Week 4

Agenda

1. Async review
2. Ensembling discussion
3. Bias/variance tradeoff discussion
4. Decision tree notebook
5. Time-permitting: Start case study

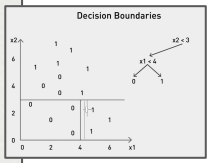
For next week: Read Breiman “Two Cultures” Paper

Quizzes as Warm-up

Decision trees can represent only linear decision boundaries.

True

False



Which of these are base cases for the decision tree algorithm?

Completely pure split of the data.

No data: default to parent's majority class.

All possible splits have zero entropy.

All of the above

Select the most fitting answer to fill in the blank in the first half of the sentence.
Use _____ with continuous features...

discrete

continuous

thresholds

algorithms

Select the most fitting answer to fill in the blank in the second half of the sentence.
and _____ for features with many attributes.

chaining

binarization

symbology

properties

There are typically two points of randomness in the generation of a decision forest. What are they?

The first point of randomness is

The second point of randomness is

Decision Trees

Tree Algorithm

GrowTree (S):

if y == 0 for all $< x, y >$ in S: return new leaf (0)

else if y == 1 for all $< x, y >$ in S: return new leaf (1)

else:

choose best attribute x_j

S_0 = all $< x, y >$ in S with x_j==0

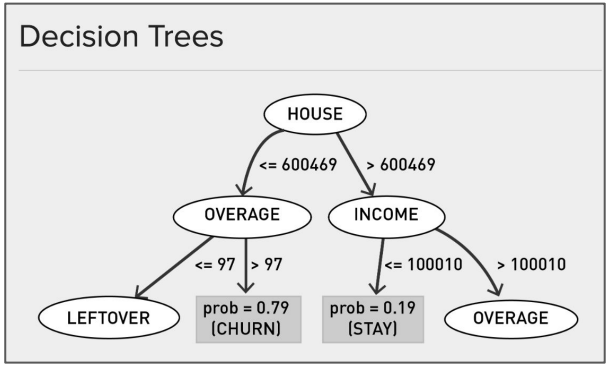
S_1 = all $< x, y >$ in S with x_j==1

return new node $(x_j$, GrowTree (S_0), GrowTree $(S_1))$

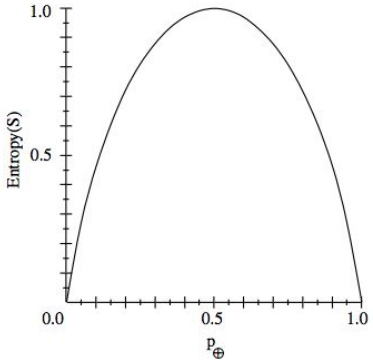
- Goal: Find a tree that is consistent with training examples.
- Strategy: Recursively choose most significant attribute as root of subtree.

Compare to KNN, Naive Bayes--

1. Describe the algorithm for training decision trees?
2. What's the training complexity?
   a. Can it be parallelized?
3. What's the prediction complexity?
4. Why is a tree sometimes said to be a 'white box'?
5. How do you output probabilities?
6. Is it an online learner?



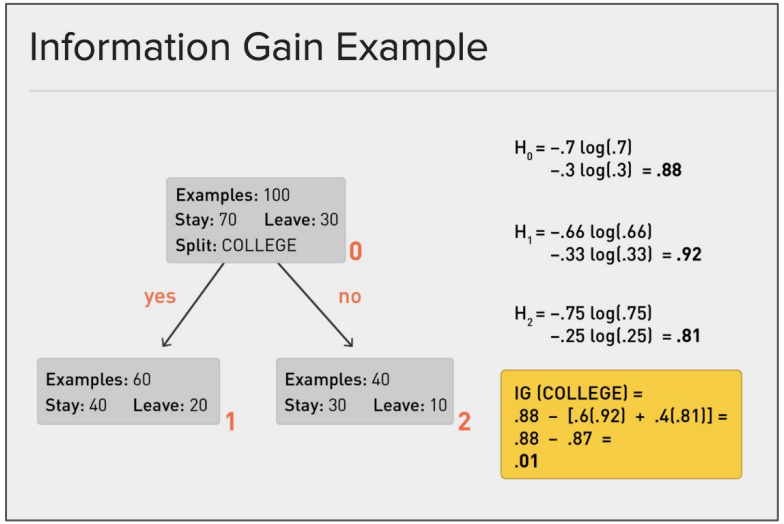
Decision Trees



- S is a sample of training examples
- $p_\oplus$ is the proportion of positive examples in S
- $p_\ominus$ is the proportion of negative examples in S
- Entropy measures the impurity of S

$$Entropy(S) \equiv -p_\oplus \log_2 p_\oplus - p_\ominus \log_2 p_\ominus$$

1. What is feature selection?
2. How is entropy used for feature selection in decision trees?
3. Can it be used for feature selection with other algorithms (e.g. Naive Bayes)?



Decision Trees

Hypothesis Space

How many distinct decision trees over n binary variables are there?

- Equal to number of binary functions over *n* attributes.
- Equal to number of distinct truth tables with 2^n rows.
- Equal to $2 \wedge (2^n)$.
  - E.g., six attributes: 18,446,744,073,709,551,616 trees

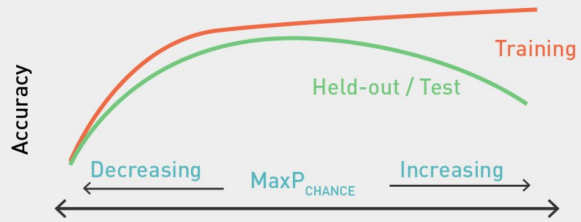
Very expressive hypothesis space with more room for variability

- Increases the chance that target function can be expressed.
- Increases number of hypotheses consistent with training data.
- Lower bias can lead to better predictions.
- Higher variance makes it possible to get worse predictions.

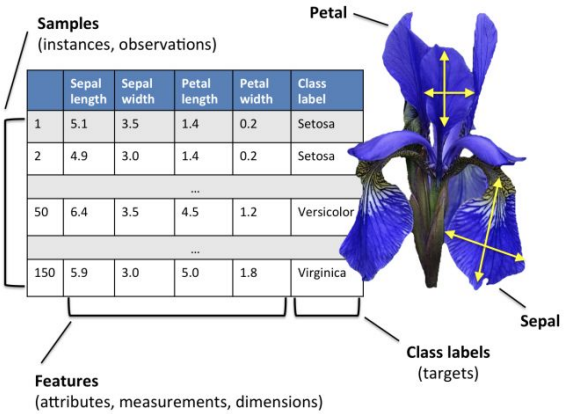
Representation Review

1. What is 'representation'?
2. What is a hypothesis space?
3. What does is mean to say that MaxPchance is a hyperparameter?

Regularization: Diagram



Ensembling



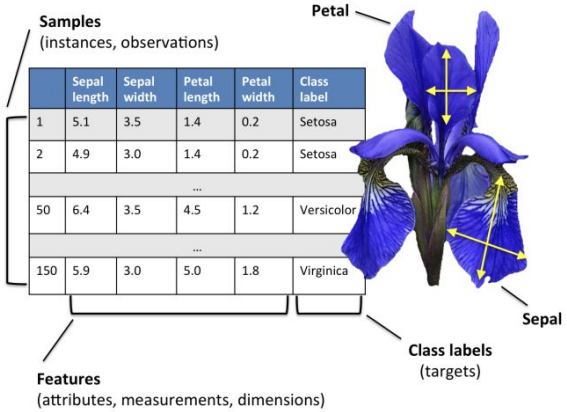
	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

Random Forest Algorithm

- Generate artificial training set through bootstrapping with replacement.
- Build decision tree.
  - Randomly choose subset of features; consider those as split points.
  - No pruning!
- Repeat process to create multiple trees.
- Run test case through all trees.
- Predict by taking vote among trees.

Bagging

1. What is bagging?
2. What is the intuition as to why bagging is effective?
3. What is the complexity of training a bagging classifier? Can it be parallelized?
4. Why use max-depth trees?
5. What are different ways to combine predictions?



	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

Boosting Classifier: AdaBoost Algorithm

1. Set weight for each training example = 1/n.
2. Train a classifier where objective respects the weights.
   - Run classifier over training examples.
3. Reduce weights for correct examples; increase weights for misclassified examples.
4. Return to step 2.
   - Second classifier is trained with objective that respects importance weights placed on each feature.

Boosting

1. What is boosting?
2. In terms of implementation, how do you incorporate example weights into the training of a tree?
3. How would you describe the complexity of training a boosting classifier? Is it parallelizable?
4. What are the pros/cons of using shallow trees or even stumps?

Bias and Variance Errors

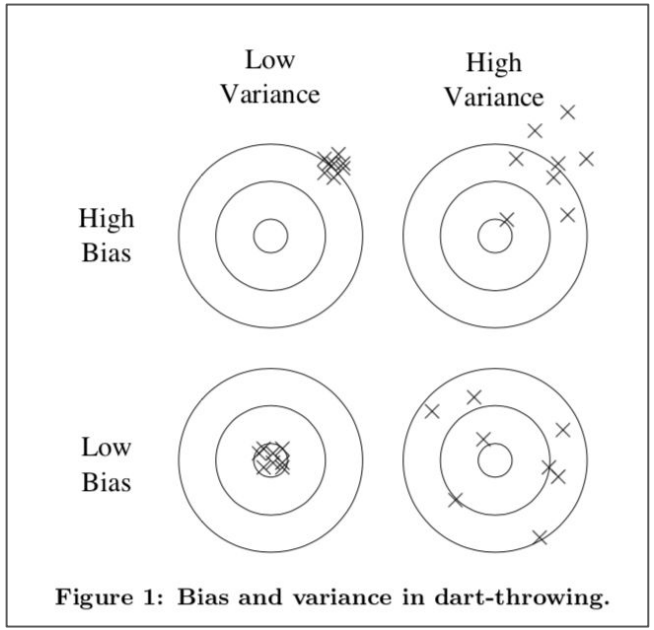
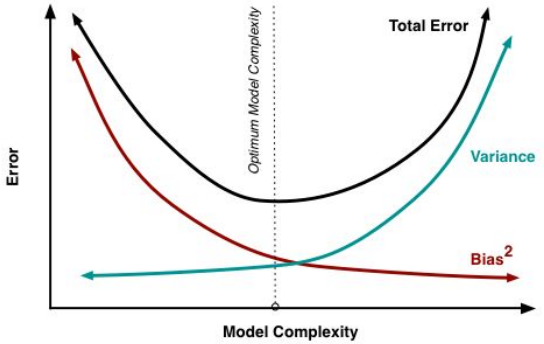


Figure 1: Bias and variance in dart-throwing.

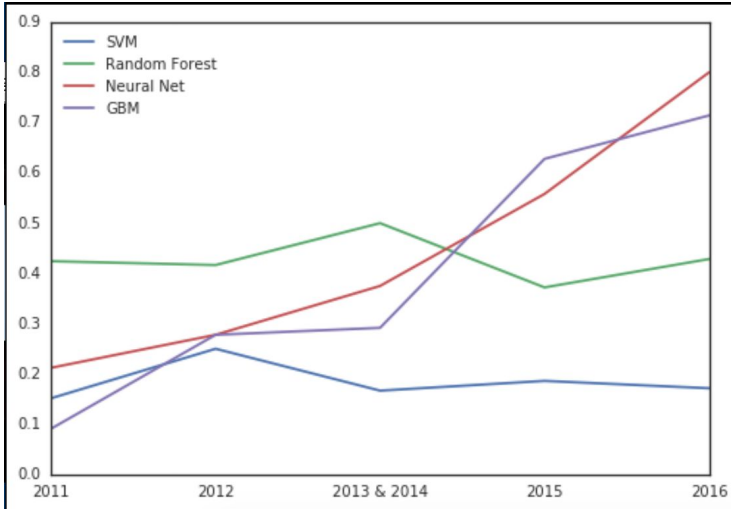
Bias / Variance Tradeoff

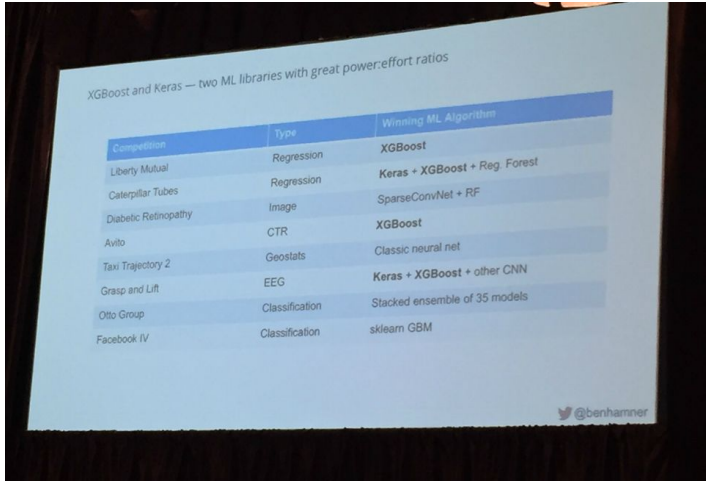
1. What do bias and variance intuitively refer to?
2. Why is there typically a tradeoff?
3. When can boosting or bagging reduce variance?
4. When can boosting or bagging reduce bias?
5. Then...is there still a tradeoff?



Trees in the Wild

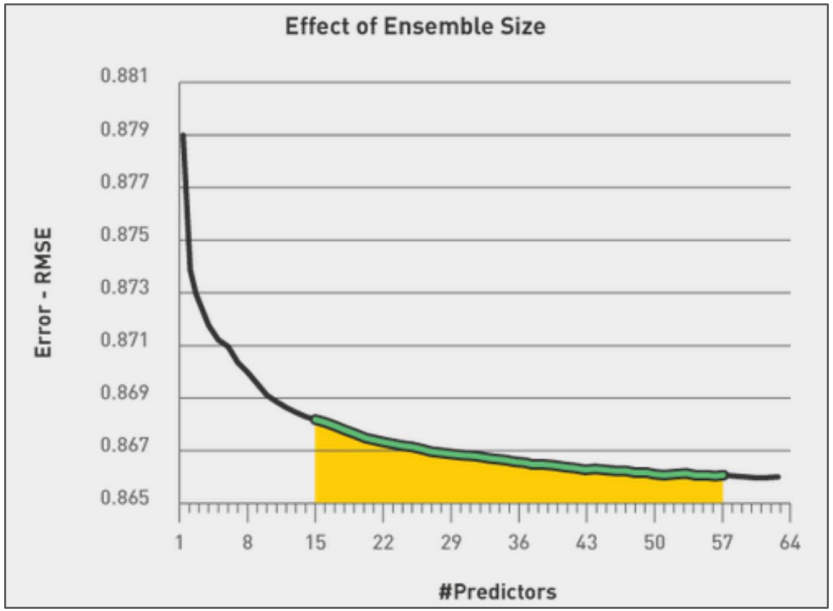
We evaluate **179 classifiers** arising from **17 families** (discriminant analysis, Bayesian, neural networks, support vector machines, decision trees, rule-based classifiers, boosting, bagging, stacking, random forests and other ensembles, generalized linear models, nearest-neighbors, partial least squares and principal component regression, logistic and multinomial regression, multiple adaptive regression splines and other methods), implemented in Weka, R (with and without the caret package), C and Matlab, including all the relevant classifiers available today. We use **121 data sets**, which represent **the whole UCI** data base (excluding the large-scale problems) and other own real problems, in order to achieve significant conclusions about the classifier behavior, not dependent on the data set collection. **The classifiers most likely to be the bests are the random forest** (RF) versions, the best of which (implemented in R and accessed via caret) achieves 94.1% of the maximum accuracy overcoming 90% in the 84.3% of the data sets. However, the difference is not statistically significant with the second best, the SVM with Gaussian kernel implemented in C using LibSVM, which achieves 92.3% of the maximum accuracy. A few models are clearly better than the remaining ones: random forest, SVM with Gaussian and polynomial kernels, extreme learning machine with Gaussian kernel, C5.0 and avNNet (a committee of multi-layer perceptrons implemented in R with the caret package). The random forest is clearly the best family of classifiers (3 out of 5 bests classifiers are RF), followed by SVM (4 classifiers in the top-10), neural networks and boosting ensembles (5 and 3 members in the top-20, respectively).





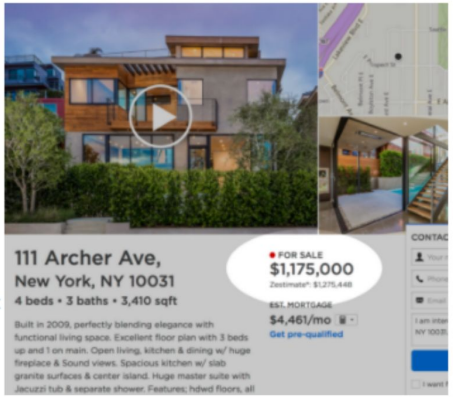
datascience@berkeley

Ensembles in the Wild



Zillow's Zestimate home valuation has shaken up the U.S. real estate industry since first released 11 years ago.

A home is often the largest and most expensive purchase a person makes in his or her lifetime. Ensuring homeowners have a trusted way to monitor this asset is incredibly important. The Zestimate was created to give consumers as much information as possible about homes and the housing market, marking the first time consumers had access to this type of home value information at no cost.



"Zestimates" are estimated home values based on 7.5 million statistical and machine learning models that analyze hundreds of data points on each property. And, by continually improving the median margin of error (from 14% at the onset to 5% today), Zillow has since become established as one of the largest, most trusted marketplaces for real estate information in the U.S. and a leading example of impactful machine learning.

Final Thoughts?