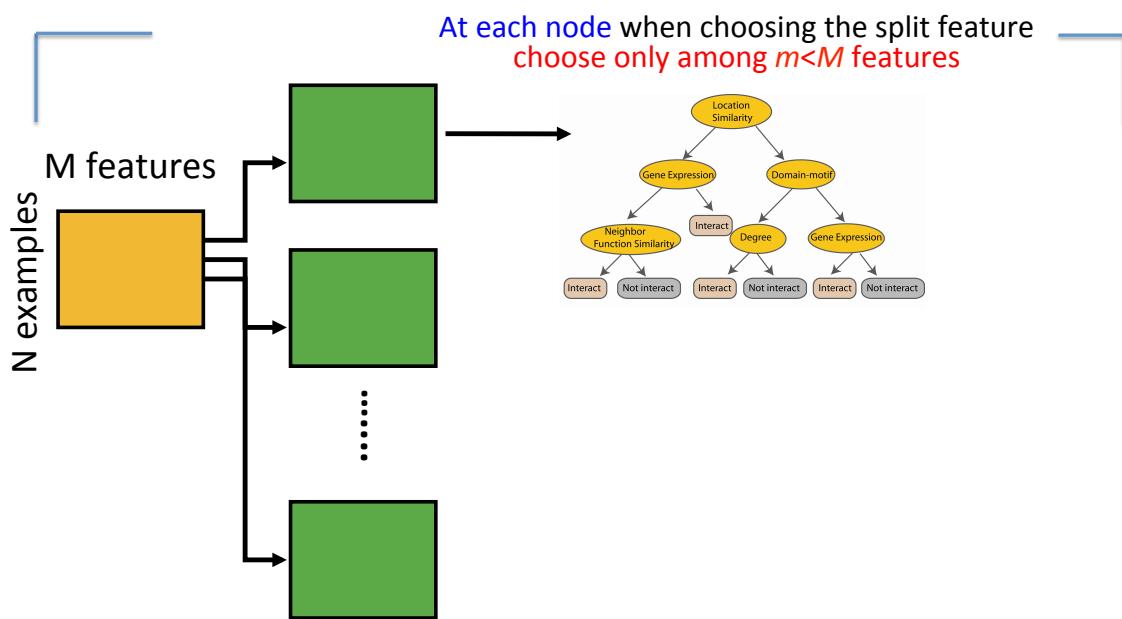
# Random Forest Classifier



11/6/14

55

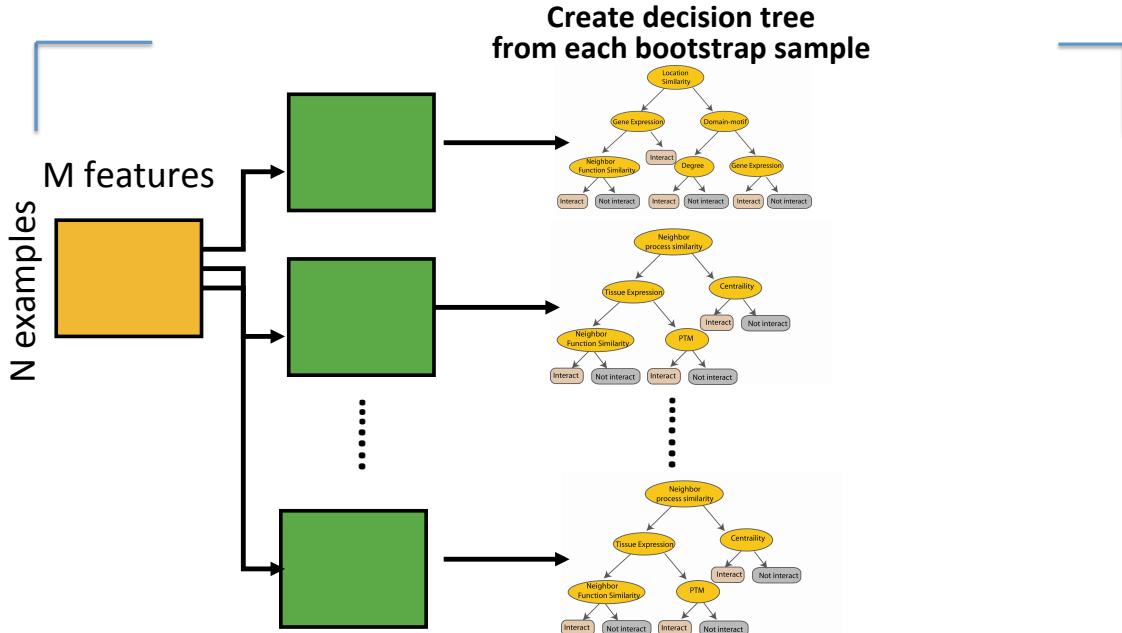
# Random Forest Classifier



11/6/14

56

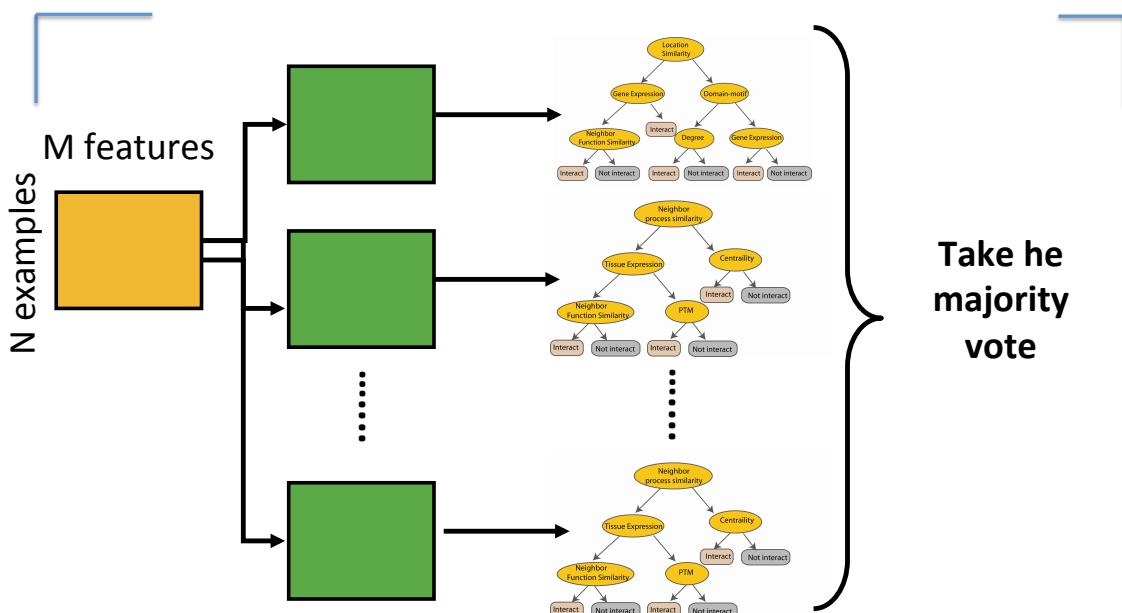
# Random Forest Classifier



11/6/14

57

# Random Forest Classifier



11/6/14

58

# Random Forests

For each of our  $B$  bootstrap samples

Form a tree in the following manner

Given  $p$  dimensions, pick  $m$  of them

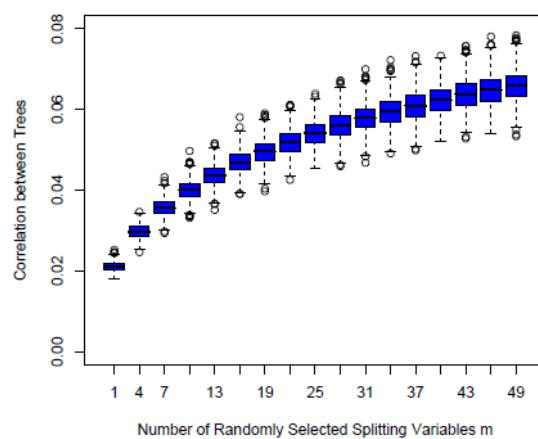
Split only according to these  $m$  dimensions

(we will NOT consider the other  $p-m$  dimensions)

Repeat the above steps i & ii for each split

Note: we pick a different set of  $m$  dimensions for each split  
on a single tree

Yanjun Qi / UVA CS 4501-01-6501-07



**FIGURE 15.9.** Correlations between pairs of trees drawn by a random-forest regression algorithm, as a function of  $m$ . The boxplots represent the correlations at 600 randomly chosen prediction points  $x$ .

## Random Forests

Random forest can be viewed as a refinement of bagging with a tweak of **decorrelating** the trees:

At each tree split, a random subset of  $m$  features out of all  $p$  features is drawn to be considered for splitting

Some guidelines provided by Breiman, but be careful to choose  $m$  based on specific problem:

$m = p$  amounts to bagging

$m = p/3$  or  $\log_2(p)$  for regression

$m = \sqrt{p}$  for classification

## Random Forests

Random Forests try to reduce correlation between the trees.

Why?

## Random Forests

Assuming each tree has variance  $\sigma^2$

If trees are independently identically distributed, then average variance is  $\sigma^2/B$

## Random Forests

Assuming each tree has variance  $\sigma^2$

If simply identically distributed, then average variance is

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2$$

As  $B \rightarrow \infty$ , second term  $\rightarrow 0$

Thus, the pairwise correlation always affects the variance

# Random Forests

How to deal?

If we reduce  $m$  (the number of dimensions we actually consider),

then we reduce the pairwise tree correlation

Thus, variance will be reduced.

Yanjun Qi / UVA CS 4501-01-6501-07

## Today

- Decision Tree (DT):
  - Tree representation
  - Brief information theory
  - Learning decision trees
  - Bagging
  - Random forests: Ensemble of DT
  - More ensemble

## e.g. Ensembles in practice



Oct 2006 - 2009

Each rating/sample:  
+ <user, movie, date of grade, grade>  
Training set (100,480,507 ratings)  
Qualifying set (2,817,131 ratings) → winner

- Training data is a set of users and ratings (1,2,3,4,5 stars) those users have given to movies.
- Predict what rating a user would give to any movie
- \$1 million prize for a 10% improvement over Netflix' s current method (MSE = 0.9514)

## Ensemble in practice

Team “Bellkor’s Pragmatic Chaos” defeated the team “ensemble” by submitting just 20 minutes earlier! → 1 million dollar !

Rank	Team Name	Best Test Score	% Improvement	Best Submit Time
<b>Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos</b>				
1	<a href="#">BellKor's Pragmatic Chaos</a>	0.8567	10.06	2009-07-26 18:18:28
2	<a href="#">The Ensemble</a>	0.8567	10.06	2009-07-26 18:38:22
3	<a href="#">Grand Prize Team</a>	0.8582	9.90	2009-07-10 21:24:40
4	<a href="#">Opera Solutions and Vandelay United</a>	0.8588	9.84	2009-07-10 01:12:31
5	<a href="#">Vandelay Industries !</a>	0.8591	9.81	2009-07-10 00:32:20
6	<a href="#">PragmaticTheory</a>	0.8594	9.77	2009-06-24 12:06:56
7	<a href="#">BellKor in BigChaos</a>	0.8601	9.70	2009-05-13 08:14:09
8	<a href="#">Dace</a>	0.8612	9.59	2009-07-24 17:18:43
9	<a href="#">Feeds2</a>	0.8622	9.48	2009-07-12 13:11:51
10	<a href="#">BigChaos</a>	0.8623	9.47	2009-04-07 12:33:59
11	<a href="#">Opera Solutions</a>	0.8623	9.47	2009-07-24 00:34:07
12	<a href="#">BellKor</a>	0.8624	9.46	2009-07-26 17:19:11

The ensemble team → blenders of multiple different methods

## References

- Prof. Tan, Steinbach, Kumar's "Introduction to Data Mining" slide
- Hastie, Trevor, et al. *The elements of statistical learning*. Vol. 2. No. 1. New York: Springer, 2009.
- Dr. Oznur Tastan's slides about RF and DT