Random forests / decision forests

Cynthia Rudin

Machine Learning Course, Duke

Random forests / decision forests

(Ho Tin-kam, 1995)
(Leo Breiman, 2001)

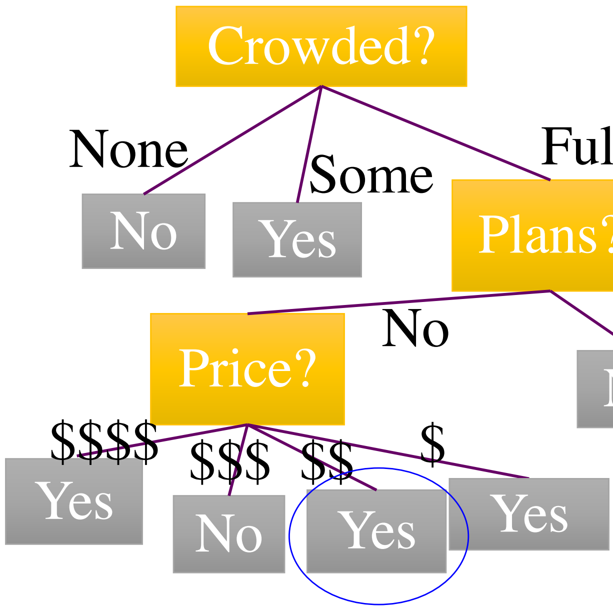
- Complex and powerful prediction tool
- Black-box
- Uses a simple but powerful idea: if you average many different yet accurate models, it reduces variance.

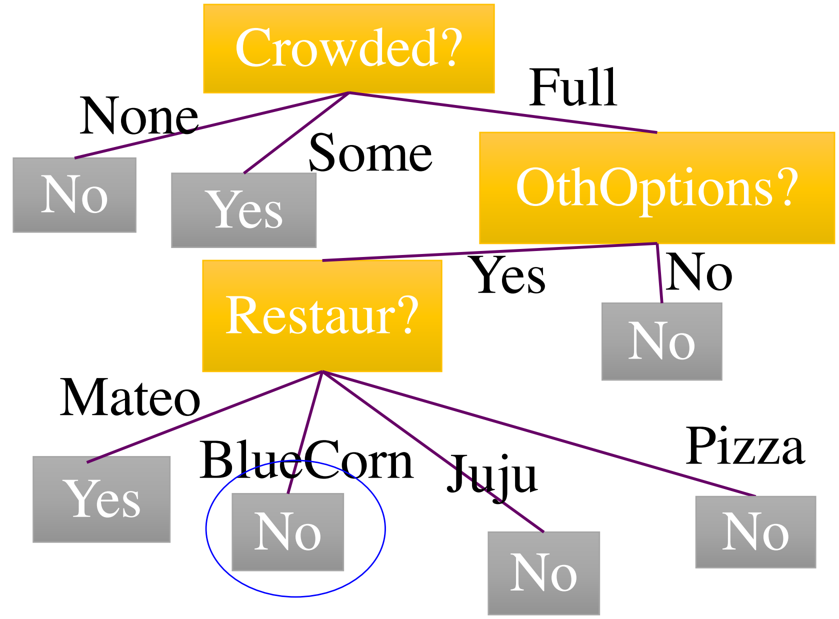
Bagging (Bootstrap Aggregating)

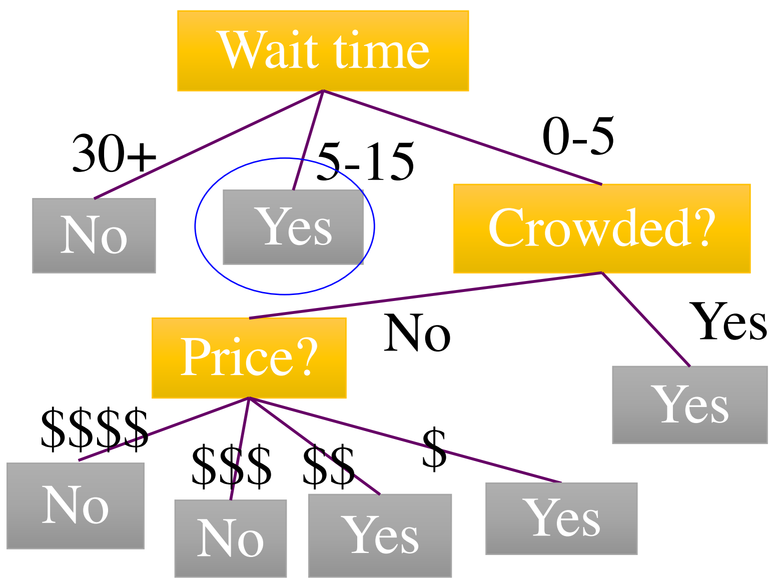
- Sample n points from the training set *with replacement*, grow a tree from them.
- Average the trees together to get the final prediction

- Example: Will the customer wait for a table at a restaurant?
  - OthOptions: Other options, True if there are restaurants nearby.
  - Weekend: This is true if it is Friday, Saturday or Sunday.
  - Area: Does it have a bar or other nice waiting area to wait in?
  - Plans: Does the customer have plans just after dinner?
  - Price: This is either $, $$, $$$, or $$$$
  - Precip: Is it raining or snowing?
  - Genre: French, Mexican, Thai, or Pizza
  - Wait: Wait time estimate: 0-5 min, 5-15 min, 15+
  - Crowded: Whether there are other customers (no, some, or full)

Credit: Adapted from Russell and Norvig







New observation:
BlueCorn, \$\$, Full, 5-15 min
No plans, Other options yes

Majority Vote: Yes

Decision Forests

For t=1 to T:

- Draw a bootstrap sample of size n from the training data.
- Grow a tree ($tree_t$) using this splitting and stopping procedure:
    - For this split, choose m features at random (out of p)
    - Evaluate the splitting criteria on all of them
    - Split on the best feature
    - If the node has less than n_{min} then stop splitting.

Output all the trees.

To predict on a new observation x, use the majority vote of the trees on x.

Decision Forests

Make trees diverse, which reduces variance

Comparison with decision trees:

- Bootstrap resamples
- Splitting considers only m possible (randomly chosen) features
- No pruning
- Majority vote of several trees is used to make predictions

Make trees fit more tightly, reduces bias

Variable Importance / Model Reliance

- How much does a model f rely on a variable?

Model Reliance(f, j)

$$= \text{Error}(f, \mathit{data}^{\mathit{scramble}\ j}) - \text{Error}(f, \mathit{data})$$

Reorder me!

$$\begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}$$

Decision Forests: Measuring Variable Importance

- Let us measure the "importance" of variable j.
- Take the data not used to construct $tree_t$. Call it "out-of-bag", OOB_t.
- Compute $error_t$ of model $tree_t$ on data OOB_t.
- Now randomly permute only the j^{th} feature values.

Decision Forests: Measuring Variable Importance

- Let us measure the "importance" of variable j.
- Take the data not used to construct tree_t. Call it "out-of-bag", OOB_t.
- Compute error_t of model tree_t on data OOB_t.
- Now randomly permute only the j^{th} feature values.

Reorder me!

$$\text{OOB} \longrightarrow \begin{pmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{pmatrix}$$

Decision Forests: Measuring Variable Importance

- Let us measure the "importance" of variable j.
- Take the data not used to construct tree_t. Call it "out-of-bag", OOB_t.
- Compute error_t of model tree_t on data OOB_t.
- Now randomly permute only the j^{th} feature values.

Reorder me!

$$\begin{pmatrix} x_{11} & x_{32} & x_{13} \\ x_{21} & x_{12} & x_{23} \\ x_{31} & x_{22} & x_{33} \end{pmatrix}$$

Decision Forests: Measuring Variable Importance

- Let us measure the "importance" of variable j.
- Take the data not used to construct tree_t. Call it "out-of-bag", OOB_t.
- Compute error_t of model tree_t on data OOB_t.
- Now randomly permute only the j^{th} feature values. Call this $\text{OOB}_{t,\text{permuted}}$.
- Compute $\text{error}_{t,\text{permuted}}$, using model tree_t on data $\text{OOB}_{t,\text{permuted}}$.
- The "raw importance" of variable j is then the average over trees of the difference: $$\frac{1}{T}\sum_{\text{trees } t}\left(\text{error}_{t,\text{ permuted}} - \text{error}_t\right)$$

Decision Forests: Measuring Variable Importance

- General notion of importance of a variable for a model.
- Specialized version for decision forests, where it is computed on out-of-bootstrap sample.

Decision Forests for Regression

For t=1 to T:

- Draw a bootstrap sample of size n from the training data.
- Grow a tree ($tree_t$) using this splitting and stopping procedure:
  - For this split, choose m features at random (out of p)
  - Evaluate the splitting criteria on all of them
  - Split on the best feature
  - If the node has less than n_{min} then stop splitting.

Output all the trees.

To predict on a new observation x, use the average vote of the trees on x.

Decision Forests

Advantages

- Complex and powerful prediction tool, highly nonlinear
- Has notion of variable importance

Disadvantages

- Black-box
- Tends to overfit unless tuned carefully (not always intuitive with the R package)
- Slow