Boruta – A System for Feature Selection

Miron B. Kursa, Aleksander Jankowski, Witold R. Rudnicki

Brandon Sherman

December 1, 2016

Outline

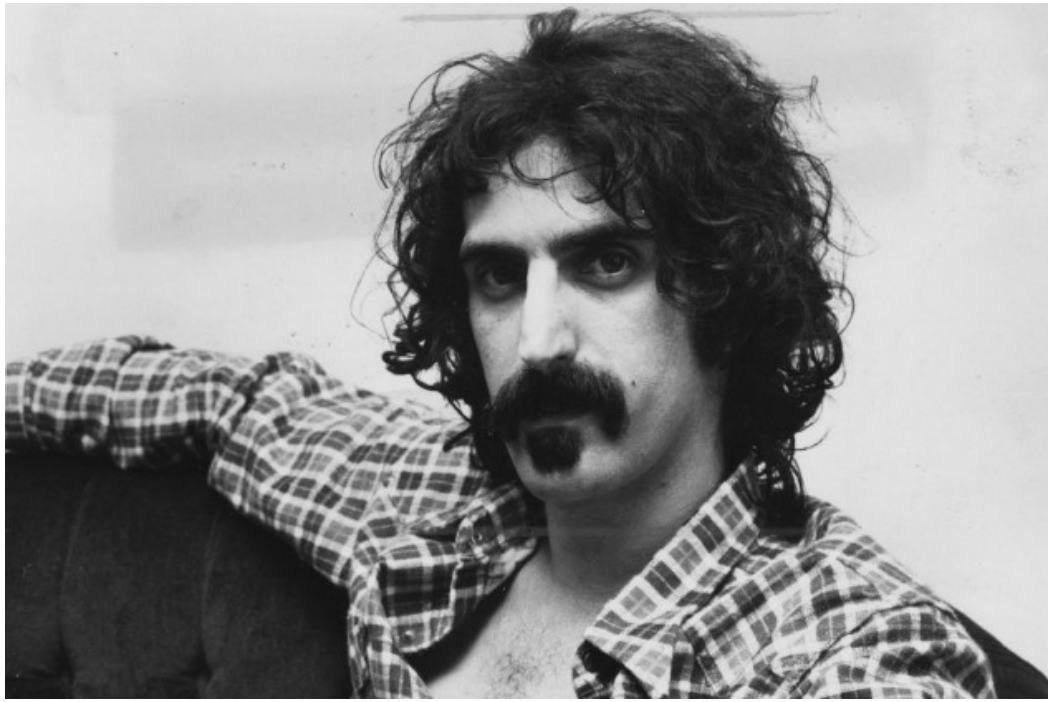
Red deer; Białowieża Forest - Podlaskie Voivodeship, Poland

How do random forests work?

- A **random forest** is a machine learning classifier that works as follows:
  1. For a dataset $(\mathbf{X}, \mathbf{Y})$ with observations $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ and response $\{y_1, y_2, \ldots, y_N\}$ $(y_i \in \{0, 1\})$, bootstrap B datasets of size N.
     - **Bootstrapping** - for a dataset $(\mathbf{X}, \mathbf{Y})$ with N observations, generate another dataset $(\mathbf{X}_b, \mathbf{Y}_b)$ with N observations by sampling from $(\mathbf{X}, \mathbf{Y})$ *with replacement*.
  2. For $b = \{1, \ldots, B\}$, train a decision tree on dataset $(\mathbf{X}_b, \mathbf{Y}_b)$.
  3. Predict a new observation $\mathbf{X}^{new}$ on each decision tree by taking the majority vote from all of the trees.
  4. Calculate the **out-of-bag error**, which is the mean prediction error for each decision tree's predictions on observations not included in its bootstrapped sample.
  5. Calculate feature importance for feature $\mathbf{X}_j$ by fitting a model on a randomly permuted $\mathbf{X}_j$ (called $\mathbf{X}_j^{(s)}$), and comparing its out-of-bag error to the out of bag error for $\mathbf{X}_j$.

Problems with Feature Importances

"Feature importances can't tell you the emotional story. They can give you the exact mathematical design, but what's missing is the eyebrows." – Frank Zappa (heavily paraphrased)



Problems with Feature Importances

- Vague, nebulous notion of "how important is this feature?"
- Feature importances are relative, so no notion of "how important is this feature on its own?"
- No idea how many features are needed.
- Is a feature with small importance unimportant or slightly important?
- The "Breiman assumption", which states that feature importance is normally distributed, is false.

Outline

Boruta (Slavic forest spirit); artist's depiction

A Quick Diversion

- Fire stock broker Dan Aykroyd and hire untrained homeless man Eddie Murphy
- Can Eddie Murphy be as good a stockbroker as Dan Aykroyd?
- If so, then there's nothing inherent about Dan Aykroyd that makes him a good stockbroker

Boruta Algorithm

- For each feature $\mathbf{X}_j$, randomly permute it to generate a "shadow feature" (random attribute) $\mathbf{X}_j^{(s)}$.
- Fit a random forest to the original and the shadow features
- Calculate feature importances on original and shadow features
- The feature is important for a single run if its importance is higher than the maximum importance of all shadow features (*MIRA*).
- Eliminate all features whose importances across all runs are low enough. Keep all features whose importances across all runs are high enough.
- Repeat from the beginning with all tentative features.

Boruta – Now with more math! (Part 1)

Iterate the following procedure N times for all original features $\{\mathbf{X}_1, \ldots, \mathbf{X}_p\}$:

1. Create a random forest consisting of original and *newly-generated* shadow features.
2. Calculate all $\text{Imp}(\mathbf{X}_j)$ and *MIRA*
3. If a particular $\text{Imp}(\mathbf{X}_j) > \textit{MIRA}$, then increment H_j and call $\mathbf{X}_j$ *important* for the run.

Boruta – Now with more math! (Part 2)

Once $\{H_1, \ldots, H_p\}$ have been calculated:

1. Perform the statistical test $H_0 : H_i = E(H)$ vs. $H_1 : H_i \neq E(H)$.
   - Because hits follow a binomial distribution, we have

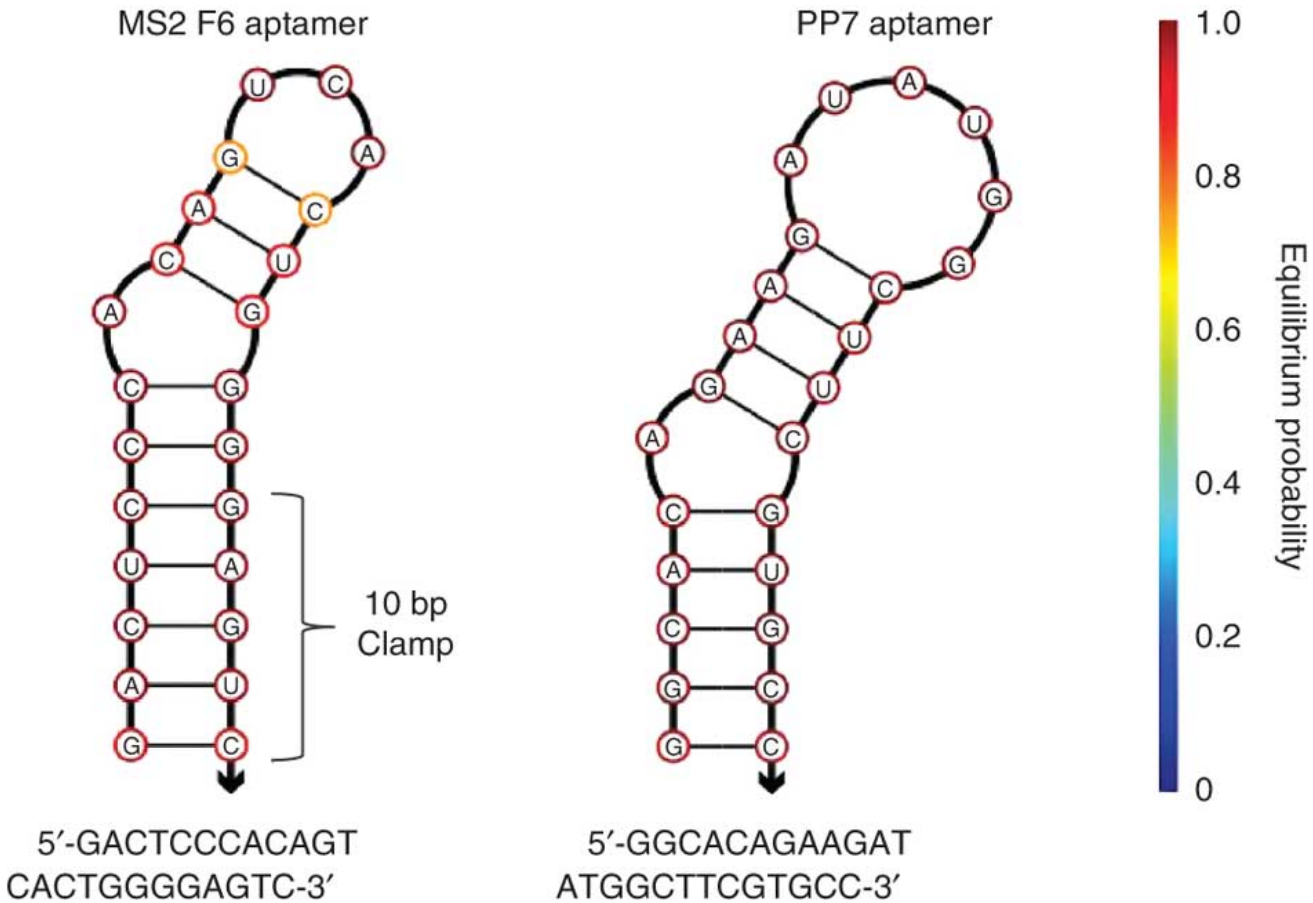
$$H_i \approx N\left((0.5N), (\sqrt{0.25N})^2\right).$$

2. If H_i is significantly *greater* than $E(H)$, then we say the feature is important.
3. If H_i is significantly *lower* than $E(H)$, then we say the feature is unimportant.
4. Finish the procedure after some number of iterations or if all features have been rejected or deemed important. Otherwise, repeat the procedure from the beginning on all tentative features.

Outline

The Big Question

Which DNA sequences are indicitative of aptamers?


MS2 F6 aptamer
PP7 aptamer
10 bp
Clamp
1.0
0.8
0.6
0.4
0.2
0
Equilibrium probability
5′-GACTCCCACAGT
CACTGGGGAGTC-3′
5′-GGCACAGAAGAT
ATGGCTTCGTGCC-3′

The Aptamer Problem

- An **aptamer** is an RNA or DNA chain that strongly binds to various molecular targets.
- An aptamer is represented as a genetic sequence that contains certain k-mers.
  - A, GC, TGA, and AGAC are a 1-, 2-, 3-, and 4-mer, respectively.
- Each row of a dataset has features consisting of a sequence split into p k-mers, and whether or not the k-mers represent an aptamer sequence. The presence of a k-mer in the sequence is marked with a 1 and the absence of a k-mer is marked with a 0.
- Very few sequences in the dataset (small n)
- Many possible k-mers (high p)
- How do we know which k-mers make up aptamers?

Boruta and the Aptamer Problem

For a dataset consisting of n genetic sequences, p 3-, 4-, and 5-mers, and whether or not the sequences make up an aptamer:

1. Create a shadow feature for each of the p k-mers
2. Run Boruta on the combined dataset.
3. Build a random forest on all k-mers selected by Boruta
4. Calculate out-of-bag (OOB) error on the new random forest model. 30% is the maximum acceptable OOB error.

Example: ATP Binding Sites

See `2-Boruta.R`

Aptamer Problem Results

Out of 23 genetic sequence datasets:

- 2 had OOB error greater than 30% and didn't select any sequences.
- 1 had *increased* OOB error after Boruta from 11% to 38%.
- 20 had average out-of-bag error of 11% and, on average, selected 18 out of 1170 k-mers.
- The k-mers selected were known to be important based on past biological knowledge.

Boruta does pretty darn well!

Outline

Any questions?