Session 4:

Matching and causal trees

Andreas Bjerre-Nielsen

Comments on Rasmus presentation

- Lloyd Shapley - matching mechanims vs. econometrics matching
- Interpretability - the why..
- Feature importance - tests vs. tools
    - Other tools include Individual Conditional Expectation, Surrogate models.

Agenda

1. Causality
2. Potential outcomes
3. Experiments

4. Matching

    - Covariate based matching
    - Propensity score matching

5. Heterogeneous treatment effects with causal trees

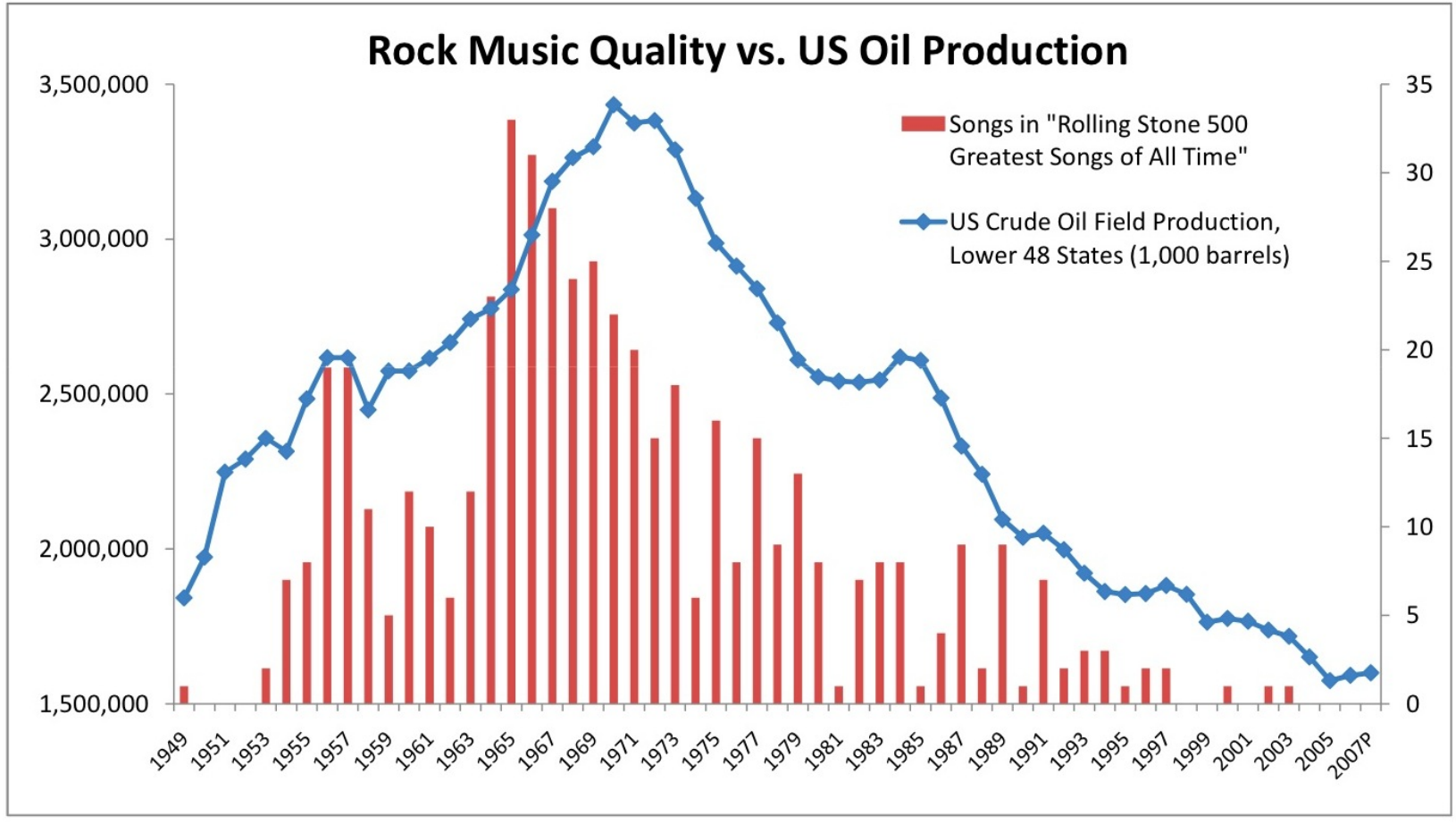
Buckle up...

In [13]:
```
import matplotlib.pyplot as plt
import networkx as nx
import numpy as np
import pandas as pd
import seaborn as sns
%matplotlib inline
```

Causality

Correlation does not imply causation

Spurious or causal?



What is causality?

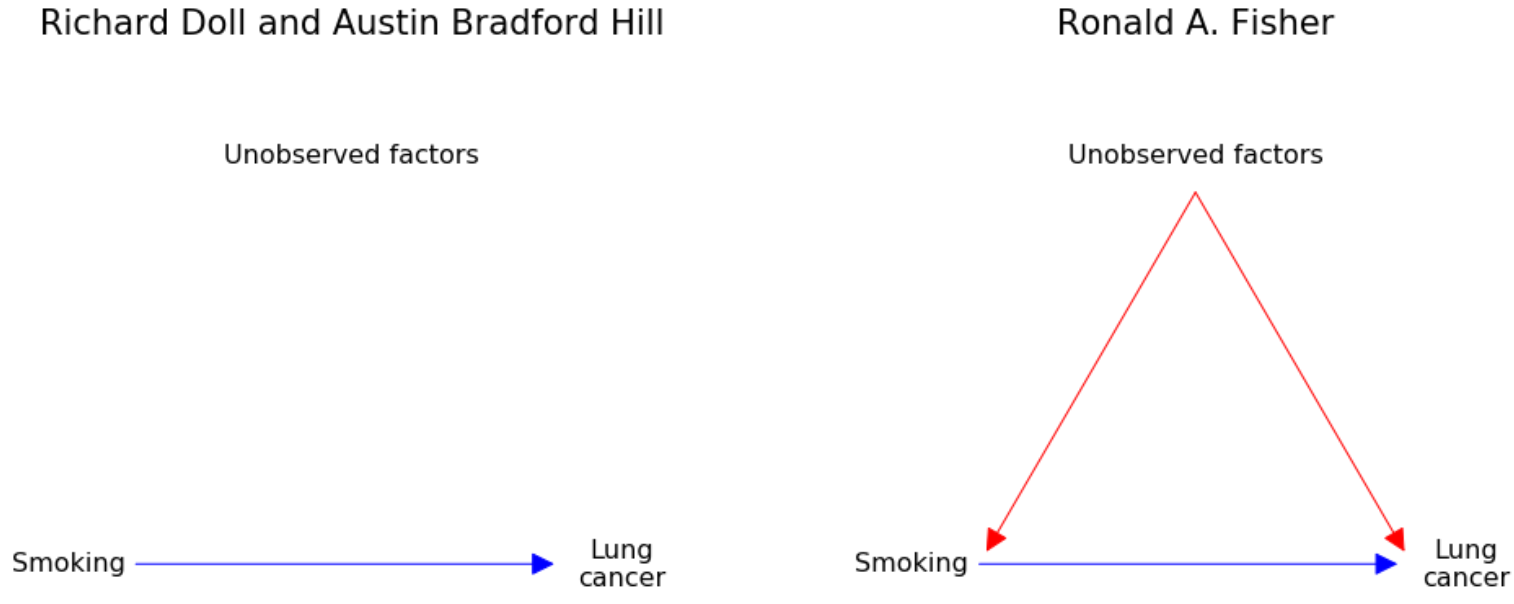
Relationship between two or more variables such that whereby a change in one or more variable(s) ***affect(s)*** the distribution of one or more other variable(s).

We can draw these relationships (from The Book of Why, Judea Pearl), e.g. smoking example.

- Ronald Fisher argued that unobserved confounders could cause smoking and lung cancer

In [10]:
```
f_lung_cancer
```

Out[10]:



Establishing causality

Currently there are two broad approaches for establishing causal relationships:

- Experiment and quasi-experiments
    - Corresponds to what is taught in *Mostly Harmless Econometrics*
- Structural equation models
    - Used for structural econometric choice models etc.
    - Also used estimating causal graphs, e.g. as by Judea Pearl

Potential outcomes

The aim

We are interested in the effect of some treatment, e.g.

- getting admitted to a certain educaton on wages, life-expectancy
- access to paternity leave on wages (husband and wife)

The Rubin Causal Model

Denote the treatment variable as D_i where $D_i = 1$ corresponds to unit i being treated, while $D_i = 0$ is not treated. Define the potential outcomes:

$$Y_i = \begin{cases} Y_i(1), & D_i = 1; \\ Y_i(0), & D_i = 0. \end{cases}$$

The observed outcome Y_i can be written in terms of potential outcomes as

$$Y_i = Y_i(0) + [Y_i(1) - Y_i(0)] \cdot D_i$$

$Y_i(1) - Y_i(0)$ is the *causal* effect of D_i on Y_i.

But we never observe the same individual i in both states. This is the **fundamental problem of causal inference**.

Selection Bias

We need some way of estimating the state we do not observe (the ***counterfactual***)

Usually, our sample contains individuals from both states - treated and untreated.

So why not do a naive comparison of averages by treatment status? i.e.
$E[Y_i|D_i = 1] - E[Y_i|D_i = 0]$

Selection Bias II

We can rewrite into:

$$\begin{aligned} E[Y_i|D_i=1] - E[Y_i|D_i=0] =& E[Y_i(1)|D_i=1] - E[Y_i(0)|D_i=1] + \\ & E[Y_i(0)|D_i=1] - E[Y_i(0)|D_i=0] \end{aligned}$$

The decomposition:

- $E[Y_i(1)|D_i=1]$: the average *causal* effect of D_i on Y.
  $- E[Y_i(0)|D_i=1]$
  $= E[Y_i(1) - Y_i(0)|D_i=1]$
- $E[Y_i(0)|D_i=1] - E[Y_i(0)|D_i=0]$: difference in average $Y_i(0)$ between the two groups. Likely to be different from 0 when individuals are allowed to self-select into treatment. Often referred to as ***selection bias***.

Experiments

Random assignment solves the problem

Random assignment implies D_i is independent of potential outcomes

- Selection bias term is zero: $E[Y_i(0)|D_i = 1] = E[Y_i(0)|D_i = 0]$
- Intuition: non-treated individuals can be used as counterfactuals for treated (*what would have happened to individual i had he not received the treatment*?)
- Overcome the fundamental problem of causal inference

Randomization

Holland and Rubin (1986)

> *no causation without manipulation*

As mentioned, we need to worry when individuals are allowed to self-select

- A lot of thought has to go into the *randomization phase*.
- Randomization into treatment groups has to be manipulated by someone.

Randomized Controlled Trials

Randomized controlled trials (RCT): randomization done by researcher

- Survey experiments
- Field experiments

Note: difficult to say one is strictly better than the other. Randomization can be impractical and/or unethical.

Case: Racial Discrimination in the Labor Market

Does racial discrimination exist in the labor market?

Experiment: Researchers send out resumes of fictitious job candidates in respons to newspaper ads.

- Varying only the names of the job applicants.
- Leaving all other information in the resumes unchanges.

Names were randomized between stereotypically black- and white-sounding names

- Lakisha vs. Emily
- Jamal vs. Greg

Case: Racial Discrimination in the Labor Market (2)

We use data from Kosuke Imai's repository on Github

In [14]:
```
from scipy.stats import ttest_ind
url = "https://raw.githubusercontent.com/kosukeimai/qss/master/CAUSALITY/resume.csv"
df_discr = pd.read_csv(url)
print(df_discr.head(3))
ttest_ind(*[sub.call for _,sub in df_discr.groupby('race')])
```

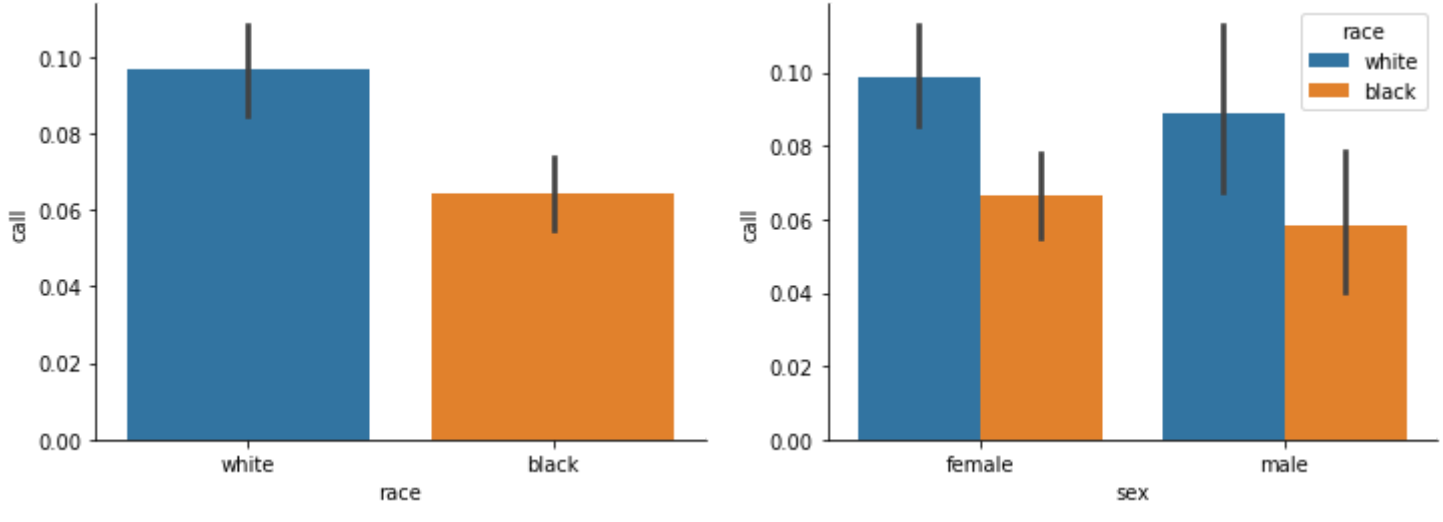
```
  firstname     sex   race  call
0   Allison  female  white     0
1   Kristen  female  white     0
2   Lakisha  female  black     0
```

Out[14]:
```
Ttest_indResult(statistic=-4.114705266723095, pvalue=3.9408025140695284e-05)
```

Case: Racial Discrimination in the Labor Market (3)

We plot the likelihood of receving a call.

In [15]:
```
f_discriminate, ax = plt.subplots(1,2,figsize=(12,4))
sns.barplot(x='race', y='call', data=df_discr, ax=ax[0])
sns.barplot(x='sex', hue='race', y='call', data=df_discr, ax=ax[1])
sns.despine(ax=ax[0])
sns.despine(ax=ax[1])
```



External & internal validity

Internal validity: Refers to the validity of causal conclusions

External validity: Refers to the extent to which the conclusions of a particular study can be generalized beyond a particular setting

Tradeoff - external vs. internal validity.

- Kosuke Imai argues that there is tradeoff - the context of experiments is too narrow and must be complemented by observational studies leveraging causal methods.
- Recent work Rachael Maeger on this.

An alternative to experiments

Quasi-experiments: randomization happens by "accident"

- Matching (*today*)
- Differences in Differences
- Regression Discontinuity Design
- Instrument variables

Matching

The what and why of matching

What - we construct counterfactual potential treated and control units.

- We *match* observations across treatment and control based on similarity.

Why - matching control for used covariates

- excludes (observable) confounders
- may improve precision of treatment estimate of experiments (less variance)

Note: An alternative to matching is to using regression - basically same idea.

Problem:

- matching does not unconfound generally!!
- unobserved factors may still confound

The how of matching

We use a set of covariates X for matching.

Two core ideas:

- We match on covariates
    - We require sufficient similarity by some metric over covarities
- We match on propensity
    - We require sufficient similar probability of treatment (prediction)

Covariate based matching

Exact matching

We match a treatment i obs. with control obs. j if

- $X_i = X_j$, i.e. they are exactly identical,
- $||X_i - X_j||_2 = 0$, i.e. zero Euclidian distance

Treatment effects

We can compute the Average Treatment Effect (ATE)

- For treatment obs. i the counterfactual outcomes $Y_i(0)$ are the average of control j where $X_j = X_i$.
- For control obs. i the counterfactual outcomes $Y_i(1)$ are the average of treatment j where $X_j = X_i$.

We can also compute treatment effects only for treament observations, known as Average Treatment Effect on the Treated (**ATT** or **ATET**).

Balance of match

What happens if some observations are not matched?

- We get biased estimates!
- We not to check whether the match is balanced
    - Problem, exact matching usually leads to very few matches.

Example of exact matching

Aim: understand whether traning program affects wages.

We have covariates and outcomes treatment and controls. (synthetic data from Scott Cunninghams's "Causal Inference - The Mixtape" book)

In [16]:
```
scuse = 'https://storage.googleapis.com/causal-inference-mixtape.appspot.com/{0}.dta'
df = pd.read_stata(scuse.format('training_example')).replace('',np.nan)
arr = df.values[:20].astype('float')
X_cntrl, y_cntrl = arr[:20,4:5], arr[:20,5]
X_treat, y_treat = arr[:10,1:2], arr[:10,2]

df.iloc[:2,[1,2,4,5]]
```

Out[16]:

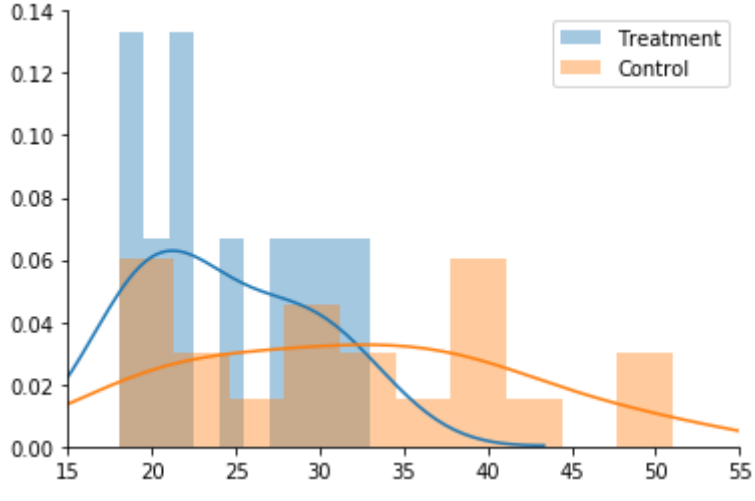
	age_treat	earnings_treat	age_control	earnings_control
0	18.0	9500	20.0	8500.0
1	29.0	12250	27.0	10075.0

Example of exact matching (2)

We have only one dimension of covariate so we can easily check the balance.

- Problem no counterfactuals for control!!

In [17]:
```
f,ax = plt.subplots()
sns.distplot(X_treat, bins=10, label='Treatment', ax=ax)
sns.distplot(X_cntrl, bins=10, label='Control', ax=ax)
ax.legend()
ax.set_xlim(15,55)
sns.despine(f)
```



Example of exact matching (3)

We can match exactly using `RadiusNeighborsRegressor` with zero radius.

- OBS: in econometrics this radius is often known as a caliper

In [19]:
```
from sklearn.neighbors import RadiusNeighborsRegressor as RNR

impute_t_exact = RNR(radius=0).fit(X_cntrl, y_cntrl).predict(X_treat)
impute_c_exact = RNR(radius=0).fit(X_treat, y_treat).predict(X_cntrl)
impute_c_exact
```

```
C:\Users\bvq720\AppData\Local\Continuum\anaconda3\lib\site-packages\sklearn\neighbors
\regression.py:327: UserWarning: One or more samples have no neighbors within specifi
ed radius; predicting NaN.
  warnings.warn(empty_warning_msg)
```

Out[19]:
```
array([10000., 11750., 10250.,    nan,    nan, 12250.,    nan, 13250.,
       11000., 12500., 13250.,    nan, 10500.,  9500.,    nan,    nan,
        9750., 12500.,    nan,    nan])
```

Exact matching (4)

We can compute unbiased estimate of ATT:

In [21]:
```
print(y_treat.mean(), y_cntrl.mean() )
diff = y_treat - impute_t_exact
print(f'ATT: {round(diff.mean(),1)} ± {round(diff.std()*1.96,1)}')
```

```
11075.0 11101.25
ATT: 1695.0 ± 646.7
```

Other covariate based matching

We can extend exact matching in several ways

- Coarsened Exact Matching:
    - where continuous variables are split into blocks
    - very popular for experiments
- Radius / Caliper matching
- Nearest neighbor matching

We can also have different metrics:

- Euclidian
- Mahalanobis distance ($= (X - \bar{X})^T COVAR(X) (X - \bar{X})$)

Note that approximate matching on covariates may introduce other biases, see Abadie and Imbenes (2011) (https://doi.org/10.1198/jbes.2009.07333).

Propensity score matching

Predicting treatment status

Alternative way of match on likelihood of treatment.

Procedure:

1. estimate a model that predicts treatment
2. match with observations of similar treatment likelihood
    - (use match function, e.g. nearest neighbor, caliper)
3. compute counterfactual outcomes for treatment and control
4. (possibly adjust for differences in observed covariates)
5. compute ATE

Uncoundedness property

Rosenbaum and Rubin (1983) (https://doi.org/10.1093/biomet/70.1.41) show that propensity score matching will unconfounded:

- can serve as an unbiased estimator of the average treatment effect
- endows non-experimental data with experimantal qualities

Critical requirement - conditional independence assumption (**CIA**):

- same as no unobserved confounders
- often CIA is violated
    - e.g. causal effect of taking education with registry data - many unobserved factors

Summary - matching

Useful tool, but requires that we know all relevant factors

- can be useful to minimize variance of experimental estimates
- problem in observational studies - often there are unobserved confounders and selection

If we think there is selection effects or endogeneity:

- Use quasi-experimental methods which can handle this, e.g. diff-in-diff or regression discontinuity

Causal trees

Average Joe

Suppose, we have credible measures of average treatment effect, τ.

Can we get personalized estimates?

- Measure whether certain groups are affected differently by our new school policy
    - e.g. boys vs. girls, natives vs. immigrants
- Some react positively to one kind of information, others to another

Beyond average Joe

Conditional Average Treatment Effects (CATE)

- Treatment effect for given characteristics x
  - $\tau(x) = \mathbb{E}[Y_i(1) - Y_i(0)|X = x]$

Methods exist, e.g. use regression analysis.

But.. **True model is unknown**..!

- May need to test model on data.
- Can lead to conclusions based on data mining (dangerous!!)

Being dishonest with you

An adaptive, data driven approach

- use all data for training decision tree
    - partitions X into categories based outcome similarity
    - enough treatment and control in each leaf
- then estimating treatment effects in partitions
    - measure treatment effects in each partition group (=leaf in tree model)

Quiz: is this different from propensity scores?

- Propensity scores has treatment assignment D_i as target.
- The adaptive approach uses outcome y_i as target.

Getting honest with you

Could we use out-of-sample intuition?

Athey and Imbens (2016) (https://doi.org/10.1073/pnas.1510489113) suggest to let data speak **honestly**:

- half of sample ($\mathcal{S}^{tr}$) for training decision tree
    - partitions X into categories based outcome similarity
    - enough treatment and control in each leaf
- other half ($\mathcal{S}^{est}$) for estimating treatment effects
    - measure treatment effects in each partition group (=leaf in tree model)

This is similar to splitting into train and test

- prevents data-leakage
- allows honest evaluation of model performance!

Core assumption

Potential outcomes and treatment assignment are unconfounded given covariates

$$D_i \perp\!\!\!\perp (Y_i(1), Y_i(0)) \mid X$$

- where $\perp\!\!\!\perp$ is a symbol for conditional independence (strong assumption!!)
- recall from earlier
    - always holds for experiments
    - or propensity scores (note: assumption cannot be tested)

Modified splitting procedure

The usual way of training decision trees is Classification And Regression Trees (CART).

- Splits leaves repeatedly based on criteria (e.g. entropy, MSE)
- We can put in restriction, e.g. depth of trees (hyperparameters)

Causal trees

- new criteria:
  - expected MSE (in hypothetical test set): $\mathbb{E}[\underbrace{(Y_i - \bar{Y}_i)^2}_{=MSE} - Y_i^2]$
  - idea: new term Y_i^2 penalizes small leaves
- note: same ranking as MSE, matters for properties

Modified splitting procedure

The usual way of training decision trees is Classification And Regression Trees (CART).

- Splits leaves repeatedly based on criteria (e.g. entropy, MSE)
- We can put in restriction, e.g. depth of trees (hyperparameters)

Causal trees

- criteria: $\mathbb{E}[(Y_i - \bar{Y}_i)^2 - {Y_i}^2]$
- note: same ranking as MSE, matters for properties

Inference

Partioning of the covariate data works like coarsened matching!

- Estimate average treatment effects locally for each group/leaf
- Corresponds to local matching!

Inference - validation

Athey and Imbens (2016) (https://doi.org/10.1073/pnas.1510489113) performs a simulation study under various scenarios.

Main take-away: **honest** outperforms **adapative** (convential CART).

Table 1. Simulation study

$N^{tr} = N^{est}$	Design 1		Design 2		Design 3	
Estimator	500	1,000	500	1,000	500	1,000
			No. of leaves			
TOT	2.9	3.2	2.9	3.5	3.6	5.4
F-A	6.1	13.1	6.3	13.0	6.2	13.0
TS-A	4.0	5.4	3.4	5.1	3.4	6.6
CT-A	4.0	5.5	3.2	3.7	3.5	5.4
F-H	6.0	12.9	6.3	13.0	6.3	13.1
TS-H	4.3	7.8	5.6	11.4	5.9	12.4
CT-H	4.2	7.6	5.6	11.4	6.1	12.5
			Infeasible MSE divided by infeasible MSE for CT-H*			
TOT-H	1.554	1.938	1.089	1.069	1.081	1.042
F-H	1.790	1.427	1.983	2.709	1.502	2.085
TS-H	0.971	0.963	1.183	1.145	1.178	1.338
			Ratio of infeasible MSE: Adaptive to honest†			
TOT-A/TOT-H		1.021		0.754		0.717
F-A/F-H		0.491		0.985		0.993
T-A/T-H		0.935		0.841		0.918
CT-A/CT-H		0.929		0.851		0.785
			Coverage of 90% confidence intervals – adaptive			
TOT-A	0.82	0.85	0.78	0.81	0.69	0.74
F-A	0.89	0.89	0.83	0.84	0.82	0.82
TS-A	0.84	0.84	0.78	0.82	0.75	0.75
CT-A	0.83	0.84	0.78	0.82	0.76	0.79
			Coverage of 90% confidence intervals – honest			
TOT-H	0.90	0.90	0.90	0.89	0.89	0.90
F-H	0.90	0.90	0.90	0.90	0.90	0.90
TS-H	0.90	0.90	0.91	0.91	0.89	0.90
CT-H	0.89	0.90	0.90	0.90	0.89	0.90

*$\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est}, \pi^{\mathrm{Estimator}}(\mathcal{S}^{tr}))/\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est}, \pi^{\mathrm{CT-H}}(\mathcal{S}^{tr}))$.
†$\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est} \cup \mathcal{S}^{tr}, \pi^{\mathrm{Estimator-A}}(\mathcal{S}^{est} \cup \mathcal{S}^{tr}))/\mathrm{MSE}_\tau(\mathcal{S}^{te}, \mathcal{S}^{est}, \pi^{\mathrm{Estimator-H}}(\mathcal{S}^{tr}))$.

Summary - causal trees

Leverage machine learning idea:

- Heterogeneity is estimated separate from treatment effects.
- New scoring function makes smaller leafs.
- Outperforms adapative procedure

Main advantage

- Structure of heterogeneity from data.
- Can be part of pre-analysis plan - only one solution (given split of data!).