Week 13 Scribing Notes

By: Jack DiPietro, Ben Morgan, Jack Woodburn, Patrick Ryan

A Case Study on Homelessness Services (Paper)

- Analyzing the increasing number of homeless people which is affecting the homeless system
- 5 types of housing
- Emergency Shelter
- Transitional Housing
- Rapid Rehousing
- Homelessness prevention
- Permanent supportive Housing
- Question: Can we improve this system using AI?

What to Keep in Mind

- Fairness: through different biases and discrimination
- Interpretability
- Ethics
- Causal Inference: Need counterfactual estimates of different interventions before you can let ML Model decide

- Question to look at: How much can we potentially improve outcomes?

Kind of Data

- Household Characteristics (Feature)
- Which of the 5 services was assigned to them (Feature)
- Predictor -> Whether they re-enter the Homeless system within 2 years of exit (Label) (Is this a good metric?) (Looking at 2 years)

Approach

- Clean this data to create a dataset
- Train an ML model on this that supports causal inference
    - BART (Bayesian Additive Regression Trees)
    - Provides counterfactual estimates for re-entry under different services
- See if ML model is a good approximation of the real dataset
- Use the output of this ML model in an optimization problem which they can solve to find an "improved assignment"

Results

- Ground Truth (from their dataset) -> 43.04% households re-enter
- Prediction from BART model -> 43.72% households re-enter
- Fairly accurate - proved in paper

Optimize

- After Optimization of Assignments -> 31.88% households re-enter
- Reduction by 27.88%
- After Fairness Constraint -> 37.38% people re-enter

Using ML to Assess the Risk and Prevention of Water Main Breaks (Paper)

- ML system used in Syracuse NY
- Syracuse has a major problem of water main breaks affecting water supply for its residents
- The city has an old infrastructure that has created small and large breaks
- Paper discusses ML model to predict breaks in the city to focus on locating and fixing them before they break

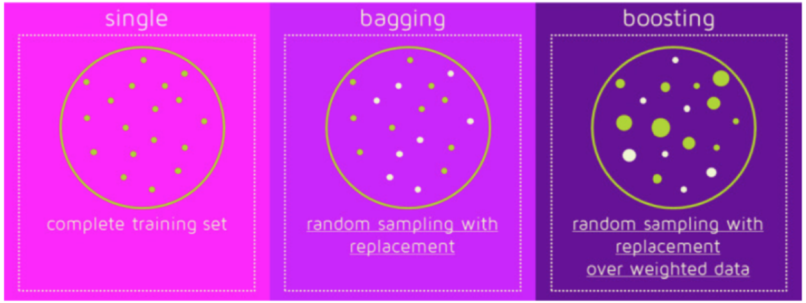
Plan and Result

- Frame it as a binary classification problem of whether a water main break will occur in a given city block within the next 3 years

- Result: Precision of 62% in the top 1% of our predictions

  Two ways in which system can be used:
- 1) for preventative maintenance on the top 1% of the riskiest breaks
- 2) To use the risk scores to coordinate with the Department of Public Works(DPW)

Bagging vs Boosting


single
complete training set
bagging
random sampling with replacement
boosting
random sampling with replacement over weighted data

Bagging

- Stands for Bootstrap Aggregation
- Useful for models with high variance and noisy data
- Takes original data set D with N training examples and creates M copies

Boosting

- Take a week learning algorithm and make it into a very strong one
- Becomes better by focusing on difficult cases
- Basic steps for most Boosting Algorithms
  - Train a weak model on some training data
  - Compute the error of the model on each training example
  - Give higher importance to examples on which the model made mistakes
  - Re-train the model using “importance weight” training examples
  - Go back to step 2

Decision Trees

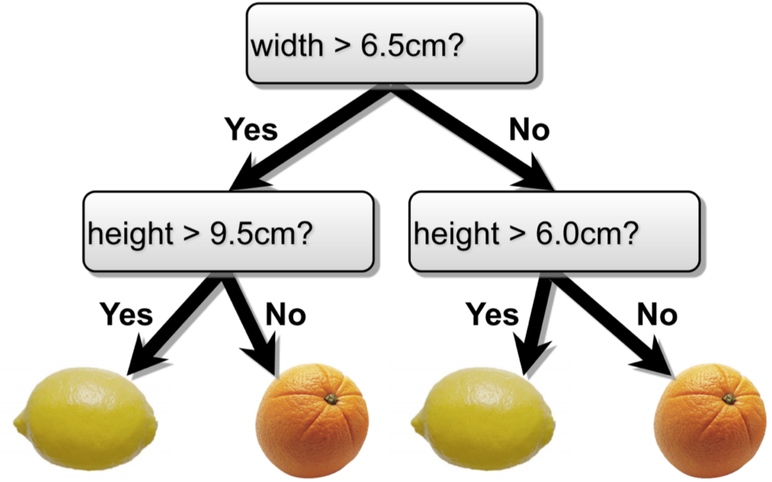
Algorithm

- An attribute for each level that can be labeled true or false
- Output at bottom level
- Boolean functions can be used to generate a truth table

Classification tree has a discrete output

Regression tree has a continuous output

Regularization is necessary to ensure decision trees are compact


width > 6.5cm?

Yes

No

height > 9.5cm?

height > 6.0cm?

Yes

No

Yes

No

Decision Trees

Creating a tree

1. Pick the best attribute and split
2. Repeat at next level

A good decision tree is…

- not too small (is accurate)
- not too big (efficiently reaches the solution)