Machine Learning Models for Causal Inference and Policy Decisionmaking

Interpretability in Machine Learning: Session 2

Elizabeth Petraglia

October 13, 2020

Interpretability in Machine Learning: Session 1

Gonzalo's talk

- Interpretability
- Explanability
- Fairness
- +Transparency

Interpretability in Machine Learning: Session 1

Gonzalo's talk

- Based on Rudin (2019)
- Big-picture ideas: should we use machine learning (ML) for high-stakes decisionmaking? And if so, how do we know that it is doing the "right" thing?
- Answers: maybe? And by using some of the same concepts we would use to check any other model.

What does this mean for us?

Some basic principles about ML (and really, modeling in general . . .)

- The model is not magic.
    - Behind the scenes, there is always an algorithm. The algorithm cannot make nuanced decisions– it will do the job it was programmed to do, and *nothing else*
    - Quality of output is based on quality of input. Garbage in = garbage out.
    - The model cannot create information. It can only do its best to detect useful signals in the information provided.
- The model does not know what you want.
    - It is agnostic to variable names.
    - It does not care about simplicity (unless you penalize it).
    - It does not care about interpretabilty or fairness or decisionmaking.

Other ML considerations

- ML models are usually optimized for *unit-level prediction* (Athey and Imbens).
- Cross-validation is designed to approximate out-of-sample performance, but is still limited to the data you have on hand.
- Nearly all ML or "black box" methods have tuning parameters with defaults set silently. These defaults may or may not be appropriate for your situation.
  - *Just because you can run it in one line of code does not mean the method is simple!*

Other ML considerations

- ML models are usually optimized for *unit-level prediction* (Athey and Imbens).
- Cross-validation is designed to approximate out-of-sample performance, but is still limited to the data you have on hand.
- Nearly all ML or "black box" methods have tuning parameters with defaults set silently. These defaults may or may not be appropriate for your situation.
    - *Just because you can run it in one line of code does not mean the method is simple!*

Other ML considerations

- All models are run with a purpose.
  - If your purpose is to make unit-level predictions for a specific application (e.g., response propensity modeling), off the shelf "black box" models probably are just fine.
  - If you want to make decisions based on model parameters and/or estimates and/or by generalizing model predictions...it’s important to think these issues through.

Practical implications

Machine learning models require just as much, if not more, planning than traditional modeling techniques.

The difference is that with ML, all planning must be done *upfront*.

Practical implications

Machine learning models require just as much, if not more, planning than traditional modeling techniques.

The difference is that with ML, all planning must be done *upfront*.

Typical flow

"Traditional" modeling
RQs $\Longrightarrow$ data $\Longrightarrow$ define model structure $\Longrightarrow$ review output $\Longrightarrow$ tweak model structure (and repeat)

Machine learning
RQs $\Longrightarrow$ data $\Longrightarrow$ select method and set tuning parameters $\Longrightarrow$ review output $\Longrightarrow$ tweak **inputs** (data and/or tuning parameters) (and repeat)

Specific problem: causal inference

Let's focus on the very common problem discussed in Athey and Imbens (2015): we have…

- A lot of potential covariates
- A treatment that we are evaluating (does it work?) **and** we think that its effects might vary by unit characteristics
- One or more outcomes to measure success (or lack therof)

…and we want to come up with the simplest possible model that still accurately captures the relationship between covariates/treatment and the outcome(s).

Predicting conditional causal effects

Some typical techniques:

- Lasso
- Classification/regression trees
- Random forest
- Support vector machines (SVM)...

With cross-validation, all of these techniques are designed to produce the most accurate predicted **outcome** value for each **unit**, with some penalty for model complexity.

However, these methods are **not** guaranteed to identify the factor(s) that are most likely to predict the causal effects of the treatment.

Predicting conditional causal effects

The algorithm doesn't know the difference between the treatment indicator and the other covariates in the model. It just cares about getting the best predictions.

We could end up with a model that makes excellent predictions, but the variables in the model don't give us any information (or worse, give us bad information!) about the causal relationship.

These could be from interaction terms, or simply because the model identifies a variable that is correlated with the true causal effect (e.g., less than high school education and being under 18).

These are issues in non-ML models too– ML is not a magical solution to fix them!

Predicting conditional causal effects: general recommendations

Most of these are in the spirit of Rudin (2019), improving interpretability by constraining the model in advance.

- Exploratory data analysis is **immensely** important. You need to know which variables have pathological distributions, which ones are mostly missing or perfectly correlated– many ML algorithms can handle these cases silently, which is not always a good thing!
- For non-exploratory work, a "kitchen sink" approach may not be best. Be selective about which variables to enter into models. Which ones have some evidence of causality?

Predicting conditional causal effects: general recommendations

- Determine what cell sizes are meaningful. Do you really want to have an interaction term in your model that is only useful for 10 cases? 50 cases? 100 cases? Recode or drop variables with categories that fall below that threshhold.
- Constrain your models from the beginning. Although it's tempting to say you want to learn everything about your data, do you really need to fit a tree that allows 10-way interaction terms?
- "Laugh test" your findings. Just because the ML algorithm spits something out doesn't mean it's "right."

Predicting conditional causal effects: Athey and Imbens

Athey and Imbens propose alternate ways of evaluating a cross-validated ML model, basing goodness-of-fit measures on the estimated treatment effect rather than the outcome directly:

- Transforming the outcome measure (using the conditional estimated ATE as the outcome).
- Matching T and C cases, and then using the matched cases to estimate the conditional average treatment effect.
- ... as well as proposing alternate ways of estimating the treatment effect within each leaf.

These are both methods that essentially change the goodness-of-fit criterion used to select the "best" model. Instead of picking the model that's best at predicting the outcome, we're picking the model that's best at predicting the conditional estimated treatment effect– which is what we're actually interested in!

Predicting conditional causal effects: Athey and Imbens

Athey and Imbens propose alternate ways of evaluating a cross-validated ML model, basing goodness-of-fit measures on the estimated treatment effect rather than the outcome directly:

- Transforming the outcome measure (using the conditional estimated ATE as the outcome).
- Matching T and C cases, and then using the matched cases to estimate the conditional average treatment effect.
- ... as well as proposing alternate ways of estimating the treatment effect within each leaf.

These are both methods that essentially change the goodness-of-fit criterion used to select the "best" model. Instead of picking the model that's best at predicting the outcome, we're picking the model that's best at predicting the conditional estimated treatment effect– which is what we're actually interested in!

Summary

- Machine learning algorithms can be useful tools for a wide range of problems, but remember that they are only tools and not oracles.
- You as a statistician or analyst need to work with your subject matter experts to build a model that makes sense. Just because you successfully run a model does not mean that it is producing the results you expect.
- Before you run an ML model, be sure that the model is appropriate for the research goals. A fancier algorithm does not mean better results!

Next (and final!) session

Session 1: Framing the big picture issues

Session 2: Practical tips **before and while** you select or run an ML algorithm.

Session 3: Practical methods for **explaining output** from an ML algorithm

Thursday, 11/5 at 12 noon

Thank you!

elizabethpetraglia@westat.com