Quant II

Heterogeneity and Quantile Regression

Ye Wang

April 2019

Outline

- A brief review: where we are
- Heterogeneous treatment effects
- Quantile regression

A brief review

- The proper method depends on what variables you have and what assumptions you are willing to make.

A brief review

- The proper method depends on what variables you have and what assumptions you are willing to make.
- With only Y and D in the data, you have to make the strong ignorability assumption: $D_i \perp \{Y_i(1), Y_i(0)\}$.
- Then either the Horvitz-Thompson estimator or the Hajek estimators give you a causal estimate.

A brief review

- The proper method depends on what variables you have and what assumptions you are willing to make.
- With only Y and D in the data, you have to make the strong ignorability assumption: $D_i \perp \{Y_i(1), Y_i(0)\}$.
- Then either the Horvitz-Thompson estimator or the Hajek estimators give you a causal estimate.
- With an extra variable Z, you can relax the assumption using the context of your research.

A brief review

- The proper method depends on what variables you have and what assumptions you are willing to make.
- With only Y and D in the data, you have to make the strong ignorability assumption: $D_i \perp \{Y_i(1), Y_i(0)\}$.
- Then either the Horvitz-Thompson estimator or the Hajek estimators give you a causal estimate.
- With an extra variable Z, you can relax the assumption using the context of your research.
- If Z is randomly assigned and does not directly affect Y, we can treat it as an IV and apply the Wald estimator to get LATE.
- If Z has a cutoff where the value of D changes discontinuously, RD (sharp or fuzzy) should be the proper choice.

A brief review

- The proper method depends on what variables you have and what assumptions you are willing to make.
- With only Y and D in the data, you have to make the strong ignorability assumption: $D_i \perp \{Y_i(1), Y_i(0)\}$.
- Then either the Horvitz-Thompson estimator or the Hajek estimators give you a causal estimate.
- With an extra variable Z, you can relax the assumption using the context of your research.
- If Z is randomly assigned and does not directly affect Y, we can treat it as an IV and apply the Wald estimator to get LATE.
- If Z has a cutoff where the value of D changes discontinuously, RD (sharp or fuzzy) should be the proper choice.
- If you have the time indicator T in your data, the identification assumption can be relaxed toward two different directions, sequential ignorability or fixed effects.

A brief review

- Now, suppose you have Y, D, and $\mathbf{X}$ in the dataset.

A brief review

- Now, suppose you have Y, D, and $\mathbf{X}$ in the dataset.
- You may still assume strong ignorability.
- The assumption is valid if the data are generated from a randomized trial.
- Should you control for $\mathbf{X}$ in this case?

A brief review

- Now, suppose you have Y, D, and $\mathbf{X}$ in the dataset.
- You may still assume strong ignorability.
- The assumption is valid if the data are generated from a randomized trial.
- Should you control for $\mathbf{X}$ in this case?
- Probably yes: 1. they improve the efficiency of estimation; 2. they help you understand the heterogeneity of the effects

A brief review

- Now, suppose you have Y, D, and $\mathbf{X}$ in the dataset.
- You may still assume strong ignorability.
- The assumption is valid if the data are generated from a randomized trial.
- Should you control for $\mathbf{X}$ in this case?
- Probably yes: 1. they improve the efficiency of estimation; 2. they help you understand the heterogeneity of the effects
- Obviously, you can treat $\mathbf{X}$ as confounders and use weak ignorability instead.
- Then, regression, weighting and matching.

A brief review

- Now, suppose you have Y, D, and $\mathbf{X}$ in the dataset.
- You may still assume strong ignorability.
- The assumption is valid if the data are generated from a randomized trial.
- Should you control for $\mathbf{X}$ in this case?
- Probably yes: 1. they improve the efficiency of estimation; 2. they help you understand the heterogeneity of the effects
- Obviously, you can treat $\mathbf{X}$ as confounders and use weak ignorability instead.
- Then, regression, weighting and matching.
- In this case, $\mathbf{X}$ can be confounders as well as moderators.
- The two sets of variables may not be identical.
- Conditioning on confounders eliminates the bias; conditioning on moderators allows us to investigate treatment effect heterogeneity.

Heterogeneous treatment effects

- We assume that all the observations have been properly weighted so no need to control for confounders.
- Group-mean-difference gives you an unbiased and consistent estimate of the ATE, $\tau = \frac{1}{N} \sum_{i=1}^{N} \tau_i$.

Heterogeneous treatment effects

- We assume that all the observations have been properly weighted so no need to control for confounders.
- Group-mean-difference gives you an unbiased and consistent estimate of the ATE, $\tau = \frac{1}{N} \sum_{i=1}^{N} \tau_i$.
- It is impossible to identify each τ_i, but we want to know more about their distribution.

Heterogeneous treatment effects

- We assume that all the observations have been properly weighted so no need to control for confounders.
- Group-mean-difference gives you an unbiased and consistent estimate of the ATE, $\tau = \frac{1}{N} \sum_{i=1}^{N} \tau_i$.
- It is impossible to identify each τ_i, but we want to know more about their distribution.
- There are many reasons for doing so:
  - Test your theory
  - Generalize the findings
  - Design better studies next time

Heterogeneous treatment effects

- A common idea is assume $\tau_i = f(\mathbf{X}_i) + \varepsilon_i$ where f may be unknown.

Heterogeneous treatment effects

- A common idea is assume $\tau_i = f(\mathbf{X}_i) + \varepsilon_i$ where f may be unknown.
- Notice that it is a prediction problem that has nothing to do with causality.
- Thus machine learning helps.

A detour to the basic idea of machine learning

- Machine learning provides you with a variety of prediction tools.
- We assume that the form of f is unknown and try to minimize a loss function l.
- For example, the quadratic loss function $l = E[Y_i - f(\mathbf{X}_i)]^2$.
- Given the sample, what is the best prediction function f?

A detour to the basic idea of machine learning

- Machine learning provides you with a variety of prediction tools.
- We assume that the form of f is unknown and try to minimize a loss function l.
- For example, the quadratic loss function $l = E[Y_i - f(\mathbf{X}_i)]^2$.
- Given the sample, what is the best prediction function f?
- Just set $\hat{f}(\mathbf{X}_i) = Y_i$ and it is a perfect fit.
- Yet if a new sample is drawn from the same DGP, its prediction performance will be terrible.

A detour to the basic idea of machine learning

- Machine learning provides you with a variety of prediction tools.
- We assume that the form of f is unknown and try to minimize a loss function l.
- For example, the quadratic loss function $l = E[Y_i - f(\mathbf{X}_i)]^2$.
- Given the sample, what is the best prediction function f?
- Just set $\hat{f}(\mathbf{X}_i) = Y_i$ and it is a perfect fit.
- Yet if a new sample is drawn from the same DGP, its prediction performance will be terrible.
- You are fitting the noise ε_i rather than the relationship of interest.

A detour to the basic idea of machine learning

- A shift in perspective: we want to minimize the loss in prediction rather than in approximation.
- We want I to be as small as possible for a new random sample drawn from the same DGP.

A detour to the basic idea of machine learning

- A shift in perspective: we want to minimize the loss in prediction rather than in approximation.
- We want I to be as small as possible for a new random sample drawn from the same DGP.
- A natural idea is to split the current sample, using half to estimate the model and the other half to test its performance: training set vs. test set.
- We select the model that does the best job on the test set.
- How to split? How to test? How to select?

A simple example

- Consider the linear regression models with Y, D, and $\mathbf{X}$.
- We assume that strong ignorability holds, but are unsure what moderators to include in the regression.
- Should we control for all the higher order terms and interaction terms of the moderators?

A simple example

- Consider the linear regression models with Y, D, and $\mathbf{X}$.
- We assume that strong ignorability holds, but are unsure what moderators to include in the regression.
- Should we control for all the higher order terms and interaction terms of the moderators?
- We do not want the model to be too complicated.
- Therefore, we penalize the number of moderators in the regression and modify the loss function to be:

$$I = E[Y_i - \mathbf{X}_i\beta]^2 + \lambda \sum_{i=1}^{N} ||\mathbf{X}_i||$$

A simple example

- For each possible λ (from 1 to 100, for example), we split the sample into five folds.
- We use four folds to minimize the loss under the given λ, and apply the fitted model to calculate the value of l on the remaining fold.
- We then select the λ that gives us the smallest l on the test set.

A simple example

- For each possible λ (from 1 to 100, for example), we split the sample into five folds.
- We use four folds to minimize the loss under the given λ, and apply the fitted model to calculate the value of I on the remaining fold.
- We then select the λ that gives us the smallest I on the test set.
- It is called cross-validation.
- The idea is that the error term ε_i in the training set is independent to that in the test set.
- Hence, a model that fits the noise on the training set won't do well on the test set.
- The chosen $\hat{f}$ should be roughly uncorrelated to the noise.

A simple example

- Result from the algorithm above is called LASSO (Least Absolute Shrinkage and Selection Operator).
- It selects "features" that have the strongest prediction power into the model.
- Compared to the result of OLS, the coefficients of strong predictors are larger and the coefficients of weak predictors become zero.
- We will have a clearer picture of what moderators better explain the variation of the treatment effect.

From tree to forest

- LASSO still assumes linearity.

From tree to forest

- LASSO still assumes linearity.
- A more flexible method is called tree, or CART (Classification and Regression Tree).
- The tree algorithm automatically classifies observations into homogeneous blocks.

From tree to forest

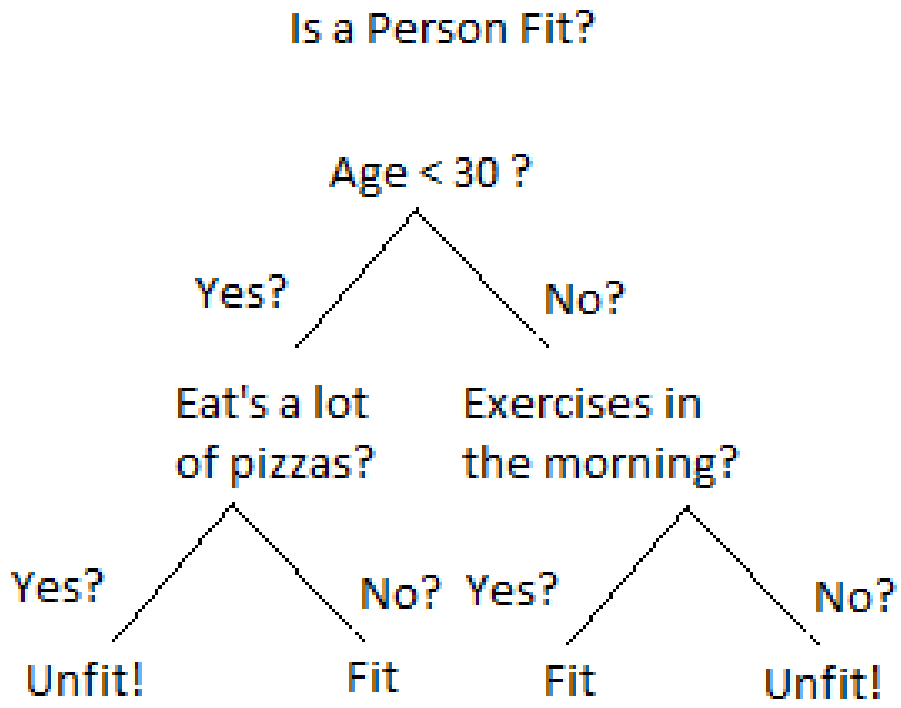
- LASSO still assumes linearity.
- A more flexible method is called tree, or CART (Classification and Regression Tree).
- The tree algorithm automatically classifies observations into homogeneous blocks.
- In the pure prediction case, we have only Y and $\mathbf{X}$.
- The estimate for each observation i is the average outcome in the block b it belongs to.
- Again, we want to balance fitness and complexity:

$$I = E[Y_i - \hat{Y}_i]^2 + \lambda|B| = E_b[E[Y_i - \bar{Y}_b]^2] + \lambda|B|$$

where B is the number of blocks.

From tree to forest

- We rely on a recursive algorithm to obtain the best partition.
- In each step, we find the "cut" that increases fitness the most.



- The algorithm proceeds on the training set until we have no more than 5 observations in each leave.
- Now we have maximized fitness.

From tree to forest

- To reduce complexity, we "prune" the tree reversely with the given λ.
- Again, cross-validation on the test set selects the λ that minimizes the loss.

From tree to forest

- To reduce complexity, we "prune" the tree reversely with the given λ.
- Again, cross-validation on the test set selects the λ that minimizes the loss.
- Recusive algorithm finds the optimum at each step, but the result may not be a global optimum as well.
- Consequently, tree is highly unstable.

From tree to forest

- To reduce complexity, we "prune" the tree reversely with the given λ.
- Again, cross-validation on the test set selects the λ that minimizes the loss.
- Recusive algorithm finds the optimum at each step, but the result may not be a global optimum as well.
- Consequently, tree is highly unstable.
- We may use the evolutionary algorithm to find the optimal tree partition.
- Or we average over a lot of trees— boosting or forest.

Causal tree

- To study HTE, we can divide observations into the blocks to optimize our prediction of the treatment effect.
- There is a problem: τ_i is not observable.

Causal tree

- To study HTE, we can divide observations into the blocks to optimize our prediction of the treatment effect.
- There is a problem: τ_i is not observable.
- Athey and Imbens (2016): Causal tree.
- Intuitively, we reward the heterogeneity across leaves and penalize the heterogeneity within leaves.

Causal tree

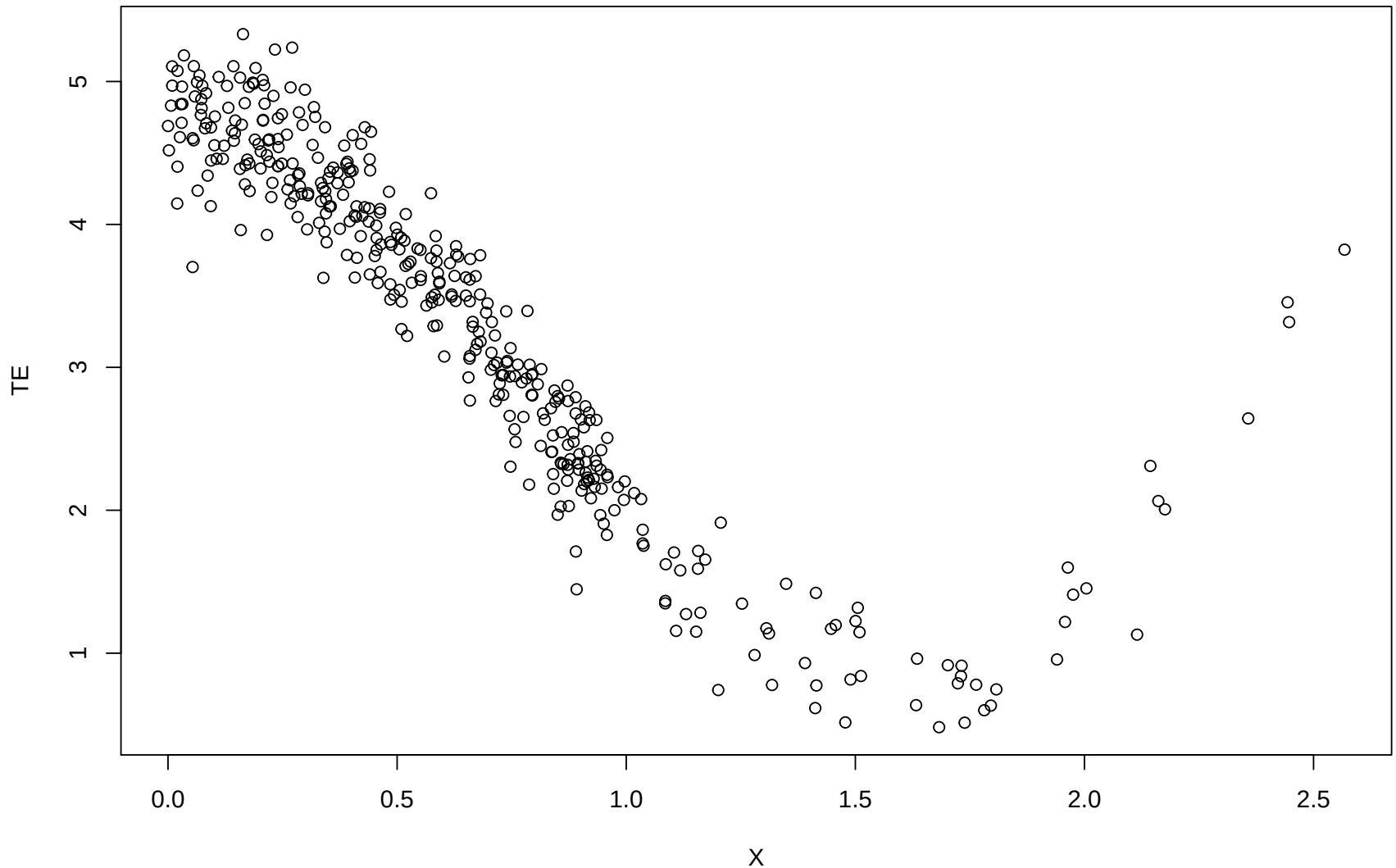
- To study HTE, we can divide observations into the blocks to optimize our prediction of the treatment effect.
- There is a problem: τ_i is not observable.
- Athey and Imbens (2016): Causal tree.
- Intuitively, we reward the heterogeneity across leaves and penalize the heterogeneity within leaves.
- They also suggest an "honest approach."
- Split the sample into three parts: the training set, the test set, and the estimation set.
- We use the first two sets to generate the optimal partition, and the last to estimate effects on each leave.
- It reduces bias and makes inference much easier.

Causal forest

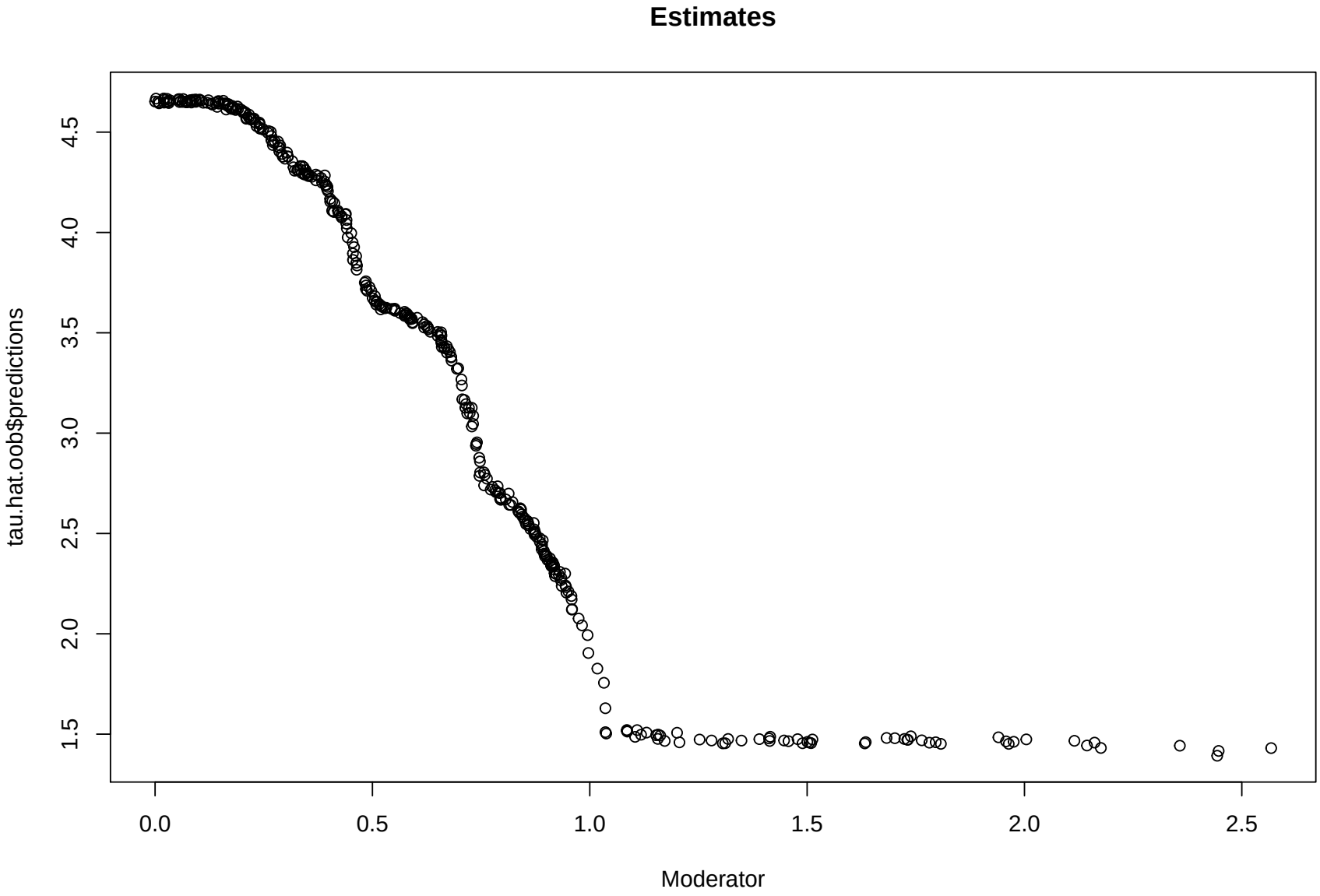
- We have the causal forest when combining results from various causal trees.
- Wager and Athey (2018) and Athey et al. (2019) establish the theory behind the algorithm.
- Forest also avoids lowering efficiency.
- The idea can be extended to estimate any local parameter (generalized causal forest).

Causal forest

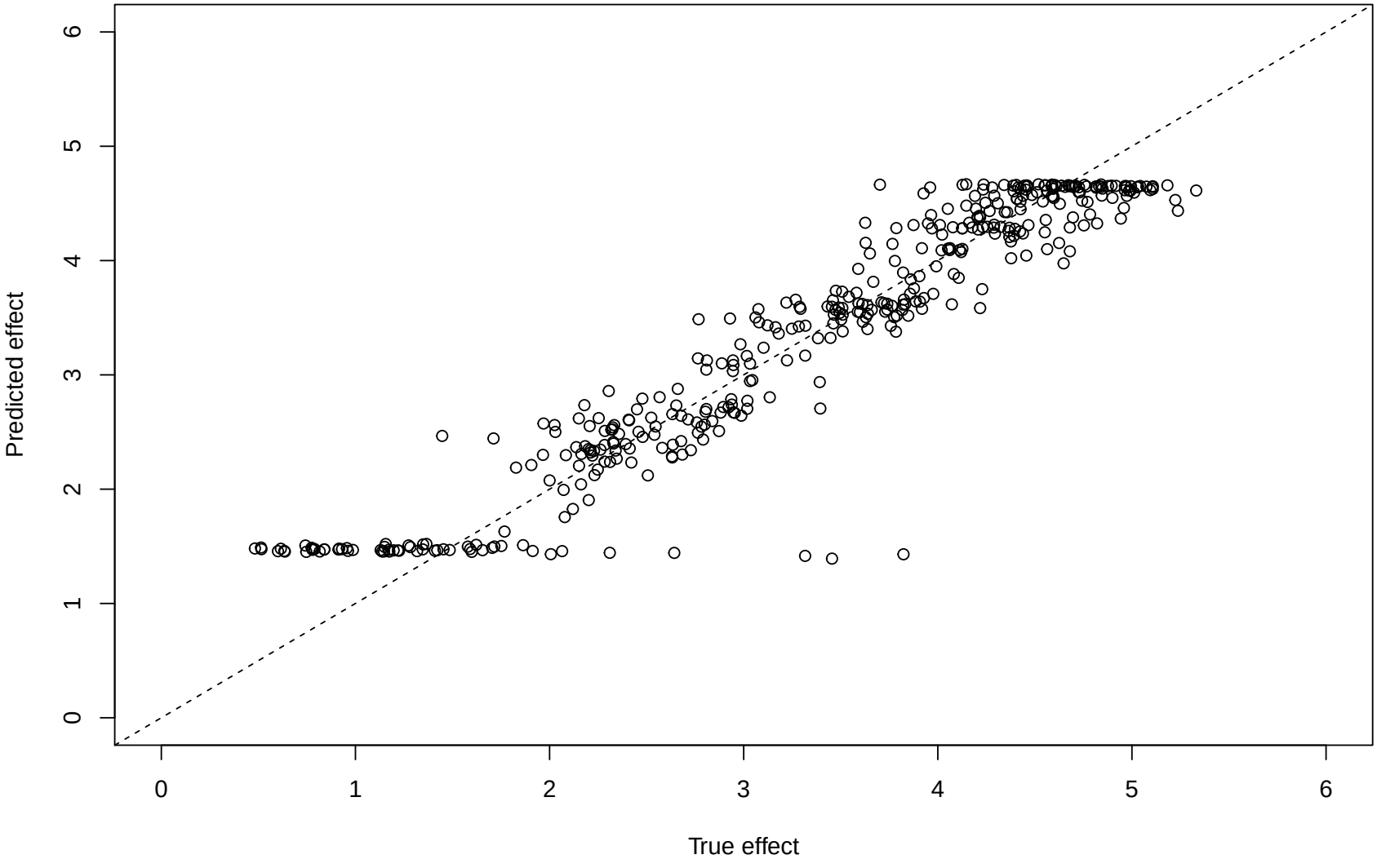
```
## Loading required package: grf
```



Causal forest



Causal forest



Ensemble methods

- Whether linear methods or tree-based methods perform better depends on the context.

Ensemble methods

- Whether linear methods or tree-based methods perform better depends on the context.
- A natural idea is to combine them together.
- Grimmer (2017):
  - Predict Y_i using M different methods and generate predicted values $(\hat{Y}_{i1}, \hat{Y}_{i2}, \ldots, \hat{Y}_{iM})$.
  - Regress Y_i on the M predictions to get their weights.
  - The ATE estimate equals to the weighted average of ATE estimates from each method.
- Ratkovic and Tingley (2018): direct estimation
  - Generate features and select them based on their correlation with the outcome.
  - Fit a sparse Bayesian model to predict the counterfactual.

Ensemble methods

- More general frameworks:
  - X-learner in Kunzel et al. (2019);
  - R-learner in Nie and Wager (2019);
  - TMLE in Van der Laan and Ross (2011);
  - Double machine learning in Belloni et al. (2017).

Ensemble methods

- Let's look at one example: R-learner.

1. Fit $\hat{m}(x)$ and $\hat{e}(x)$ via methods tuned for optimal predictive accuracy, then
2. Estimate treatment effects via a plug-in version of (5), where $\hat{e}^{(-i)}(X_i)$, etc., denote held-out predictions, i.e., predictions made without using the i-th training example,[1]

$$\hat{\tau}(\cdot) = \text{argmin}_\tau \left\{ \widehat{L}_n \left(\tau(\cdot) \right) + \Lambda_n \left(\tau(\cdot) \right) \right\},$$
$$\widehat{L}_n \left(\tau(\cdot) \right) = \frac{1}{n} \sum_{i=1}^{n} \left(\left(Y_i - \hat{m}^{(-i)}(X_i) \right) - \left(W_i - \hat{e}^{(-i)}(X_i) \right) \tau(X_i) \right)^2. \quad (6)$$

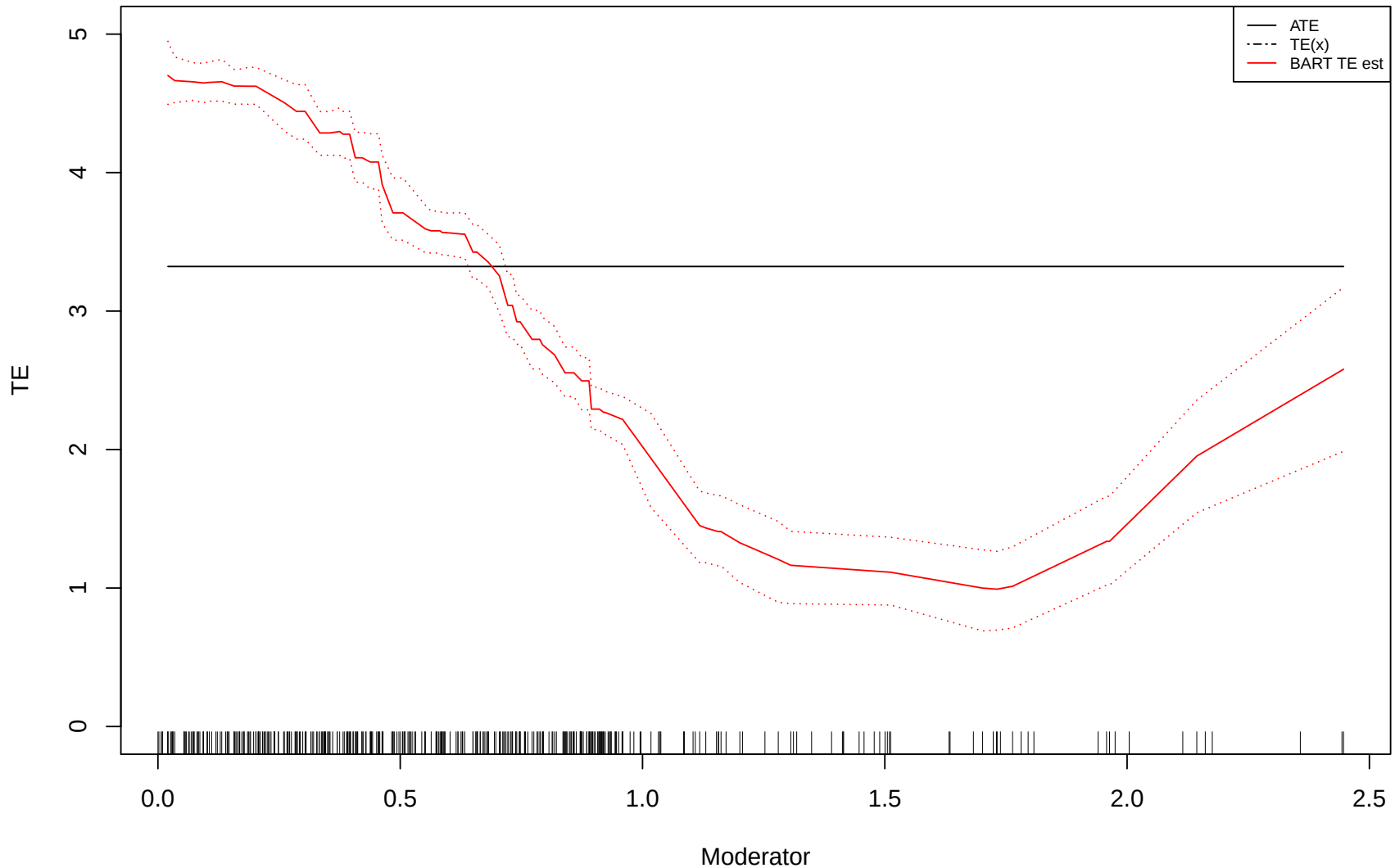
- We first non-parametrically estimate the propensity score and response surface.
- Then another algorithm is used to estimate $\tau_i(x)$.

T-learner vs. S-learner

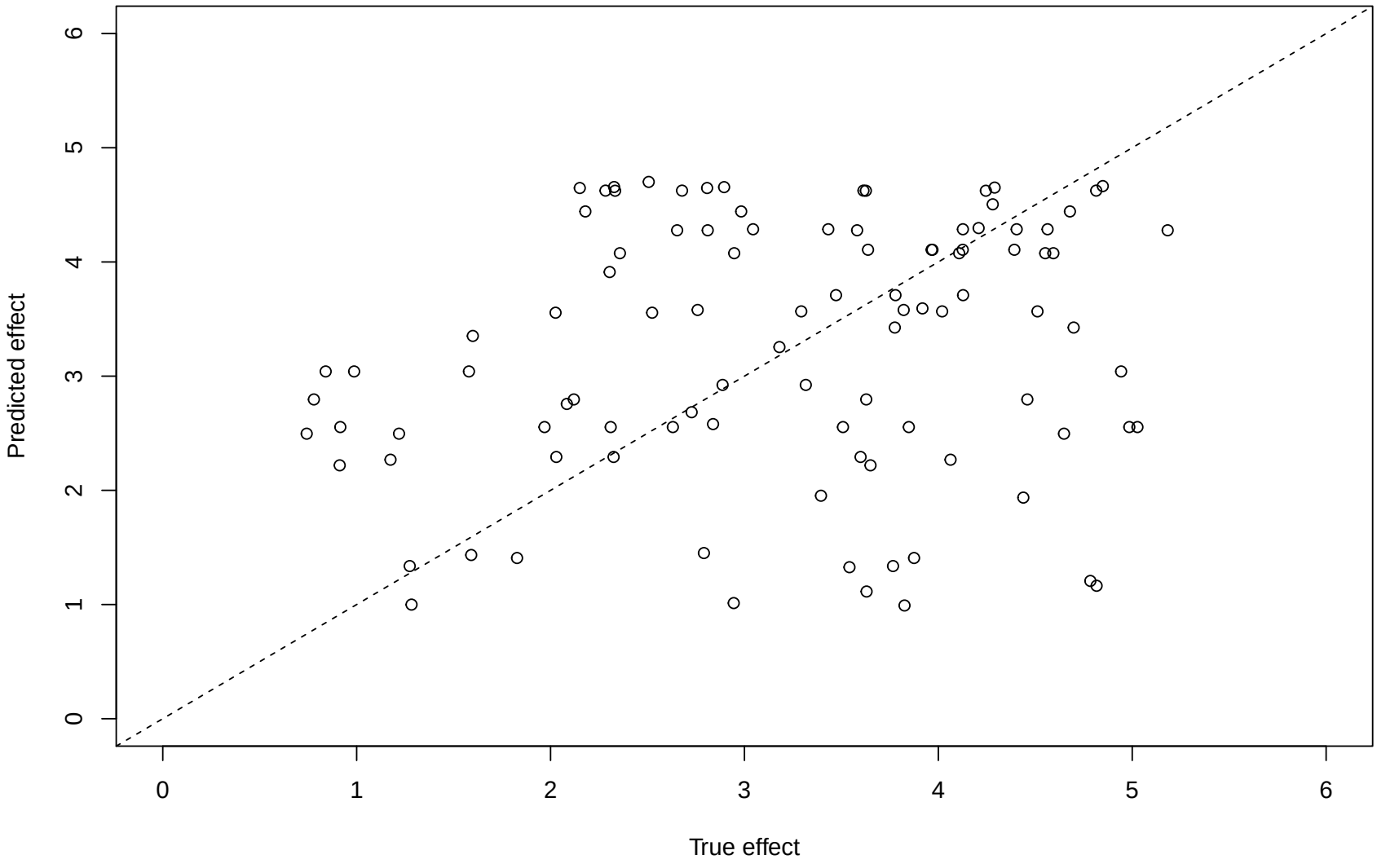
- All the previous methods run on the whole sample (S-learner).
- We can fit separate models for the treatment group and control group (T-learner).
- A popular approach is BART (Bayesian Additive Regression Trees).
- It is a Bayesian approach and the CIs are generated from the posterior.
- The T-learner is less stable as two response surfaces are fitted separately.

T-learner vs. S-learner

```
## Warning in rug(X): some values will be clipped
```



T-learner vs. S-learner



Structural estimation of the HTE

- Modern approaches remain agnostic about the source of heterogeneity.
- But we may add more "structures" into the model.
- For example, people may select into the program because they know its benefits.
- See the work of Heckman and Vytlacil, and the recent review by Xie and Zhou.
- Some simple structures may significantly enhance the model's performance.

Quantile regression

- Quantile regression belongs to the class of moment estimators.
- Even though the moment function is not smooth, we still have good asymptotic results.
- Intuitively, only the expectation of the moment function matters.
- Variance and CI estimates can be obtained via class results on moment estimators.

Quantile regression

```
##
## Attaching package: 'SparseM'

## The following object is masked from 'package:base':
##
##     backsolve

##
## Call: rq(formula = foodexp ~ income, tau = 0.5, data = e
##
## tau: [1] 0.5
##
## Coefficients:
##             coefficients lower bd  upper bd
## (Intercept)  81.48225     53.25915 114.01156
## income        0.56018      0.48702   0.60199
```

Counterfactual estimation

- The idea of counterfactual estimation (Chernozhukov et al. 2013) is an extension of the Oaxaca-Blinder decomposition.

Counterfactual estimation

- The idea of counterfactual estimation (Chernozhukov et al. 2013) is an extension of the Oaxaca-Blinder decomposition.
- Suppose we are interested in the wage distribution for male and female workers on the labor market.
- We can observe the distribution of characteristics for male and female workers: $F_{\mathbf{X_m}}$ and $F_{\mathbf{X_f}}$.
- We can also see the conditional distribution of wage for both genders given the characteristics: $F_{Y_m|\mathbf{X_m}}$ and $F_{Y_f|\mathbf{X_f}}$.

Counterfactual estimation

- The difference in wage between male and female workers equals to:

$$\begin{aligned}
&\int_{\mathbf{X_m}} F_{Y_m|\mathbf{X_m}} dF_{\mathbf{X_m}} - \int_{\mathbf{X_f}} F_{Y_f|\mathbf{X_f}} dF_{\mathbf{X_f}} \\
&= \int_{\mathbf{X_m}} F_{Y_m|\mathbf{X_m}} dF_{\mathbf{X_m}} - \int_{\mathbf{X_m}} F_{Y_m|\mathbf{X_m}} dF_{\mathbf{X_f}} \\
&\quad + \int_{\mathbf{X_m}} F_{Y_m|\mathbf{X_m}} dF_{\mathbf{X_f}} - \int_{\mathbf{X_f}} F_{Y_f|\mathbf{X_f}} dF_{\mathbf{X_f}}
\end{aligned}$$

- The first term is the composition effect and the second is the structure effect.
- The smart-zero term is called "counterfactual."
- The authors propose the distributio regression to estimate the counterfactual term.
- These effects are causal when "groups" are independent to the potential outcomes given the characteristics.

Counterfactual estimation

```
## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning: glm.fit: fitted probabilities numerically 0 or

## Warning in regularize.values(x, y, ties, missing(ties)):
## unique 'x' values

##
## Conditional Model:                          logit
## Number of regressions estimated:             100
##
## The variance has not been computed.
## Do not turn the option noboot on if you want to compute
##
## No. of obs. in the reference group:         1407
## No. of obs. in the counterfactual group: 459
##
##
```