Data Mining as Exploratory Data Analysis

Zachary Jones

The Problem(s)

- presumptions
  - social systems are complex
  - causal identification is difficult/impossible with many data sources
  - theory not generally predictively reliable (may be exceptions to this)
- conclusions
  - confidence in assumptions is low
  - analysis is exploratory/descriptive and/or predictive
  - ability to discover unexpected patterns is desirable

Data Mining I

Not a bad thing!

- estimation of $f : X \to Y$ under minimal assumptions
- adapt to data (within a representation class)
- control overadaptation (by minimizing excess risk)

Expected risk (generalization error):

$$R(f) = \mathbb{E}\left[L(Y, f(X))\right]$$

Learning f does not result in directly interpretable output

Data Mining (II)

- statistical theory is hard to come by
- estimation/learning is often heuristic (i.e., not globally optimal)
- Some examples. . .

Decision Trees (I)

Idea: approximate f by recursively splitting $\mathbf{y}$ into bins until $\mathbf{y}$ is sufficiently homogenous in said bins: predict by using a constant function of $\mathbf{y}$ in each bin

- Pros: interpretability, fitting/evaluation speed
- Cons: overadaptation, variance (sharp boundaries), greedy (some work on global optimality though, see evtree)

Decision Trees (II)

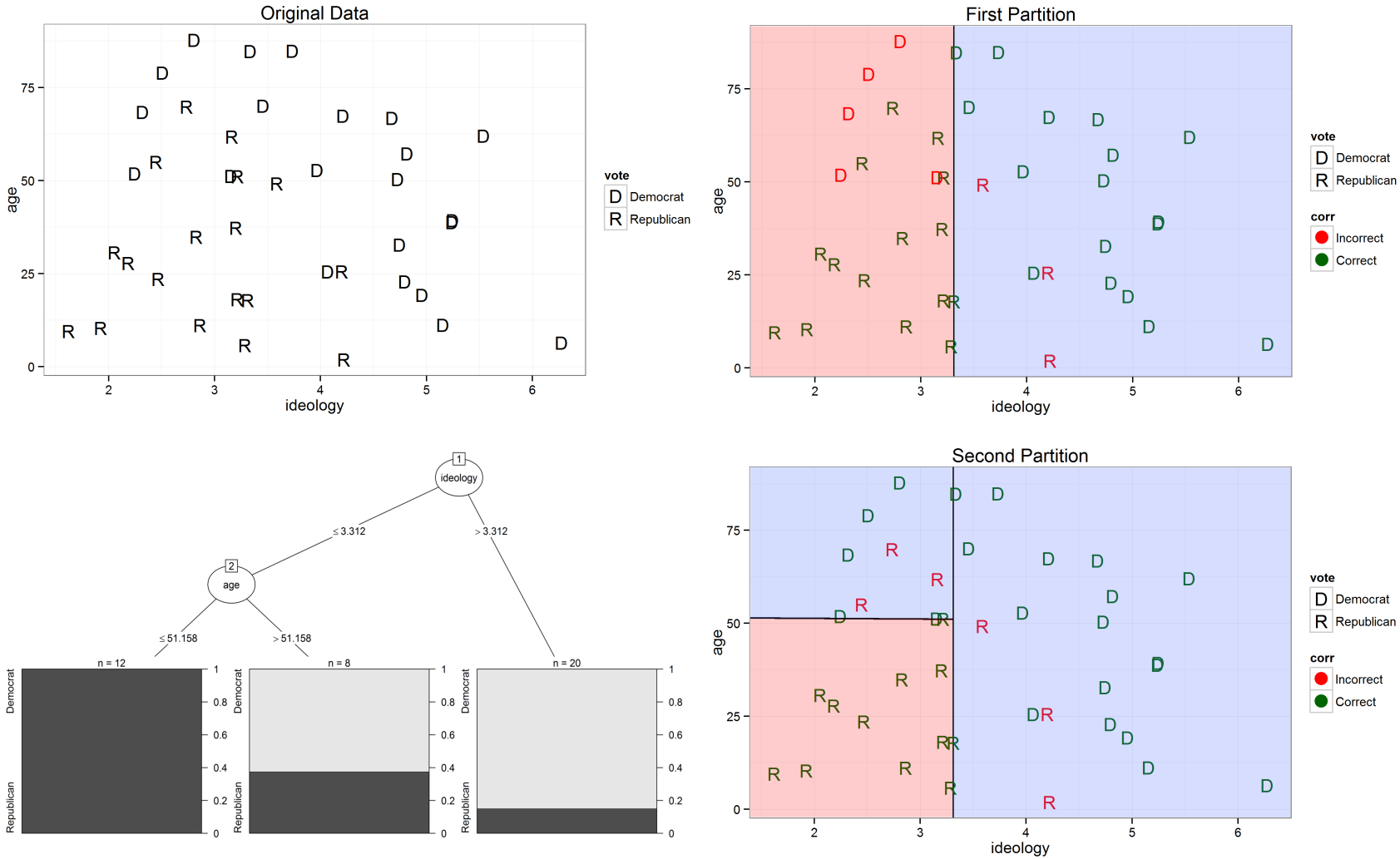

Figure 1:Predicting partisanship from age and ideology (simulated).

Decision Trees (III)

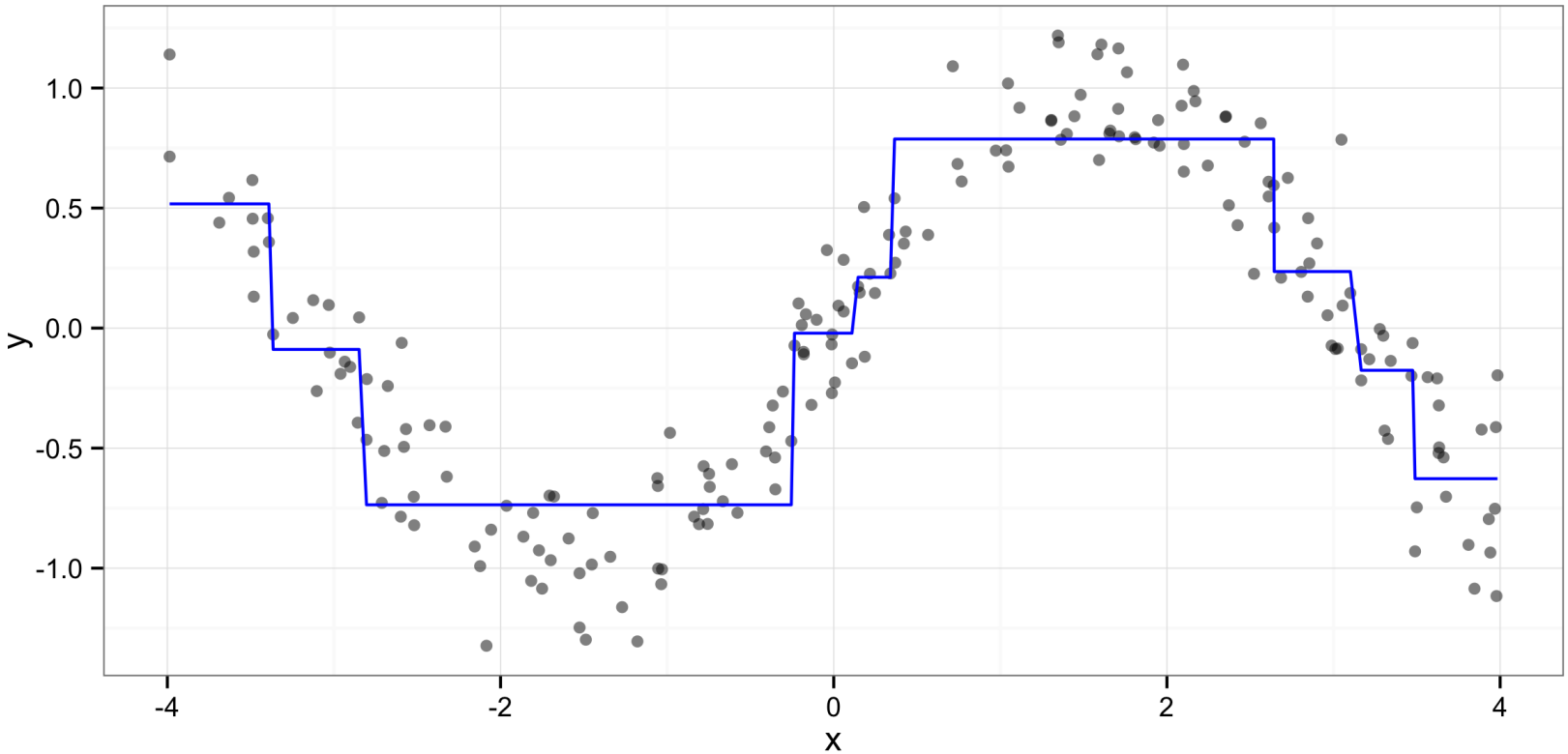

Figure 2: $\hat{f}$ learned from $\sin(x), x \sim U(-4, 4)$ with a decision tree.

Ensembles of Decision Trees

aggregation (bagging)

meta-learning (boosting)

randomization (random forests)

Random Forests (I)

Nice for description/EDA for:

- computational reasons
- usability for many tasks
- some (studied) methods for interpretation
- low number of tuning/hyperparameters
- good empirical performance
- some theory

Random Forests (II)

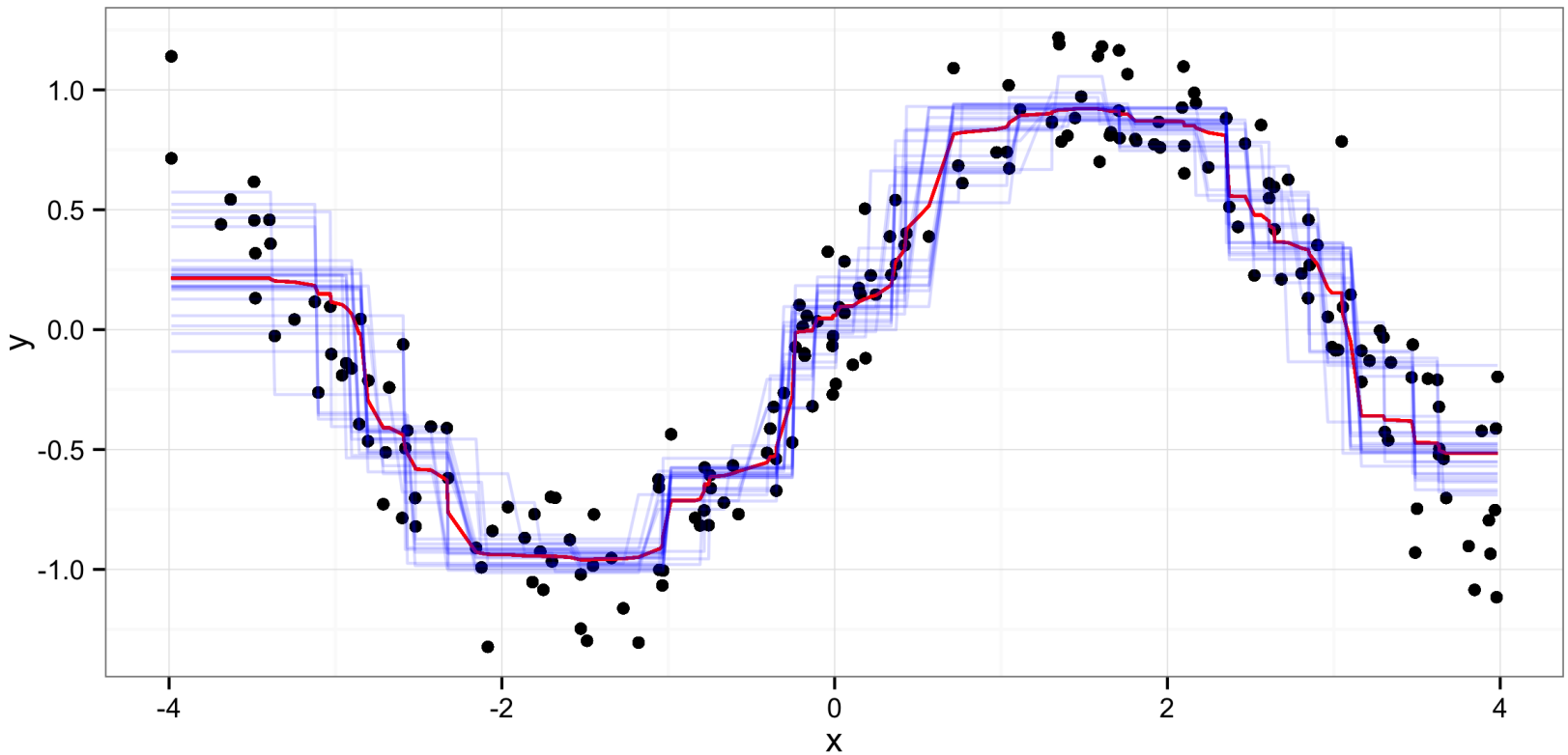

Figure 3: $\hat{f}$ learned from $\sin(x), x \sim U(-4, 4)$ with a randomized, bagged, ensemble of decision trees.

Supervised Learning for Description/EDA

Since most machine learning methods are designed for prediction, their generalization error is low (because they are attempting to make the optimal bias/variance tradeoff

Predicting a complex phenomena reliably gives us some basis on which to interpret $\hat{f}$ (though obviously this is not a causal inference)

But what did $\hat{f}$ learn about f by using X?

Interpreting Black Box Functions (I)

The Marginal Distribution (A)

$$X = X_S \cup X_C$$

S we care about and C we do not

The marginal distribution summarizes how $\hat{f}$ depends on X_s.

$$\hat{f}_S(X_S) = \mathbb{E}_{X_C} \hat{f}(X_S, X_C)$$

The expectation, variance, multiple moments, or the full marginal distribution can then be used.

Interpreting Black Box Functions (I)

The Marginal Distribution (B)

Ideas from Friedman (2001), ESL, and Goldstein et. al. (2015)

We have a function which we can evaluate, so functions of the learned joint distribution are easy!

$$\hat{f}_S(\mathbf{x}_S) = \hat{\mathbb{E}}_{X_C}(\hat{f}(\mathbf{x})) = \frac{1}{N}\sum_{i=1}^{N} \hat{f}(\mathbf{x}_S, \mathbf{x}_C^{(i)})$$

$$\hat{f}_S(\mathbf{x}_S^{(i)}) = \hat{\mathbb{E}}_{X_C}^{(i)}(\mathbf{x}^{(i)}) = \hat{f}(\mathbf{x}_s, \mathbf{x}_c^{(i)})$$

Interpreting Black Box Functions (II)

Derivatives

If $\hat{f}$ is additive in (X_S, X_C) then:

$$\frac{\partial \hat{f}}{\partial X_S} = g(X_S)$$

If not then:

$$\frac{\partial \hat{f}}{\partial X_S} = g(X_S)h(X_C)$$

Numerical differentiation can be applied if $\hat{f}$ continuous and in estimating the derivative of the individual conditional expectation function we can get an idea of whether or not $\hat{f}$ is additive in (X_S, X_C).

Interpretation of Black Box Functions (III)

Feature/Variable Importance

How important is X_S in achieving $R(\hat{f})$?

If the theoretical joint distribution $\mathbb{P}(Y, X_S, X_C) = \mathbb{P}(Y, X_C)\mathbb{P}(X_S)$ then permuting X_S won't increase the prediction error.

$$I_{X_S} = \frac{1}{N} \sum_{i=1}^{N} C(\mathbf{x}_{S\pi}^{(i)}, \mathbf{x}_C^{(i)})$$

$$I_{X_S^{(i)}} = C(\mathbf{x}_{S\pi}^{(i)}, \mathbf{x}_C^{(i)})$$

By using the individual (i) importance rather than the expectation combined with a density estimator, we can estimate the density of the cost function under $\mathbf{x}_{S\pi}$ for different points in the distribution of Y (as estimated from $\mathbf{y}$).

Implementations

- mlr: **M**achine **L**earning with **R** (contributor, first via GSoC)
- edarf: **E**xploratory **D**ata **A**nalysis using **R**andom **F**orests (my package)
- ICEbox: **I**ndividual **C**onditional **E**xpectation plot toolbox (Goldstein et. al. 2015)

On to the demonstration! (`mlr.R` and `edarf.R`)

On my website under "Talks." and at github.com/zmjones/imc

Future Work on Interpretation

All of this will be in MLR!

- extrapolation detection
- more variance estimation
- functional ANOVA decomposition (e.g., best additive decomposition of $\hat{f}$, c.f., Giles Hooker's work)
- local feature importance and density estimation

Future Work on Learning/Estimation

- dependent data! (coming to MLR)
- conditional independence not generally different (i.e., include structure as features)
- estimation of latent variables
- resampling methods
- preprocessing/filtering

Relevant Papers/Writing on Interpretation

- ESL (10.13.2)
- Freidman (2001)
- Roosen (1995)
- Hooker (2004, 2007)
- Goldstein et. al. (2015)