Machine Learning and Causal Inference

Mixtape Track

Prediction vs. Causality

Imagine you are a life insurance underwriter. You receive an application for life insurance from someone with the following characteristics:

- male

Prediction vs. Causality

Imagine you are a life insurance underwriter. You receive an application for life insurance from someone with the following characteristics:

- male
- age 67

Prediction vs. Causality

Imagine you are a life insurance underwriter. You receive an application for life insurance from someone with the following characteristics:

- male
- age 67
- high blood pressure

Prediction vs. Causality

Imagine you are a life insurance underwriter. You receive an application for life insurance from someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol

Prediction vs. Causality

Imagine you are a life insurance underwriter. You receive an application for life insurance from someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol
- family history of heart disease

Prediction vs. Causality

Imagine you are a life insurance underwriter. You receive an application for life insurance from someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol
- family history of heart disease
- and . . .

Prediction vs. Causality

Imagine you are a life insurance underwriter. You receive an application for life insurance from someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol
- family history of heart disease
- and . . .
- was admitted to the hospital yesterday

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

- male

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

- male
- age 67

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

- male
- age 67
- high blood pressure

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol
- family history of heart disease

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol
- family history of heart disease
- and . . .

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

- male
- age 67
- high blood pressure
- high cholesterol
- family history of heart disease
- and . . .
- is having chest pains.

Prediction vs. Causality

Now imagine you are a loved one of someone with the following characteristics:

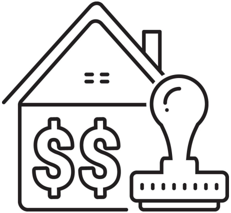
- male
- age 67
- high blood pressure
- high cholesterol
- family history of heart disease
- and . . .
- is having chest pains.
- Should you take him to the hospital?

Prediction vs. Causality: Purpose

Prediction vs. Causality: Purpose

Prepare

- A loan officer wants to know the likelihood of an individual repaying a loan based on income, employment, and other characteristics.



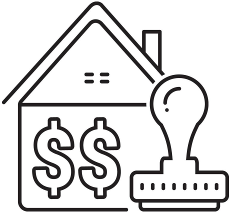
Prediction vs. Causality: Purpose

Prepare

- A loan officer wants to know the likelihood of an individual repaying a loan based on income, employment, and other characteristics.

Influence

- A mortgage lender wants to know if direct debit will increase loan repayments



Prediction vs. Causality: Purpose

Prepare

Influence

- In order to decide whether to invest in a start-up, an investor needs to know how likely the start-up is to succeed, given the entrepreneur's experience and the characteristics of the industry.

Prediction vs. Causality: Purpose

Prepare

- In order to decide whether to invest in a start-up, an investor needs to know how likely the start-up is to succeed, given the entrepreneur's experience and the characteristics of the industry.

Influence

- An entrepreneur needs to know what the effect of receiving funding from a private equity investor (rather than getting a loan) is on the ultimate success of an enterprise.

Prediction vs. Causality: Purpose

Prepare

Influence

- A bail hearing judge needs to know how likely a defendant is to flee before trial, given his or her charges, criminal history, and other characteristics

Prediction vs. Causality: Purpose

Prepare

- A bail hearing judge needs to know how likely a defendant is to flee before trial, given his or her charges, criminal history, and other characteristics

Influence

- A policy maker needs to know the effect of being released on bail (rather than detained) prior to trial on ultimate conviction

Prediction vs. Causality: Purpose

Prepare

Influence

- A home seller wants to know what price homes with the characteristics of his or her home typically sell for

Prediction vs. Causality: Purpose

Prepare

- A home seller wants to know what price homes with the characteristics of his or her home typically sell for

Influence

- A home seller wants to know by how much installing new windows will raise the value of his or her home

Prediction vs. Causality: Purpose

Prepare

- A Harvard admissions officer wants to know how likely an applicant with given credentials is to graduate in 4 years

Influence

Prediction vs. Causality: Purpose

Prepare

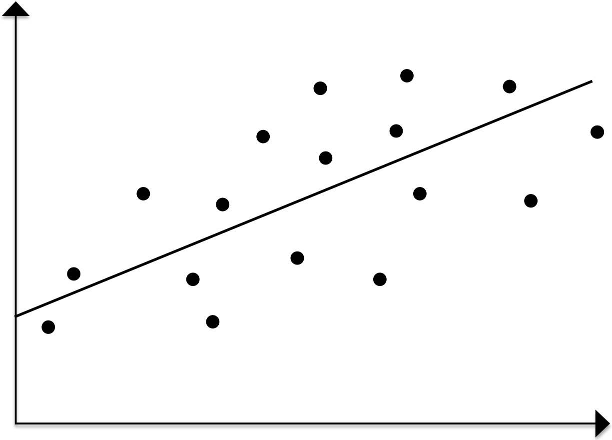
- A Harvard admissions officer wants to know how likely an applicant with given credentials is to graduate in 4 years

Influence

- A labor economist wants to know whether individuals of a certain ethnic background are less likely to get into Harvard than applicants with similar academic credentials

Prediction vs. Causality: Target

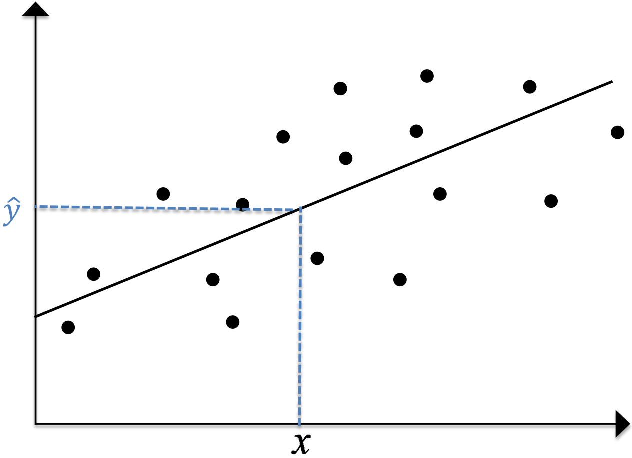
$$y_i = \alpha + \beta x_i + \varepsilon_i$$



Prediction vs. Causality: Target

Prediction

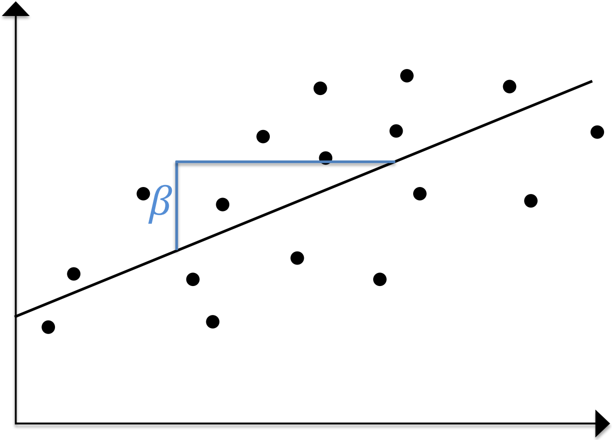
$$y_i = \underbrace{\alpha + \beta x_i}_{\hat{y}} + \varepsilon_i$$



Prediction vs. Causality: Target

Causality

$$y_i = \alpha + \beta x_i + \varepsilon_i$$



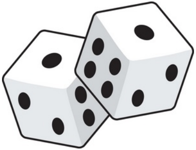
Prediction vs. Causality: Methods

Causality

Prediction vs. Causality: Methods

Causality

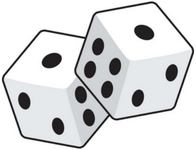
- Gold standard: RCT



Prediction vs. Causality: Methods

Causality

- Gold standard: RCT

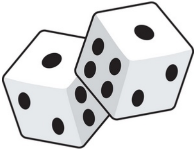


- Aluminum standard: Regression or IV strategies that approximate controlled experiments

Prediction vs. Causality: Methods

Causality

- Gold standard: RCT



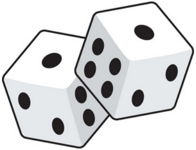
- Aluminum standard: Regression or IV strategies that approximate controlled experiments

Prediction

Prediction vs. Causality: Methods

Causality

- Gold standard: RCT



- Aluminum standard: Regression or IV strategies that approximate controlled experiments

Prediction

- Supervised machine learning algorithms

Prediction vs. Causality: Where shall the twain meet?

We've seen that prediction and causality

- answer different questions

Prediction vs. Causality: Where shall the twain meet?

We've seen that prediction and causality

- answer different questions
- serve different purposes

Prediction vs. Causality: Where shall the twain meet?

We've seen that prediction and causality

- answer different questions
- serve different purposes
- seek different targets

Prediction vs. Causality: Where shall the twain meet?

We've seen that prediction and causality

- answer different questions
- serve different purposes
- seek different targets
- use different methods

Prediction vs. Causality: Where shall the twain meet?

We've seen that prediction and causality

- answer different questions
- serve different purposes
- seek different targets
- use different methods

Different strokes for different folks, or complementary tools in an applied economist's toolkit?

Prediction vs. Causality: Where shall the twain meet?

We've seen that prediction and causality

- answer different questions
- serve different purposes
- seek different targets
- use different methods

Different strokes for different folks, or **complementary tools in an applied economist's toolkit**?

Prediction vs. Causality: Where shall the twain meet?

We've seen that prediction and causality

- answer different questions
- serve different purposes
- seek different targets
- use different methods

Different strokes for different folks, or **complementary tools in an applied economist's toolkit**?

- Illustrate using the Oregon Health Insurance Experiment (go to python)

Where ML fits into causal inference

Traditional regression strategy:

1. Regress Y_i on X_i and compute the residuals,

$$
\begin{aligned}
\tilde{Y}_i &= Y_i - \hat{Y}_i^{OLS}, \\
\hat{Y}_i^{OLS} &= X_i' \left(X'X \right)^{-1} X'Y
\end{aligned}
$$

2. Regress D_i on X_i and compute the residuals,

$$
\begin{aligned}
\tilde{D}_i &= D_i - \hat{D}_i^{OLS}, \\
\hat{D}_i^{OLS} &= X_i' \left(X'X \right)^{-1} X'D
\end{aligned}
$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

Where ML fits into causal inference

Traditional regression strategy:

1. Regress Y_i on X_i and compute the residuals,

$$
\begin{aligned}
\tilde{Y}_i &= Y_i - \hat{Y}_i^{OLS}, \\
\hat{Y}_i^{OLS} &= X_i' \left(X'X\right)^{-1} X'Y
\end{aligned}
$$

2. Regress D_i on X_i and compute the residuals,

$$
\begin{aligned}
\tilde{D}_i &= D_i - \hat{D}_i^{OLS}, \\
\hat{D}_i^{OLS} &= X_i' \left(X'X\right)^{-1} X'D
\end{aligned}
$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

When OLS might not be the right tool for the job:

- there are many variables in X_i
- the relationship between X_i and Y_i or D_i may not be linear

Where ML fits into causal inference

ML-augmented regression strategy:

1. Predict Y_i using X_i with ML and compute the residuals,

$$\begin{aligned} \tilde{Y}_i &= Y_i - \hat{Y}_i^{ML}, \\ \hat{Y}_i^{ML} &= \text{prediction generated by ML} \end{aligned}$$

2. Predict D_i using X_i with ML and compute the residuals,

$$\begin{aligned} \tilde{D}_i &= D_i - \hat{D}_i^{ML}, \\ \hat{D}_i^{ML} &= \text{prediction generated by ML} \end{aligned}$$

3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

Where ML fits into causal inference

ML-augmented regression strategy:

1. Predict Y_i using X_i with ML and compute the residuals,

$$\begin{aligned} \tilde{Y}_i &= Y_i - \hat{Y}_i^{ML}, \\ \hat{Y}_i^{ML} &= \text{prediction generated by ML} \end{aligned}$$

2. Predict D_i using X_i with ML and compute the residuals,

$$\begin{aligned} \tilde{D}_i &= D_i - \hat{D}_i^{ML}, \\ \hat{D}_i^{ML} &= \text{prediction generated by ML} \end{aligned}$$

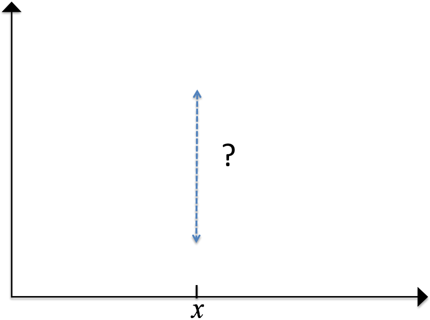
3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.

Most common ML methods in applied economics:

- Lasso
- Ridge
- Elastic net
- Random forest

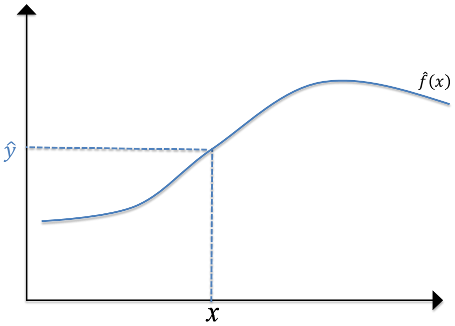
Getting serious about prediction

- **Goal:** Predict an out-of-sample outcome Y



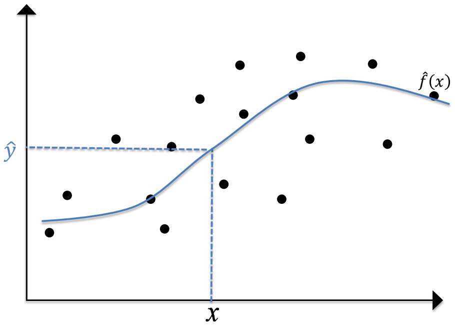
Getting serious about prediction

- **Goal:** Predict an out-of-sample outcome Y
- as a function, $\hat{f}(X)$, of **features** $X = (1, X_1, X_2, \ldots, X_K)'$.



Getting serious about prediction

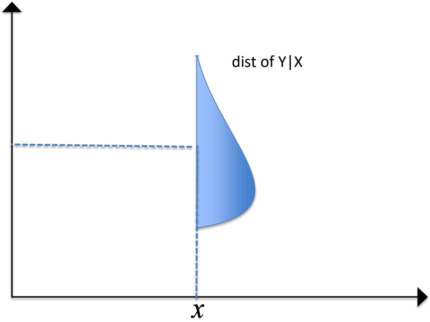
- **Goal:** Predict an out-of-sample outcome Y
- as a function, $\hat{f}(X)$, of **features** $X = (1, X_1, X_2, \ldots, X_K)'$.
- Estimate the function $\hat{f}$ (aka "train the model") based on **training sample** $\{(Y_i, X_i); i = 1, ..., N\}$



Cutting our losses

- Want our prediction to be "close," i.e. minimize the expected **loss function**:

$$\min_{f(x)} E\left[L\left(Y - f\left(x\right)\right) | X = x\right]$$

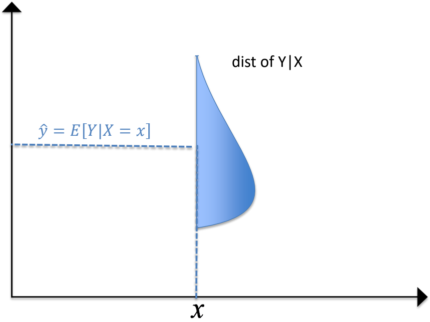


Cutting our losses

- Want our prediction to be "close," i.e. minimize the expected **loss function**:

$$\min_{f(x)} E\left[L\left(Y - f\left(x\right)\right) | X = x\right]$$

- **Squared loss**: $L(d) = d^2 \implies f^*(x) = E\left[Y | X = x\right]$

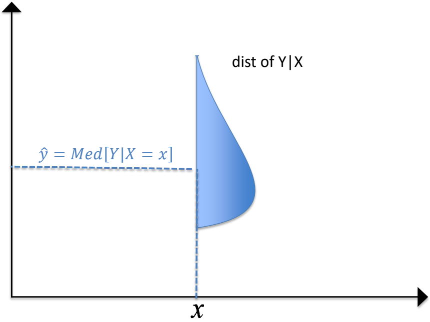


Cutting our losses

- Want our prediction to be "close," i.e. minimize the expected **loss function**:

$$\min_{f(x)} E\left[L\left(Y - f\left(x\right)\right) \middle| X = x\right]$$

- **Squared loss**: $L(d) = d^2 \implies f^*(x) = E\left[Y|X = x\right]$
- **Absolute loss**: $L(d) = |d| \implies f^*(x) = Med\left[Y|X = x\right]$

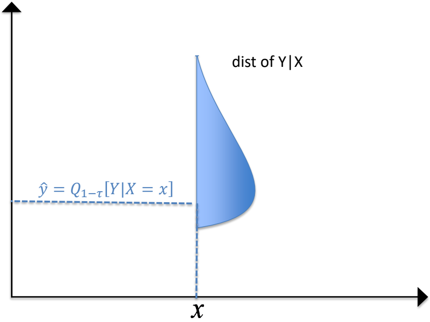


Cutting our losses

- Want our prediction to be "close," i.e. minimize the expected **loss function**:

$$\min_{f(x)} E\left[L\left(Y - f\left(x\right)\right) | X = x\right]$$

- **Squared loss**: $L(d) = d^2 \implies f^*(x) = E\left[Y|X = x\right]$
- **Absolute loss**: $L(d) = |d| \implies f^*(x) = Med\left[Y|X = x\right]$
- **Asymmetric loss**:
  $L_\tau\left(d\right) = d\left(\tau - 1\left(d < 0\right)\right) \implies f^*(x) = Q_{1-\tau}\left[Y|X = x\right]$



Navigating the Bias-Variance Tradeoff

- Prediction problem solved if we knew $f^*(x) = E\left[Y|X = x\right]$

Navigating the Bias-Variance Tradeoff

- Prediction problem solved if we knew $f^*(x) = E\left[Y|X = x\right]$
- But we have to settle for an estimate: $\hat{f}(x)$;

$E\left[\left(Y - \hat{f}(x)\right)^2 \middle| X = x\right]$ becomes:

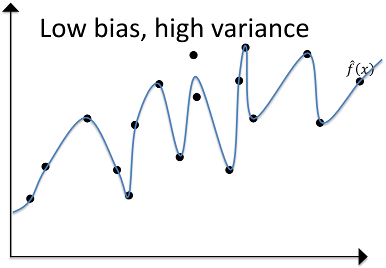
$$
\begin{aligned}
&\left(E\left[\hat{f}(x) - f^*(x)\right]\right)^2 && \text{prediction bias squared} \\
&+ E\left[\left(\hat{f}(x) - E\left[\hat{f}(x)\right]\right)^2\right] && \text{prediction variance} \\
&+ E[(Y - f^*(x))^2|X = x] && \text{irreducible error.}
\end{aligned}
$$

Navigating the Bias-Variance Tradeoff

- Prediction problem solved if we knew $f^*(x) = E\left[Y|X = x\right]$
- But we have to settle for an estimate: $\hat{f}(x)$;

$E\left[\left(Y - \hat{f}(x)\right)^2 \middle| X = x\right]$ becomes:

$$\left(E\left[\hat{f}(x) - f^*(x)\right]\right)^2 \quad \text{prediction bias squared}$$
$$+\, E\left[\left(\hat{f}(x) - E\left[\hat{f}(x)\right]\right)^2\right] \quad \text{prediction variance}$$
$$+\, E[(Y - f^*(x))^2|X = x] \quad \text{irreducible error.}$$

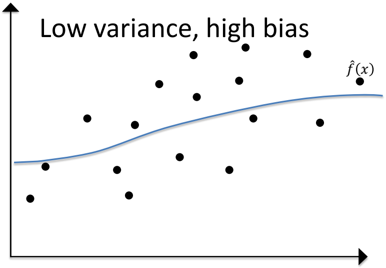


Navigating the Bias-Variance Tradeoff

- Prediction problem solved if we knew $f^*(x) = E\left[Y|X = x\right]$
- But we have to settle for an estimate: $\hat{f}(x)$;

$E\left[\left(Y - \hat{f}(x)\right)^2\middle|X = x\right]$ becomes:

$$\begin{aligned} &\left(E\left[\hat{f}(x) - f^*(x)\right]\right)^2 && \text{prediction bias squared} \\ +\ &E\left[\left(\hat{f}(x) - E\left[\hat{f}(x)\right]\right)^2\right] && \text{prediction variance} \\ +\ &E[(Y - f^*(x))^2|X = x] && \text{irreducible error.} \end{aligned}$$



Python example: predicting earnings in the NLSY

Penalized Regression: Lasso

- When is it the right tool for the job:
  - When you have a large number of potential regressors (including powers or other transformations), maybe even more than the sample size!
  - Out of these, only a relatively few (but you don't know which) really matter (what do we mean by "matter?"). We call this **approximate sparsity**
- Theoretical definition:

$$\arg\min_b \sum_{i=1}^{n} \left(y_i - x_i' b\right)^2 + \lambda \sum_{j=1}^{k} |b_j|$$

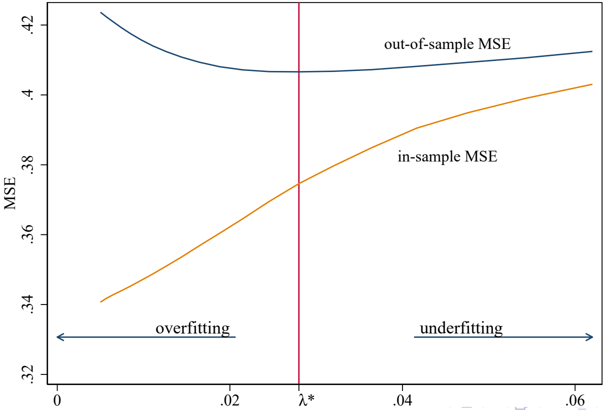
  What does λ do and how do we choose it?
- Caveats and considerations:
  - Important to standardize regressors pre-lasso
  - Can give unexpected results with dummy variables
  - Resist the temptation to interpret coefficients or the included variables as the "true model!"
- Let's give it a go in python!

Choosing Tuning Parameters: Cross-Validation

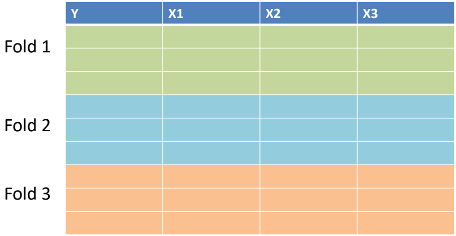
All supervised ML methods have tuning parameters:

- Lasso: λ
- Ridge: α
- Random forests: tree depth, etc.

Tuning parameters are the rudder by which we navigate the bias-variance tradeoff.



Choosing Tuning Parameters: Cross-Validation



Cross-validation procedure: Divide sample in K folds

- Choose some value of the tuning parameter, λ
- For each fold $k = 1, \ldots, K$
  1. Train model leaving out fold k
  2. Generate predictions in fold k
  3. Compute MSE for fold k: $MSE_k = \frac{1}{n_k} \sum_{i \in k} \left(Y_i - \hat{Y}_i \right)^2$
- Compute overall MSE correponding to the current choice of λ: $MSE(\lambda) = \frac{1}{K} \sum_{k=1}^{K} MSE_k$

Repeat the above for many values of λ, and choose the value λ^* with the lowest cross-validated MSE—time for python!

Penalized Regression: Ridge

- When is it the right tool for the job:
  - When you have a large number of regressors including highly collinear ones
- Theoretical definition:

$$\begin{aligned} & \arg\min_b \sum_{i=1}^{n} \left(y_i - x_i' b\right)^2 + \alpha \sum_{j=1}^{k} b_j^2 \\ = \quad & \left(X'X + \alpha I\right)^{-1} X'Y \end{aligned}$$

- Caveats and considerations:
  - Important to standardize regressors pre-ridge
  - Shrinks (biases) coefficients towards zero, but not all the way (unlike lasso)
- Let's give it a go in python!

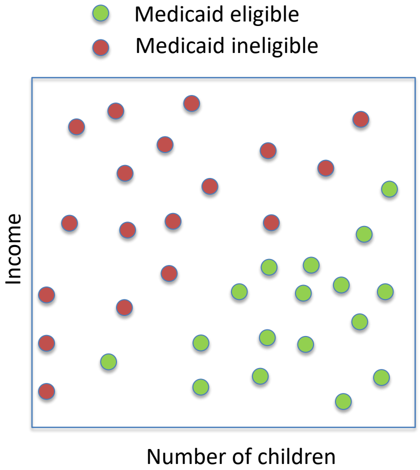
Penalized Regression: Elastic Net

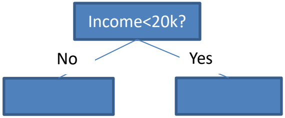
- Combines lasso and ridge approaches
- Theoretical definition:

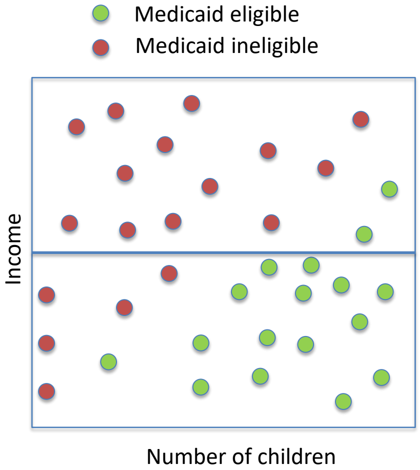
$$\arg\min_b \sum_{i=1}^{n} \left(y_i - x_i' b\right)^2 + \alpha\gamma \sum_{j=1}^{k} |b_j| + .5\alpha\,(1-\gamma) \sum_{j=1}^{k} b_j^2$$

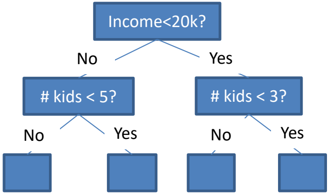
- Caveats and considerations:
  - Two tuning parameters: α and γ
  - Important to standardize regressors pre-ridge
  - Zeros out many regressors, shrinks (biases) remaining coefficients towards zero
- Let's give it a go in python!

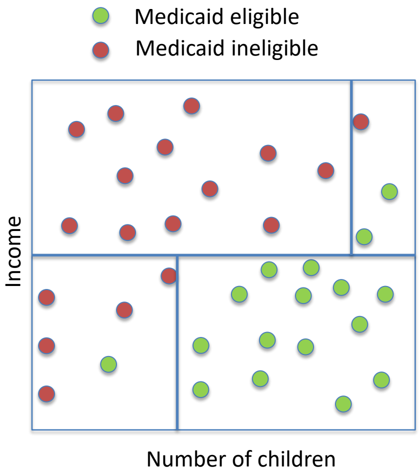
Initial node

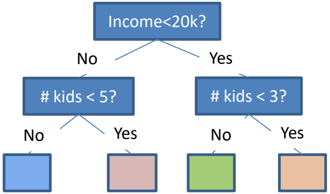


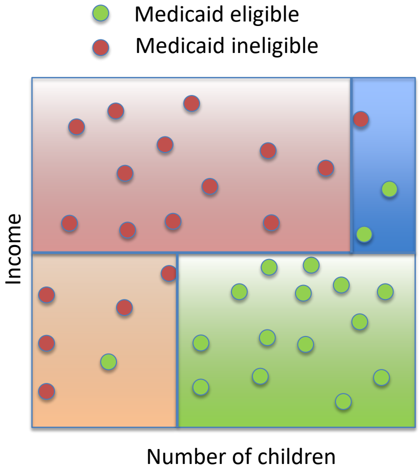

Income<20k?
No
Yes


Medicaid eligible
Medicaid ineligible
Income
Number of children

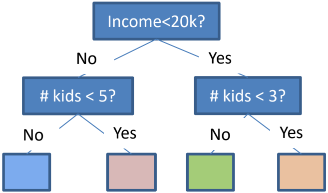

Income<20k?
No
Yes
kids < 5?
kids < 3?
No
Yes
No
Yes


Medicaid eligible
Medicaid ineligible
Income
Number of children


Income<20k?
No
Yes
kids < 5?
kids < 3?
No
Yes
No
Yes

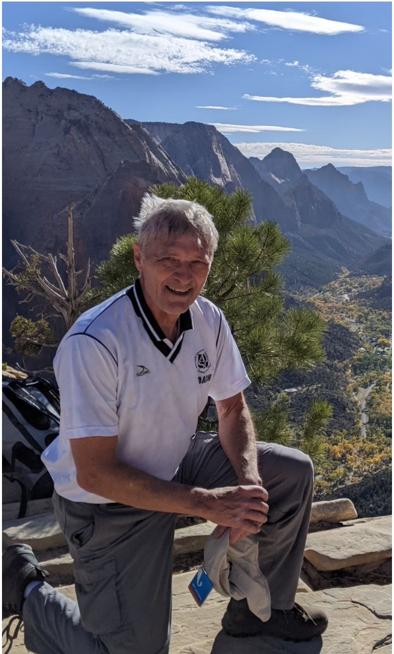

Medicaid eligible
Medicaid ineligible
Income
Number of children

Trees and Forests

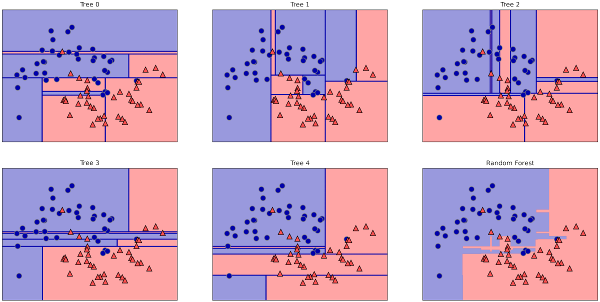


- Where to split:
  Choose the feature from $\{x_1, \ldots, x_p\}$ and the value of that feature to minimize MSE in the resulting child nodes
- Tuning parameters
  - Max depth
  - Min training obs per leaf
  - Min improvement in fit

Wisdom of the crowd: predict my father's age!



Forest for the Trees



- Value proposition: reduce variance by averaging together multiple predictions
- The catch: individual trees need to be **de-correlated**
- Algorithm:
  - Grow B trees, each on a different bootstrapped sample
  - At each split, consider only a random subset of features
  - Average together the individual predictions
- Let's grow some trees in python!

Back to our original framework for where machine learning There are two flavors of doing this: post double-selection lasso (PDS lasso), and double machine learning (DML).

First PDS lasso. The idea is to replace $\hat{Y}_i^{OLS}$ with $\hat{Y}_i^{PDS} = X_i' \left(\tilde{X}'\tilde{X}\right)^{-1} \tilde{X}'Y$ and $\hat{D}_i^{OLS}$ with $\hat{D}_i^{PDS} = X_i' \left(\tilde{X}'\tilde{X}\right)^{-1} \tilde{X}'D$, where $\tilde{X}$ contains the union of variables retained by a lasso of Y_i on X_i and a lasso of D_i on X_i.

But then this is the same as the following procedure:

1. Lasso Y_i on X_i, call the retained regressors $\tilde{X}_i^Y$
2. Lasso D_i on X_i, call the retained regressors $\tilde{X}_i^D$
3. Estimate via OLS $Y_i = \delta D_i + \tilde{X}_i\tilde{\beta} + \tilde{\varepsilon}_i$, where $\tilde{X}_i = \tilde{X}_i^Y \cup \tilde{X}_i^D$

Couple of notes. X_i should contain transformations and interactions of any underlying regressors to approximate nonlinear functional forms. Lasso relies on approximate sparsity, which may not always be appropriate. An important choice with lasso is the penalty parameter, λ. Chernozhukov, et al. have formulas you can use, or just cross validate. Inference? Just use the normal standard errors from the last step.

Now double machine learning. PDS lasso really only works for lasso, because only lasso "selects" variables. But lasso isn't always the right tool for the job (maybe sparsity isn't plausible). DML can work with any supervised machine learning method. Here's the idea. Replace $\hat{Y}_i^{OLS}$ with $\hat{Y}_i^{DML}$, a prediction generated by a machine learning model trained on a set of observations that does not include i. We accomplish this via *cross-fitting*: divide the training set into K folds (usually we choose $K = 5$ or 10). Train the machine learning model 5 times, leaving out a different fold each time. Each $\hat{Y}_i^{DML}$ is generated from the model trained when its fold was left out. Inference? Just use the normal standard errors from the last step. Here is the procedure:

1. Divide the sample into K folds
2. For $k = 1, \ldots, K$
   - 2.1 Train a model to predict Y given X, leaving out observations i in fold k: $\hat{Y}^{-k}(x)$
   - 2.2 Train a model to predict D given X, leaving out observations i in fold k: $\hat{D}^{-k}(x)$
   - 2.3 Form residuals $\tilde{Y}_i = Y_i - \hat{Y}^{-k}(X_i)$ and $\tilde{D}_i = D_i - \hat{D}^{-k}(X_i)$
3. Regress $\tilde{Y}_i$ on $\tilde{D}_i$.