Model Free Predictive Inference

Larry Wasserman
Carnegie Mellon University

With: Yotam Hechtlinger, Nic DalMasso, Jing Lei, Max G'Sell, Alessandro Rinaldo, Jamie Robins, Mauricio Sadinle, Ryan Tibshirani, JaeHyeok Shin, Robin Dunn

OUTLINE

OUTLINE

"All models are wrong ..." George Box

OUTLINE

"All models are wrong ..." George Box

"Use models but don't believe them" Tukey?

OUTLINE

"All models are wrong ..." George Box

"Use models but don't believe them" Tukey?

> *"Investigators who use [regression] are not paying adequate attention to the connection - if any - between the models and the phenomena they are studying. ... By the time the models are deployed, the scientific position is nearly hopeless. Reliance on models in such cases is Panglossian ..."*
> *—David Freedman*

OUTLINE

"All models are wrong ..." George Box

"Use models but don't believe them" Tukey?

> *"Investigators who use [regression] are not paying adequate attention to the connection - if any - between the models and the phenomena they are studying. ... By the time the models are deployed, the scientific position is nearly hopeless. Reliance on models in such cases is Panglossian ..."*
> *—David Freedman*

By focusing on predictive quantities, we can do many things with (almost) no assumptions.
Not assuming: linearity, constant variance, incoherence, sparsity, etc.

Outline

Outline

- Part I: High-dimensional regression and classification.

Outline

- Part I: High-dimensional regression and classification.
- Part II: Clustering.

Outline

- Part I: High-dimensional regression and classification.
- Part II: Clustering.
- Part III: Random effects.

High Dimensional Regression

Let $(X_1, Y_1), \ldots, (X_n, Y_n) \sim P$. $X_i \in \mathbb{R}^d$, $Y_i \in \mathbb{R}$.

High Dimensional Regression

Let $(X_1, Y_1), \ldots, (X_n, Y_n) \sim P$. $X_i \in \mathbb{R}^d$, $Y_i \in \mathbb{R}$.

No assumptions (except iid).

High Dimensional Regression

Let $(X_1, Y_1), \ldots, (X_n, Y_n) \sim P$. $X_i \in \mathbb{R}^d$, $Y_i \in \mathbb{R}$.

No assumptions (except iid).

Linear regression:
The projection interpretation: β is the best linear predictor:

$$\beta_* = \text{argmin}\, \mathbb{E}[(Y - \beta^T X)^2] = \Lambda^{-1}\alpha$$

where $\Lambda = \mathbb{E}[XX^T]$ and $\alpha = \mathbb{E}[YX]$. So $\beta_* = g(\text{vec}(\Lambda), \alpha)$.

High Dimensional Regression

Let $(X_1, Y_1), \ldots, (X_n, Y_n) \sim P$. $X_i \in \mathbb{R}^d$, $Y_i \in \mathbb{R}$.

No assumptions (except iid).

Linear regression:
The projection interpretation: β is the best linear predictor:

$$\beta_* = \text{argmin}\, \mathbb{E}[(Y - \beta^T X)^2] = \Lambda^{-1}\alpha$$

where $\Lambda = \mathbb{E}[XX^T]$ and $\alpha = \mathbb{E}[YX]$. So $\beta_* = g(\text{vec}(\Lambda), \alpha)$.

Inference? Use CLT or bootstrap to approximate the distribution of $\sqrt{n}(\widehat{\beta} - \beta)$?

High Dimensional Regression

Let $(X_1, Y_1), \ldots, (X_n, Y_n) \sim P$. $X_i \in \mathbb{R}^d$, $Y_i \in \mathbb{R}$.

No assumptions (except iid).

Linear regression:
The projection interpretation: β is the best linear predictor:

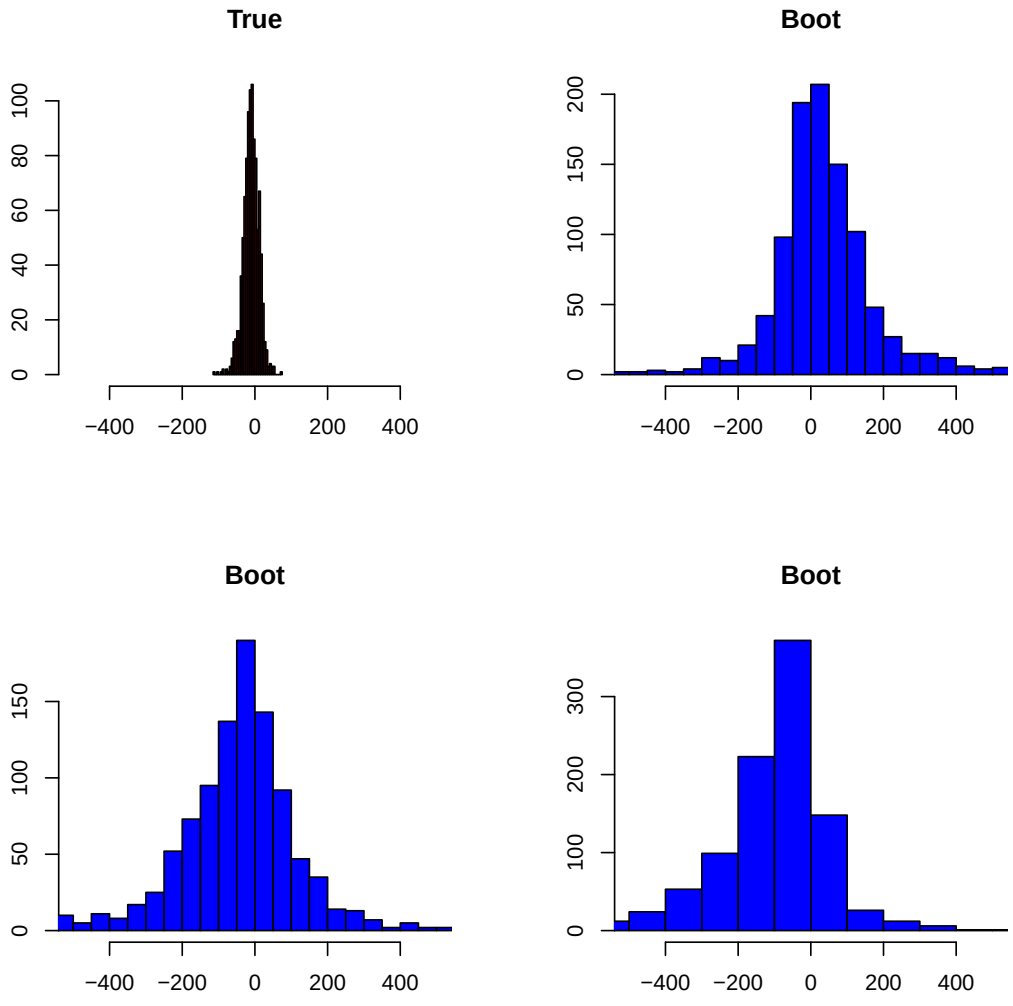
$$\beta_* = \text{argmin}\, \mathbb{E}[(Y - \beta^T X)^2] = \Lambda^{-1}\alpha$$

where $\Lambda = \mathbb{E}[XX^T]$ and $\alpha = \mathbb{E}[YX]$. So $\beta_* = g(\text{vec}(\Lambda), \alpha)$.

Inference? Use CLT or bootstrap to approximate the distribution of $\sqrt{n}(\widehat{\beta} - \beta)$?

Unfortunately, they are quite poor.

Example: n = 100, d = 50



d Large

Rinaldo, Wasserman, G'Sell (2018). Berry-Esseen bound:

d Large

Rinaldo, Wasserman, G'Sell (2018). Berry-Esseen bound:

$$\Psi_n = \sup_{P \in \mathcal{P}_n} \sup_t |P(\sqrt{n}(\widehat{\beta} - \beta) \leq t) - \text{Normal}| \leq \Delta_1 + \Delta_2 + \Delta_3$$

$$\Delta_{1,n} \approx \left(\frac{d^2}{n}\right)^{1/6} \quad \text{linear terms}$$

$$\Delta_{2,n} \approx \left(\frac{d^4}{n}\right)^{1/2} \quad \text{non} - \text{linear terms}$$

$$\Delta_{3,n} \approx \left(\frac{d^5}{n}\right)^{1/6} \quad \text{covariance estimation (sandwich)}$$

Bad News

$\Delta_{3,n} \approx \left(\frac{d^5}{n}\right)^{1/6}$ is the worse term. (Due to estimating the covariance of $\widehat{\beta}$ i.e. sandwich estimator.)

Bad News

$\Delta_{3,n} \approx \left(\frac{d^5}{n}\right)^{1/6}$ is the worse term. (Due to estimating the covariance of $\widehat{\beta}$ i.e. sandwich estimator.)

Need: $d = o(n^{1/5}) \approx \textit{constant}$!

Bad News

$\Delta_{3,n} \approx \left(\frac{d^5}{n}\right)^{1/6}$ is the worse term. (Due to estimating the covariance of $\widehat{\beta}$ i.e. sandwich estimator.)

Need: $d = o(n^{1/5}) \approx \textit{constant}$!

Similar bound for bootstrap.

Bad News

$\Delta_{3,n} \approx \left(\frac{d^5}{n}\right)^{1/6}$ is the worse term. (Due to estimating the covariance of $\widehat{\beta}$ i.e. sandwich estimator.)

Need: $d = o(n^{1/5}) \approx \textit{constant}$!

Similar bound for bootstrap.

It's bad. Easily confirmed by simulation.

Bad News

$\Delta_{3,n} \approx \left(\frac{d^5}{n}\right)^{1/6}$ is the worse term. (Due to estimating the covariance of $\widehat{\beta}$ i.e. sandwich estimator.)

Need: $d = o(n^{1/5}) \approx$ *constant*!

Similar bound for bootstrap.

It's bad. Easily confirmed by simulation.

We think it is tight (but have not proved it). Lower bounds for Berry-Esseen theorems are rare.

Bad News

$\Delta_{3,n} \approx \left(\frac{d^5}{n}\right)^{1/6}$ is the worse term. (Due to estimating the covariance of $\widehat{\beta}$ i.e. sandwich estimator.)

Need: $d = o(n^{1/5}) \approx$ *constant*!

Similar bound for bootstrap.

It's bad. Easily confirmed by simulation.

We think it is tight (but have not proved it). Lower bounds for Berry-Esseen theorems are rare.

Similar conclusion in El Karoui and Purdom (2016).

Solutions

Solutions

1. Change the parameter (LOCO)

Solutions

1. Change the parameter (LOCO)

2. Focus on prediction (conformalization).

LOCO (Leave Out COvariates) Inference

LOCO (Leave Out COvariates) Inference

Split data $\mathcal{D}_1$, $\mathcal{D}_2$.

LOCO (Leave Out COvariates) Inference

Split data $\mathcal{D}_1$, $\mathcal{D}_2$.

Fit $\widehat{m}$ (any regression estimator) on $\mathcal{D}_1$.

LOCO (Leave Out COvariates) Inference

Split data $\mathcal{D}_1$, $\mathcal{D}_2$.

Fit $\widehat{m}$ (any regression estimator) on $\mathcal{D}_1$.

Drop variable j, re-run and get $\widehat{m}^{(-j)}$.

LOCO (Leave Out COvariates) Inference

Split data $\mathcal{D}_1$, $\mathcal{D}_2$.

Fit $\widehat{m}$ (any regression estimator) on $\mathcal{D}_1$.

Drop variable j, re-run and get $\widehat{m}^{(-j)}$.

Use $\mathcal{D}_2$ to get exact confidence interval for the conditional quantity

$$\theta_j = \phi\left(|Y - \widehat{m}^{(-j)}(X)| - |Y - \widehat{m}(X)| \;\middle|\; \mathcal{D}_1 \right)$$

where ϕ is median or mean.

LOCO (Leave Out COvariates) Inference

Split data $\mathcal{D}_1$, $\mathcal{D}_2$.

Fit $\widehat{m}$ (any regression estimator) on $\mathcal{D}_1$.

Drop variable j, re-run and get $\widehat{m}^{(-j)}$.

Use $\mathcal{D}_2$ to get exact confidence interval for the conditional quantity

$$\theta_j = \phi\left(|Y - \widehat{m}^{(-j)}(X)| - |Y - \widehat{m}(X)| \;\middle|\; \mathcal{D}_1 \right)$$

where ϕ is median or mean.

$\widehat{m}$ can be: lasso, random forest, neural net, ...

LOCO (Leave Out COvariates) Inference

Split data $\mathcal{D}_1$, $\mathcal{D}_2$.

Fit $\widehat{m}$ (any regression estimator) on $\mathcal{D}_1$.

Drop variable j, re-run and get $\widehat{m}^{(-j)}$.

Use $\mathcal{D}_2$ to get exact confidence interval for the conditional quantity

$$\theta_j = \phi\left(|Y - \widehat{m}^{(-j)}(X)| - |Y - \widehat{m}(X)| \;\middle|\; \mathcal{D}_1 \right)$$

where ϕ is median or mean.

$\widehat{m}$ can be: lasso, random forest, neural net, ...

Exact for median: For mean: Berry-Esseen bound: $1/\sqrt{n}$

LOCO (Leave Out COvariates) Inference

Split data $\mathcal{D}_1$, $\mathcal{D}_2$.

Fit $\widehat{m}$ (any regression estimator) on $\mathcal{D}_1$.

Drop variable j, re-run and get $\widehat{m}^{(-j)}$.

Use $\mathcal{D}_2$ to get exact confidence interval for the conditional quantity

$$\theta_j = \phi\left(|Y - \widehat{m}^{(-j)}(X)| - |Y - \widehat{m}(X)| \;\middle|\; \mathcal{D}_1 \right)$$

where ϕ is median or mean.

$\widehat{m}$ can be: lasso, random forest, neural net, ...

Exact for median: For mean: Berry-Esseen bound: $1/\sqrt{n}$

Similar to: Brieman permutation importance, Mentch and Hooker (2016), Abbasi-Asl and Yu (2017), Davies (2018)

Bias

This is a conditional quantity. (Randomness is in X, Y not in the estimator which is fixed.)

Bias

This is a conditional quantity. (Randomness is in X, Y not in the estimator which is fixed.)

The corresponding marginal (population) quantity, namely,

$$\psi = \mathbb{E}[|Y - m^{(-j)}(X)| - |Y - m(X)|]$$

or

$$\psi = \mathbb{E}[(Y - m^{(-j)}(X))^2 - (Y - m(X))^2]$$

is not estimable in the model free framework. (Similarly, for conditional independence.)

Bias

As an estimate of ψ, the plug-in is biased.

Bias

As an estimate of ψ, the plug-in is biased.

LOCO based on the 1-nn estimate (DeVroye, Gyorfi, Lugosi, Walk 2018) has low bias but very low power.

Bias

As an estimate of ψ, the plug-in is biased.

LOCO based on the 1-nn estimate (DeVroye, Gyorfi, Lugosi, Walk 2018) has low bias but very low power.

Subtracting the first order influence function (Williamson, Gilbert, Simon, Carone 2017) is reasonable but still requires many assumptions.

Bias

As an estimate of ψ, the plug-in is biased.

LOCO based on the 1-nn estimate (DeVroye, Gyorfi, Lugosi, Walk 2018) has low bias but very low power.

Subtracting the first order influence function (Williamson, Gilbert, Simon, Carone 2017) is reasonable but still requires many assumptions.

Also, it is degenerate under the null.
(But we (JR and LW) are looking into higher-order statistics.)

Bias

As an estimate of ψ, the plug-in is biased.

LOCO based on the 1-nn estimate (DeVroye, Gyorfi, Lugosi, Walk 2018) has low bias but very low power.

Subtracting the first order influence function (Williamson, Gilbert, Simon, Carone 2017) is reasonable but still requires many assumptions.

Also, it is degenerate under the null.
(But we (JR and LW) are looking into higher-order statistics.)

Ultimately, we have to regard it as exact inference for a conditional quantity .

Examples of LOCO: n=200, d=500 (RT)

Example: $n = 200$, $d = 500$, with data model $Y = \beta^T X + \epsilon$, such that $X \sim N(0, I_d)$, $\epsilon \sim N(0, 1)$, and

$$\beta_j \begin{cases} \sim N(0, 2) & j = 1, \ldots, 5 \\ = 0 & \text{otherwise} \end{cases}$$

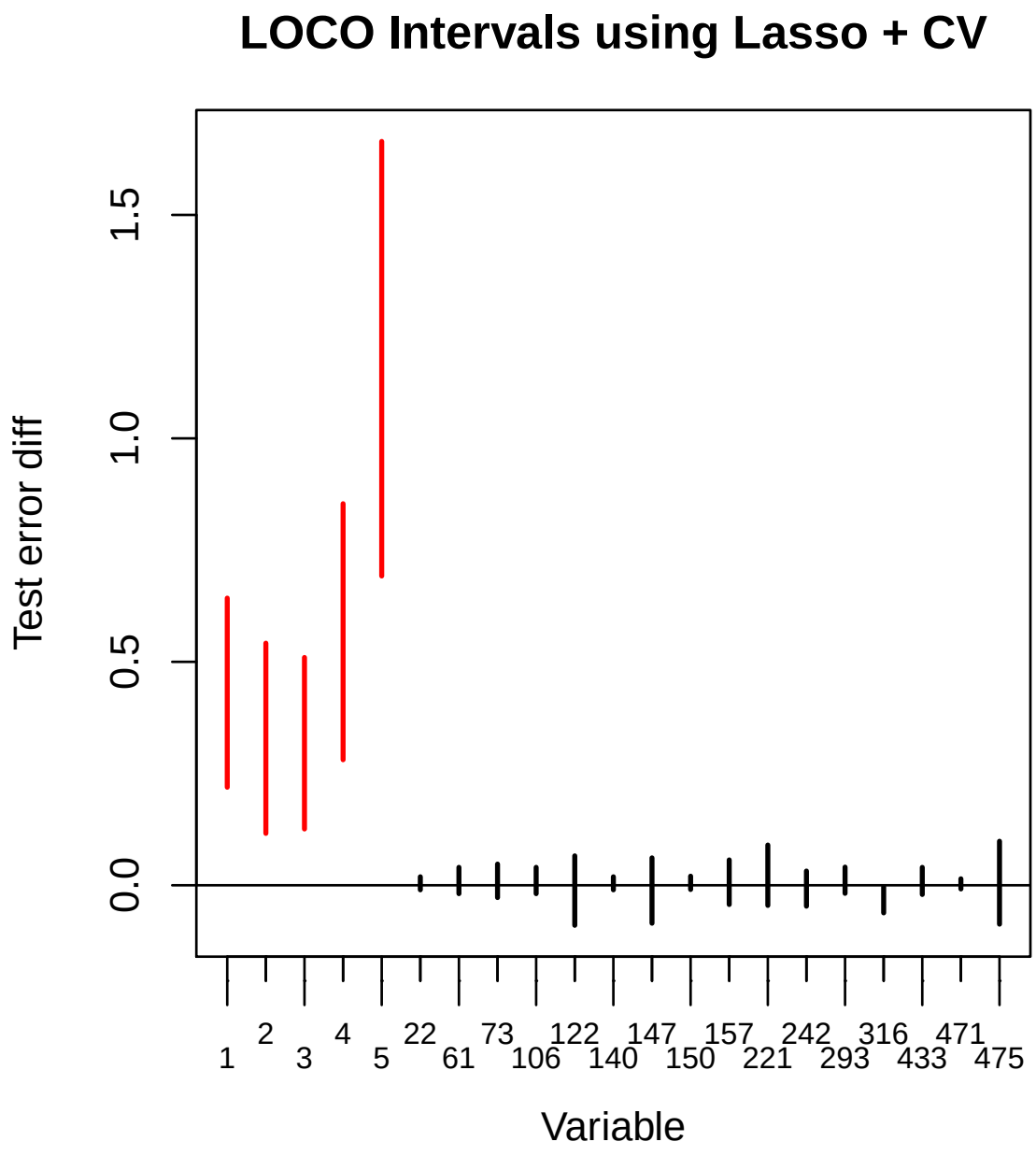
- Algorithm is the lasso, with 5-fold CV and 1se rule to select λ
- Compute an interval for

$$\theta_j(D_1) = \mathrm{med}\Big(|Y - \widehat{m}_{n_1}^{-j}(X)| - |Y - \widehat{m}_{n_1}(X)| \,\Big|\, D_1 \Big)$$

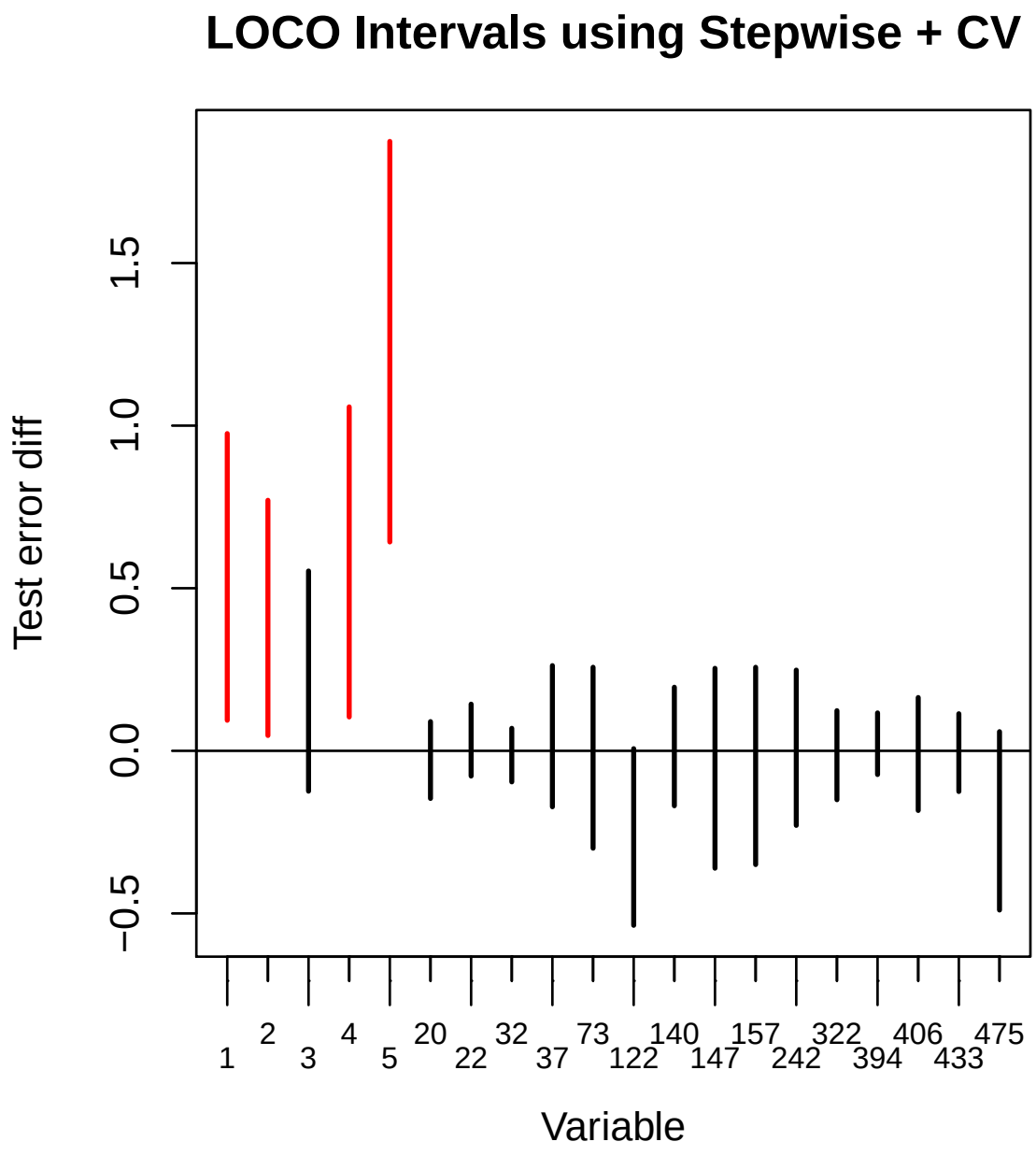
  for each j in lasso active set
- Use Bonferroni correction: if s variables are selected, then we compute each LOCO interval at level $1 - \alpha/s$

Note: slides marked RT are courtesy of Ryan Tibshirani.

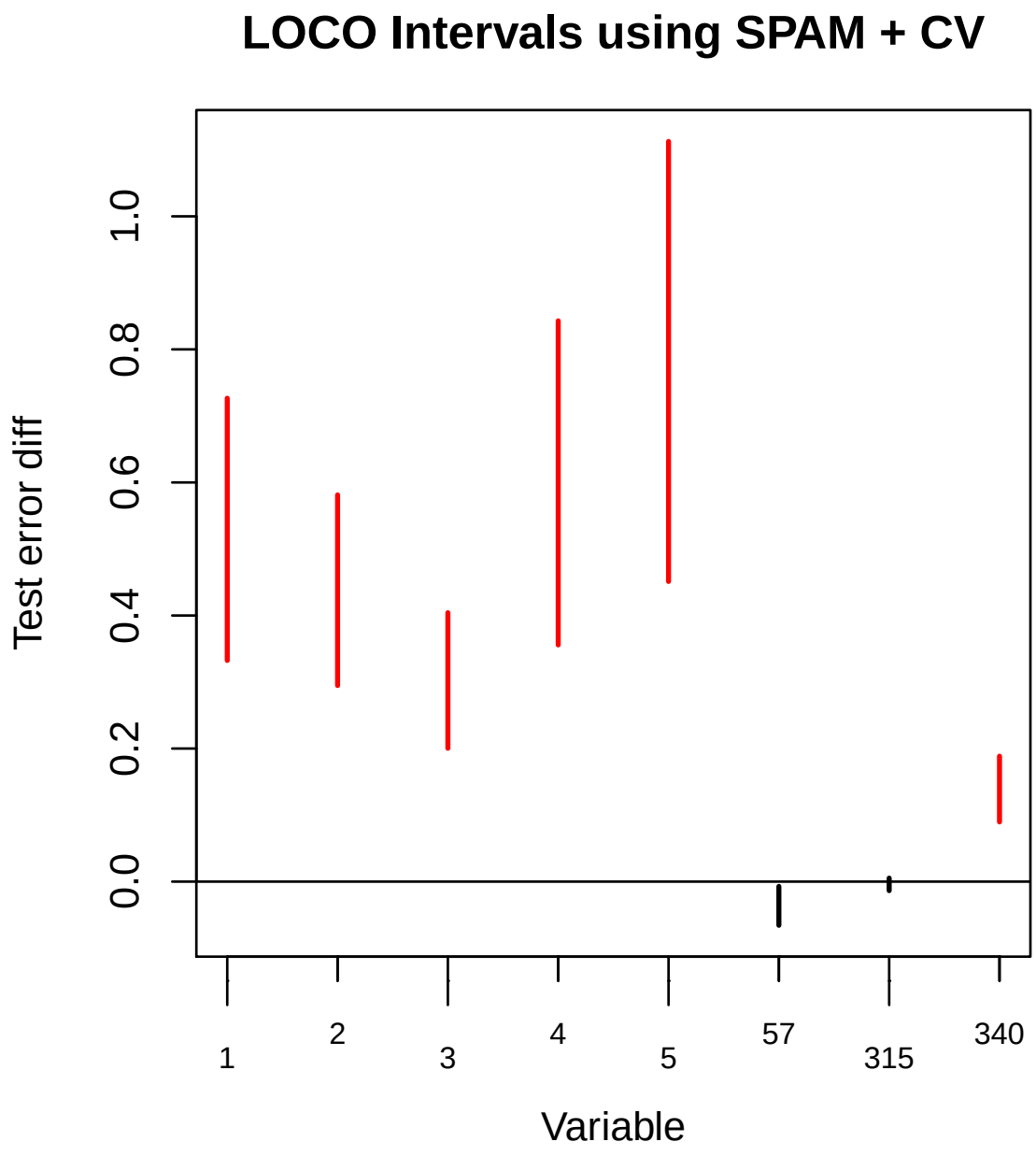
Examples of LOCO



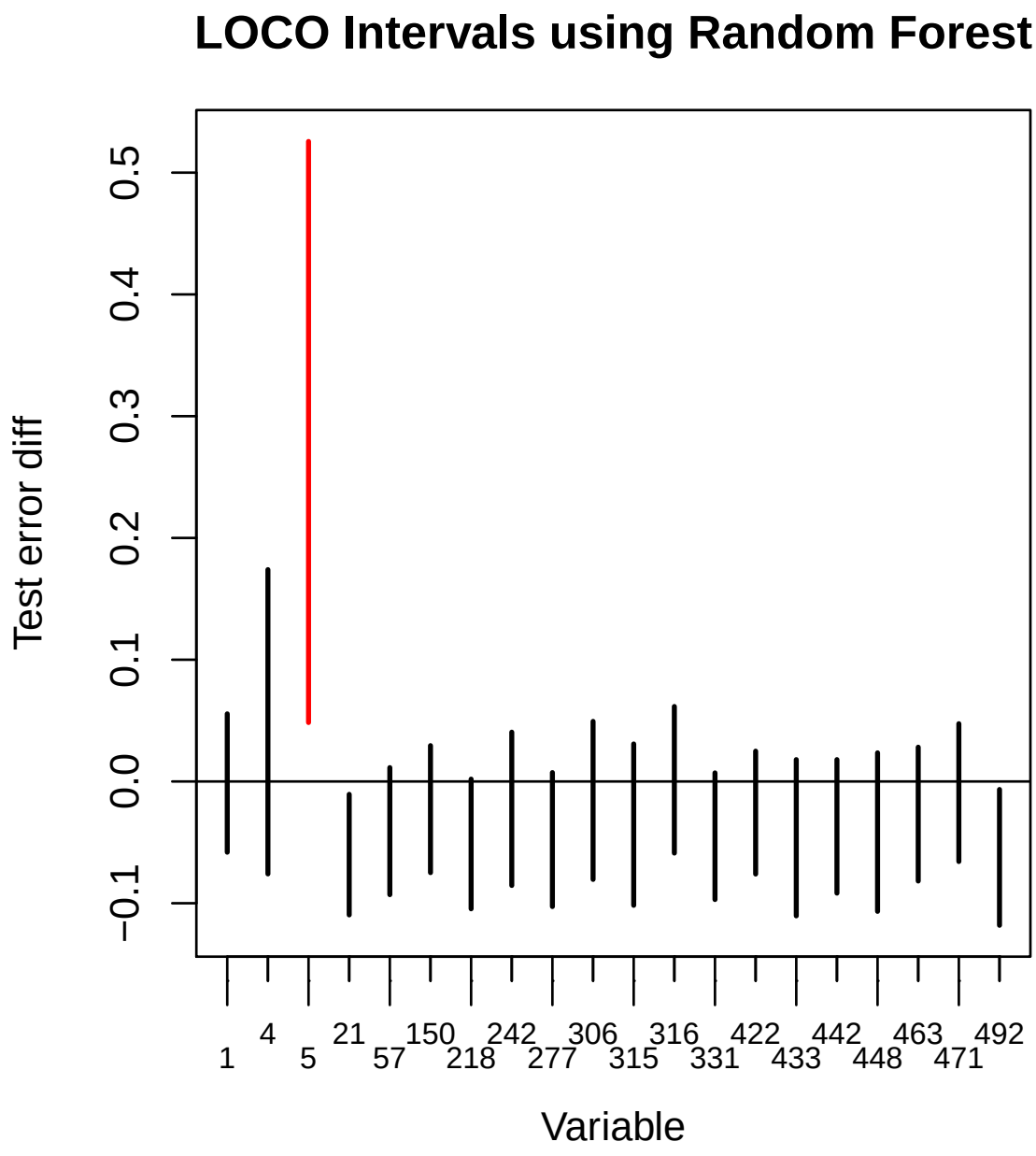
Examples of LOCO



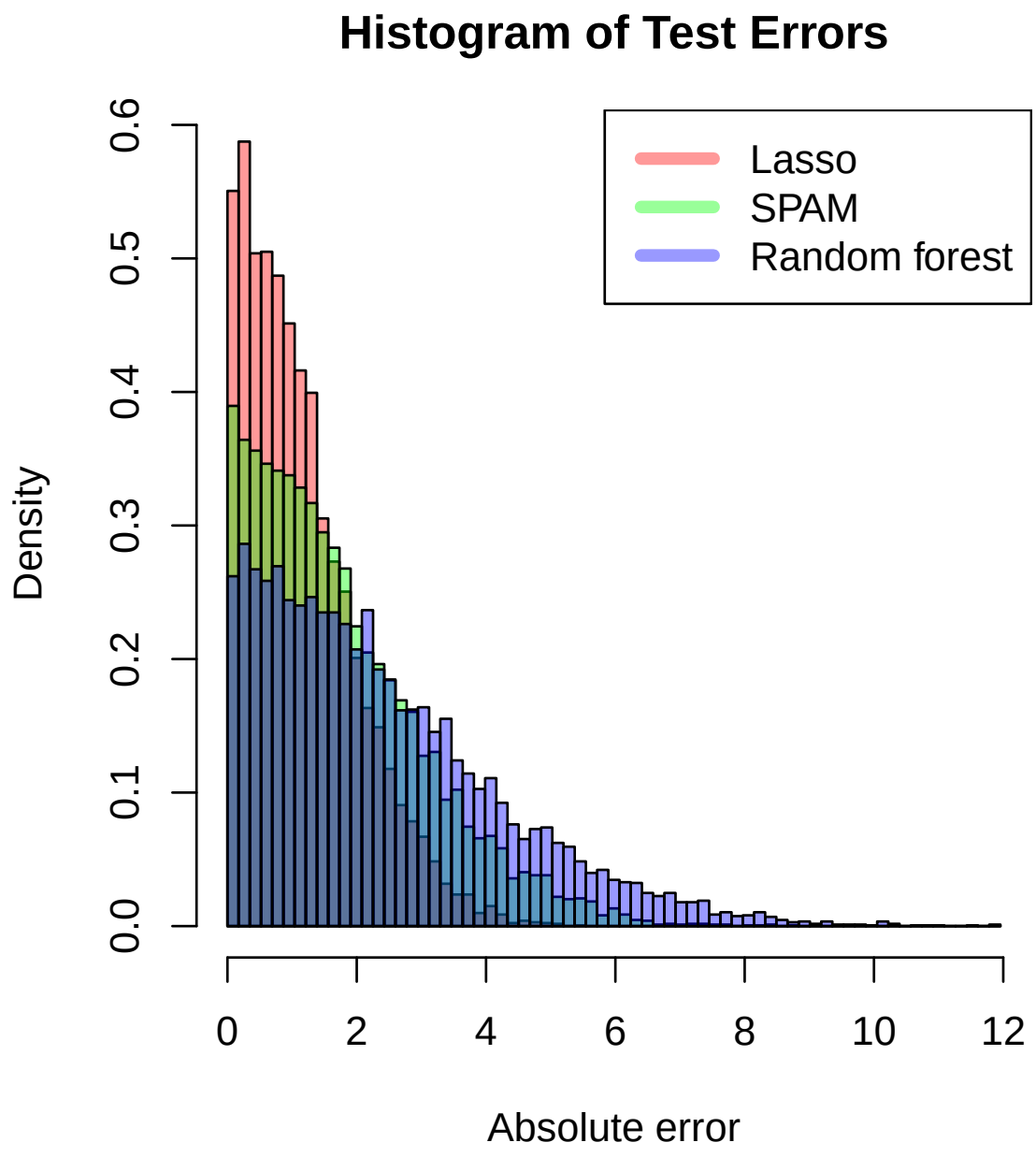
Examples of LOCO



Examples of LOCO



Examples of LOCO



The good (RT)

- Algorithmically flexible: any algorithm can be used to measure variable importance

The good (RT)

- Algorithmically flexible: any algorithm can be used to measure variable importance
- Computationally cheap(-ish): one refitting of the algorithm at hand per variable considered

The good (RT)

- Algorithmically flexible: any algorithm can be used to measure variable importance
- Computationally cheap(-ish): one refitting of the algorithm at hand per variable considered
- No distributional assumptions: intervals for $\theta_j(D_1)$ have exact coverage in finite-sample, for any distribution P of (X, Y)

The good (RT)

- Algorithmically flexible: any algorithm can be used to measure variable importance
- Computationally cheap(-ish): one refitting of the algorithm at hand per variable considered
- No distributional assumptions: intervals for $\theta_j(D_1)$ have exact coverage in finite-sample, for any distribution P of (X, Y)
- Selective validity: intervals cover the selected variables

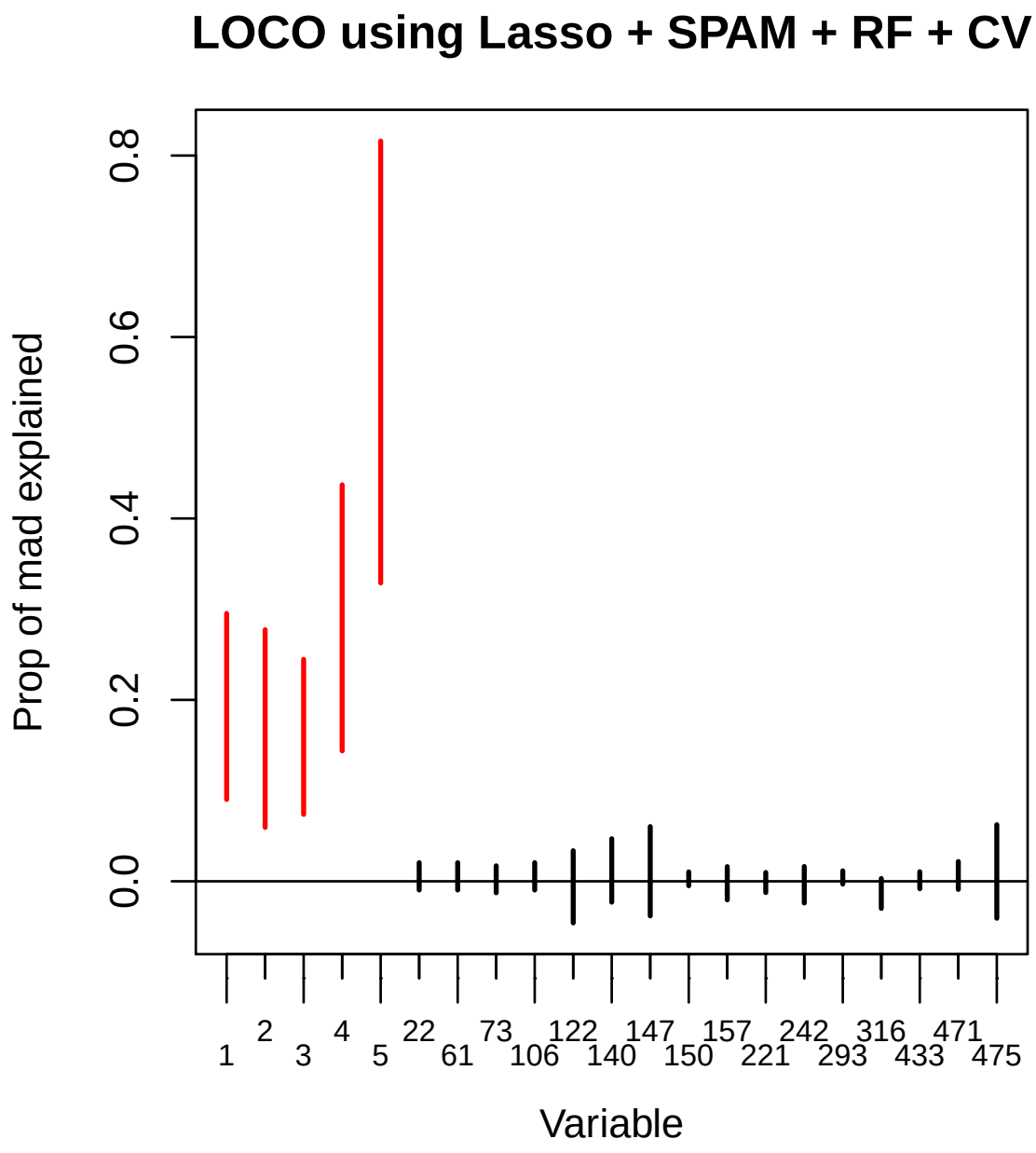
The good (RT)

- Algorithmically flexible: any algorithm can be used to measure variable importance
- Computationally cheap(-ish): one refitting of the algorithm at hand per variable considered
- No distributional assumptions: intervals for $\theta_j(D_1)$ have exact coverage in finite-sample, for any distribution P of (X, Y)
- Selective validity: intervals cover the selected variables
- Accuracy: Intervals (with Bonferroni correction, for s variables) have length $O(\sqrt{\log(sn)/n})$

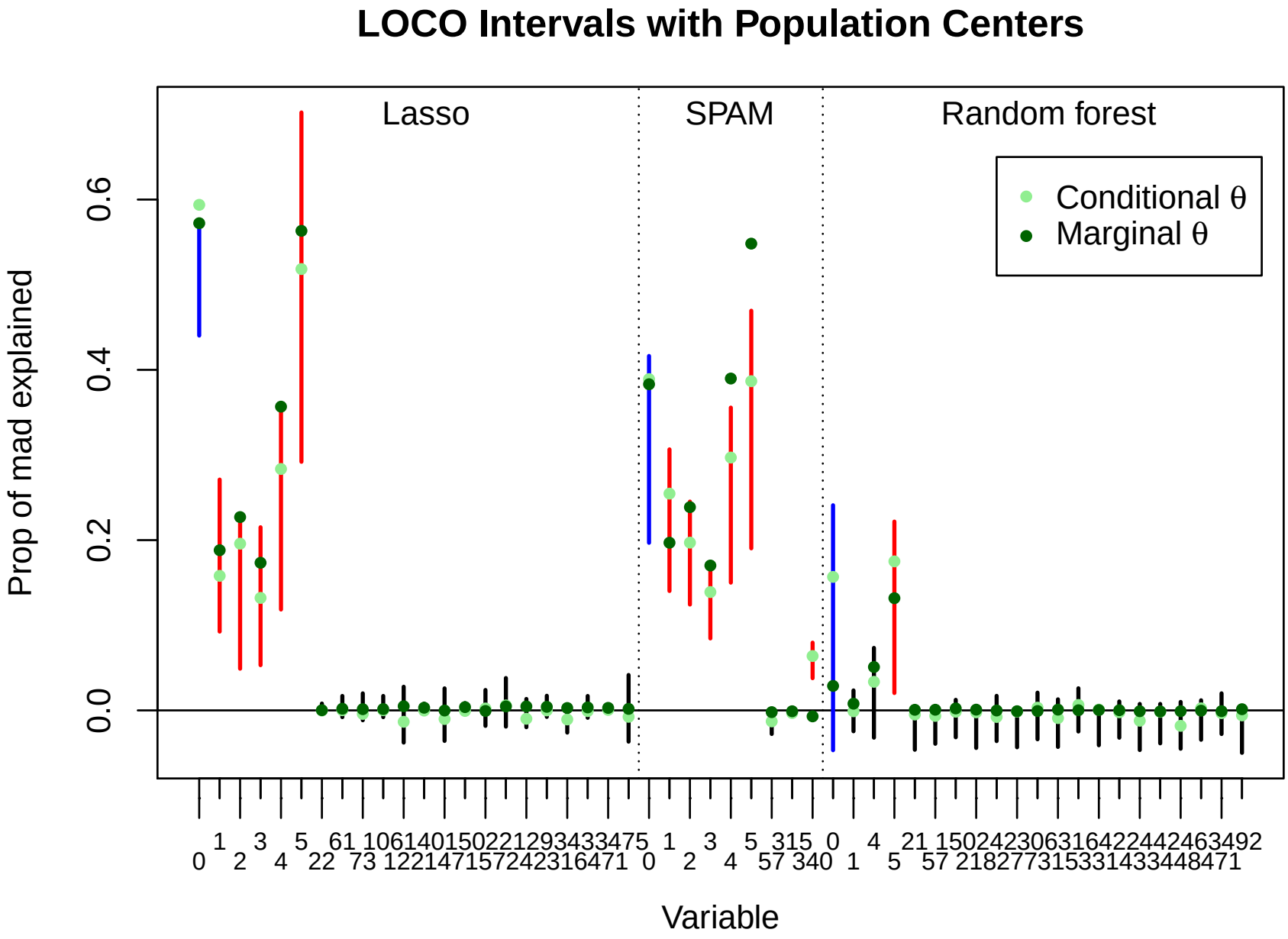
The good (RT)

- Algorithmically flexible: any algorithm can be used to measure variable importance
- Computationally cheap(-ish): one refitting of the algorithm at hand per variable considered
- No distributional assumptions: intervals for $\theta_j(D_1)$ have exact coverage in finite-sample, for any distribution P of (X, Y)
- Selective validity: intervals cover the selected variables
- Accuracy: Intervals (with Bonferroni correction, for s variables) have length $O(\sqrt{\log(sn)/n})$
- Simplicity: very simple/portable. Easy implementation

Combining many estimators (RT)



The Bias (RT)


LOCO Intervals with Population Centers
Lasso
SPAM
Random forest
Conditional θ
Marginal θ
Prop of mad explained
0.0
0.2
0.4
0.6
Variable

The Purely Predictive Approach: Conformal Prediction

The Purely Predictive Approach: Conformal Prediction

This is an approach which makes NO model assumptions.

The Purely Predictive Approach: Conformal Prediction

This is an approach which makes NO model assumptions.

It can be used from many tasks (not just regression).

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Connected to minimax density estimation, regression and classification: (Lei, Robins, Wasserman 2013), (Lei, Wasserman 2013) and classification (Lei 2014).

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Connected to minimax density estimation, regression and classification: (Lei, Robins, Wasserman 2013), (Lei, Wasserman 2013) and classification (Lei 2014).

High-dimensional regression, lasso (Lei, G'Sell, Rinaldo, Tibshirani, Wasserman 2018).

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Connected to minimax density estimation, regression and classification: (Lei, Robins, Wasserman 2013), (Lei, Wasserman 2013) and classification (Lei 2014).

High-dimensional regression, lasso (Lei, G'Sell, Rinaldo, Tibshirani, Wasserman 2018).

Multiclass problems. (Sadinle, Lei and Wasserman 2018).

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Connected to minimax density estimation, regression and classification: (Lei, Robins, Wasserman 2013), (Lei, Wasserman 2013) and classification (Lei 2014).

High-dimensional regression, lasso (Lei, G'Sell, Rinaldo, Tibshirani, Wasserman 2018).

Multiclass problems. (Sadinle, Lei and Wasserman 2018).

Random effects: Dunn and Wasserman (2018).

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Connected to minimax density estimation, regression and classification: (Lei, Robins, Wasserman 2013), (Lei, Wasserman 2013) and classification (Lei 2014).

High-dimensional regression, lasso (Lei, G'Sell, Rinaldo, Tibshirani, Wasserman 2018).

Multiclass problems. (Sadinle, Lei and Wasserman 2018).

Random effects: Dunn and Wasserman (2018).

Deep learning: Hechtlinger, Poczos, Wasserman (2018), Hechtlinger, Dalmasso, Rinaldo and Wasserman (2018).

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Connected to minimax density estimation, regression and classification: (Lei, Robins, Wasserman 2013), (Lei, Wasserman 2013) and classification (Lei 2014).

High-dimensional regression, lasso (Lei, G'Sell, Rinaldo, Tibshirani, Wasserman 2018).

Multiclass problems. (Sadinle, Lei and Wasserman 2018).

Random effects: Dunn and Wasserman (2018).

Deep learning: Hechtlinger, Poczos, Wasserman (2018), Hechtlinger, Dalmasso, Rinaldo and Wasserman (2018).

Clustering: Lei, Rinaldo and Wasserman (2016), Shin, Rinaldo and Wasserman (in progress).

Recent History

Conformal prediction (Vovk, Gammerman, Shafer 2005). Only assumes exchangeability.

Connected to minimax density estimation, regression and classification: (Lei, Robins, Wasserman 2013), (Lei, Wasserman 2013) and classification (Lei 2014).

High-dimensional regression, lasso (Lei, G'Sell, Rinaldo, Tibshirani, Wasserman 2018).

Multiclass problems. (Sadinle, Lei and Wasserman 2018).

Random effects: Dunn and Wasserman (2018).

Deep learning: Hechtlinger, Poczos, Wasserman (2018), Hechtlinger, Dalmasso, Rinaldo and Wasserman (2018).

Clustering: Lei, Rinaldo and Wasserman (2016), Shin, Rinaldo and Wasserman (in progress).

Robustness: Balakrishnan, Patil, Shrotriya and Wasserman (in progress).

Conformal Prediction

Augment $\longrightarrow$ Fit $\longrightarrow$ Test $\longrightarrow$ Invert

Conformal Prediction

Augment $\longrightarrow$ Fit $\longrightarrow$ Test $\longrightarrow$ Invert

Augment data: $\mathcal{A} = \{(X_1, Y_1), \ldots, (X_n, Y_n), (x, y)\}$

Conformal Prediction

Augment $\longrightarrow$ Fit $\longrightarrow$ Test $\longrightarrow$ Invert

Augment data: $\mathcal{A} = \{(X_1, Y_1), \ldots, (X_n, Y_n), (x, y)\}$

Fit and get 'residual' or 'conformal score' (this is the art)

$$R_i = \phi((X_i, Y_i), \mathcal{A})$$

Conformal Prediction

Augment $\longrightarrow$ Fit $\longrightarrow$ Test $\longrightarrow$ Invert

Augment data: $\mathcal{A} = \{(X_1, Y_1), \ldots, (X_n, Y_n), (x, y)\}$

Fit and get 'residual' or 'conformal score' (this is the art)

$$R_i = \phi((X_i, Y_i), \mathcal{A})$$

Test: $H_0 : (X_{n+1}, Y_{n+1}) = (x, y)$. Get p-value

$$\pi(x) = \frac{1}{n+1} \sum_{i=1}^{n+1} I(R_i \geq R_{n+1})$$

Conformal Prediction

Augment $\longrightarrow$ Fit $\longrightarrow$ Test $\longrightarrow$ Invert

Augment data: $\mathcal{A} = \{(X_1, Y_1), \ldots, (X_n, Y_n), (x, y)\}$

Fit and get 'residual' or 'conformal score' (this is the art)

$$R_i = \phi((X_i, Y_i), \mathcal{A})$$

Test: $H_0 : (X_{n+1}, Y_{n+1}) = (x, y)$. Get p-value

$$\pi(x) = \frac{1}{n+1} \sum_{i=1}^{n+1} I(R_i \geq R_{n+1})$$

Invert

$$C_n(x) = \{y : \ \pi(x) \geq \alpha\}$$

Conformal Prediction

Conformal Prediction

We have

$$1 - \alpha \leq \inf_{P} P(Y_{n+1} \in C_n(X_{n+1})) \leq 1 - \alpha + \frac{1}{n+1}.$$

Conformal Prediction

We have

$$1 - \alpha \leq \inf_{P} P(Y_{n+1} \in C_n(X_{n+1})) \leq 1 - \alpha + \frac{1}{n+1}.$$

No assumptions!

Conformal Prediction

We have

$$1 - \alpha \leq \inf_{P} P(Y_{n+1} \in C_n(X_{n+1})) \leq 1 - \alpha + \frac{1}{n+1}.$$

No assumptions!

Does not require the model $\widehat{m}$ to be correct.

Split Conformal

Split the data into two groups $\mathcal{D}_1$ and $\mathcal{D}_2$.

Split Conformal

Split the data into two groups $\mathcal{D}_1$ and $\mathcal{D}_2$.

Compute residuals of the form

$$R_i = \phi(Y_i, Q)$$

where Q is computed from $\mathcal{D}_1$ and $Y_i \in \mathcal{D}_2$.

Split Conformal

Split the data into two groups $\mathcal{D}_1$ and $\mathcal{D}_2$.

Compute residuals of the form

$$R_i = \phi(Y_i, Q)$$

where Q is computed from $\mathcal{D}_1$ and $Y_i \in \mathcal{D}_2$.

Let q be the $1 - \alpha$ quantile of the R_i's and let $C = \{y : \ \phi(y, Q) \leq q\}$.

Split Conformal

Split the data into two groups $\mathcal{D}_1$ and $\mathcal{D}_2$.

Compute residuals of the form

$$R_i = \phi(Y_i, Q)$$

where Q is computed from $\mathcal{D}_1$ and $Y_i \in \mathcal{D}_2$.

Let q be the $1-\alpha$ quantile of the R_i's and let $C = \{y : \ \phi(y, Q) \leq q\}$.

Then

$$1 - \alpha \leq \inf_P P(Y_{n+1} \in C_n(X_{n+1})) \leq 1 - \alpha + \frac{2}{n+2}.$$

This is much faster (no augmentation step) but leads to wider intervals.

Split Conformal

We could split N times at level $1 - \alpha/N$ and set

$$C^N = \bigcup_j C_j$$

but:

Split Conformal

We could split N times at level $1 - \alpha/N$ and set

$$C^N = \bigcup_j C_j$$

but:

under weak conditions, with probability tending to 1,

$$\mu(C^N) > \mu(C_{\text{split}}).$$

The Choice of Conformal Score

The Choice of Conformal Score

Examples:

$$|Y_i - \widehat{m}_{\text{aug}}(X_i)|$$

$$\frac{1}{\widehat{p}_{\text{aug}}(X_i, Y_i)}$$

$$\frac{1}{\widehat{p}_{\text{aug}}(Y_i|X_i)}$$

$$\frac{1}{\widehat{p}_{\text{aug}}(X_i|Y_i)}$$

$$\frac{1}{\widehat{p}_{\text{aug}}(X_i, Y_i; \widehat{\theta})}$$

Efficiency

Efficiency

The choice affects efficiency (size of set) not validity

Efficiency

The choice affects efficiency (size of set) not validity

Can be tuned to get minimax prediction sets (see Lei, Robins and Wasserman 2013; Lei and Wasserman 2014).

Efficiency

The choice affects efficiency (size of set) not validity

Can be tuned to get minimax prediction sets (see Lei, Robins and Wasserman 2013; Lei and Wasserman 2014).

Example: Unsupervised prediction.

Efficiency

The choice affects efficiency (size of set) not validity

Can be tuned to get minimax prediction sets (see Lei, Robins and Wasserman 2013; Lei and Wasserman 2014).

Example: Unsupervised prediction.

Set

$$R_i(y) = \frac{1}{\widehat{p}_h(Y_i; \{Y_1, \ldots, Y_n, y\})}$$

Unsupervised Prediction: Preservation of Minimaxity

The resulting set C_n satisfies $\inf_P P(Y_{n+1} \in C_n) \geq 1 - \alpha$ and

Unsupervised Prediction: Preservation of Minimaxity

The resulting set C_n satisfies $\inf_P P(Y_{n+1} \in C_n) \geq 1 - \alpha$ and

(A1) Suppose p is β-Holder smooth.
(A2) Suppose

$$c_1|\epsilon|^\gamma \leq |P(\{y : \ p(y) \leq t_\alpha + \epsilon\}) - \alpha| \leq c_2|\epsilon|^\gamma.$$

Unsupervised Prediction: Preservation of Minimaxity

The resulting set C_n satisfies $\inf_P P(Y_{n+1} \in C_n) \geq 1 - \alpha$ and

(A1) Suppose p is β-Holder smooth.
(A2) Suppose

$$c_1|\epsilon|^\gamma \leq |P(\{y : \ p(y) \leq t_\alpha + \epsilon\}) - \alpha| \leq c_2|\epsilon|^\gamma.$$

Then, for all $\lambda > 0$,

$$\mathbb{P}\left(\mu(C_n \Delta C_\alpha) \succeq \left(\frac{\log n}{n}\right)^{\beta\gamma/(2\beta+d)} + \left(\frac{\log n}{n}\right)^{1/2}\right) \leq \left(\frac{1}{n}\right)^\lambda$$

where C_α is the oracle (smallest possible) prediction set.

Unsupervised Prediction: Preservation of Minimaxity

The resulting set C_n satisfies $\inf_P P(Y_{n+1} \in C_n) \geq 1 - \alpha$ and

(A1) Suppose p is β-Holder smooth.
(A2) Suppose

$$c_1|\epsilon|^\gamma \leq |P(\{y : \ p(y) \leq t_\alpha + \epsilon\}) - \alpha| \leq c_2|\epsilon|^\gamma.$$

Then, for all $\lambda > 0$,

$$\mathbb{P}\left(\mu(C_n \Delta C_\alpha) \succeq \left(\frac{\log n}{n}\right)^{\beta\gamma/(2\beta+d)} + \left(\frac{\log n}{n}\right)^{1/2}\right) \leq \left(\frac{1}{n}\right)^\lambda$$

where C_α is the oracle (smallest possible) prediction set.
But if (A1), (A2) fail, we still have coverage (even if P does not have a density).

Unsupervised Prediction: Preservation of Minimaxity

The resulting set C_n satisfies $\inf_P P(Y_{n+1} \in C_n) \geq 1 - \alpha$ and

(A1) Suppose p is β-Holder smooth.
(A2) Suppose

$$c_1|\epsilon|^\gamma \leq |P(\{y : \ p(y) \leq t_\alpha + \epsilon\}) - \alpha| \leq c_2|\epsilon|^\gamma.$$

Then, for all $\lambda > 0$,

$$\mathbb{P}\left(\mu(C_n \Delta C_\alpha) \succeq \left(\frac{\log n}{n}\right)^{\beta\gamma/(2\beta+d)} + \left(\frac{\log n}{n}\right)^{1/2}\right) \leq \left(\frac{1}{n}\right)^\lambda$$

where C_α is the oracle (smallest possible) prediction set.
But if (A1), (A2) fail, we still have coverage (even if P does not have a density).
Can also use size of C_n to choose the bandwidth.

Properties (Regression Case)

Oracle:
$C(x) = [\widehat{\mu}(x) - q_n, \widehat{\mu}(x) + q_n]$
where q_n is the upper quantile from $\mathrm{Law}(|Y - \widehat{\mu}|)$.

Properties (Regression Case)

Oracle:
$C(x) = [\widehat{\mu}(x) - q_n, \widehat{\mu}(x) + q_n]$
where q_n is the upper quantile from $\mathrm{Law}(|Y - \widehat{\mu}|)$.

Super-Oracle:
$C(x) = [\mu(x) - q, \mu(x) + q]$
where q is the upper quantile from $\mathrm{Law}(|Y - \mu|)$.

Properties (Regression Case)

Then

$$\text{Length(Conformal Interval)} - \text{Length(Oracle)} = O_P(\eta_n + \rho_n + n^{-1/2})$$

where

$$P(\sup_y ||\widehat{\mu} - \widehat{\mu}_{X,y}||_\infty > \eta_n) \leq \rho_n.$$

Properties (Regression Case)

Then

$$\text{Length(Conformal Interval)} - \text{Length(Oracle)} = O_P(\eta_n + \rho_n + n^{-1/2})$$

where

$$P(\sup_y ||\widehat{\mu} - \widehat{\mu}_{X,y}||_\infty > \eta_n) \leq \rho_n.$$

With some additional assumptions:

$$\mu(\text{Oracle} \Delta \text{Conformal}) = o_P(1)$$

and

$$P(Y \in C(X) | X = x) \to 1 - \alpha.$$

Properties (Regression Case)

Then

$$\text{Length(Conformal Interval)} - \text{Length(Oracle)} = O_P(\eta_n + \rho_n + n^{-1/2})$$

where

$$P(\sup_y ||\widehat{\mu} - \widehat{\mu}_{X,y}||_\infty > \eta_n) \leq \rho_n.$$

With some additional assumptions:

$$\mu(\text{Oracle} \Delta \text{Conformal}) = o_P(1)$$

and

$$P(Y \in C(X) | X = x) \to 1 - \alpha.$$

Under the 'usual' assumptions (linearity, incoherhence etc)

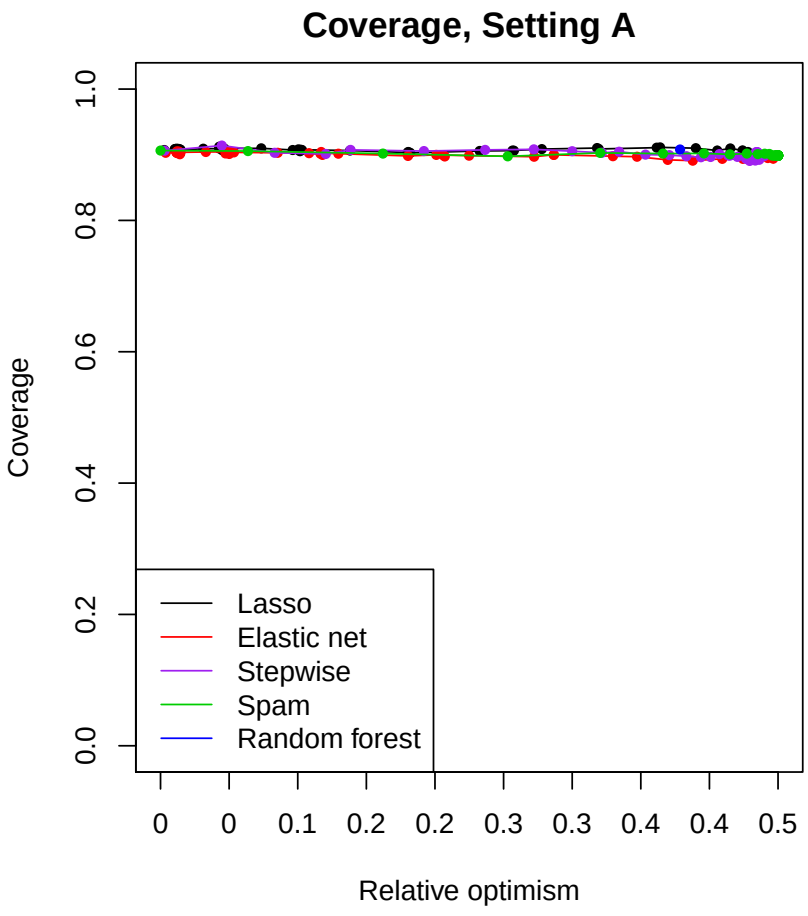
$$\eta_n = \frac{\kappa^2 s \sqrt{\log d}}{\sqrt{n}}$$

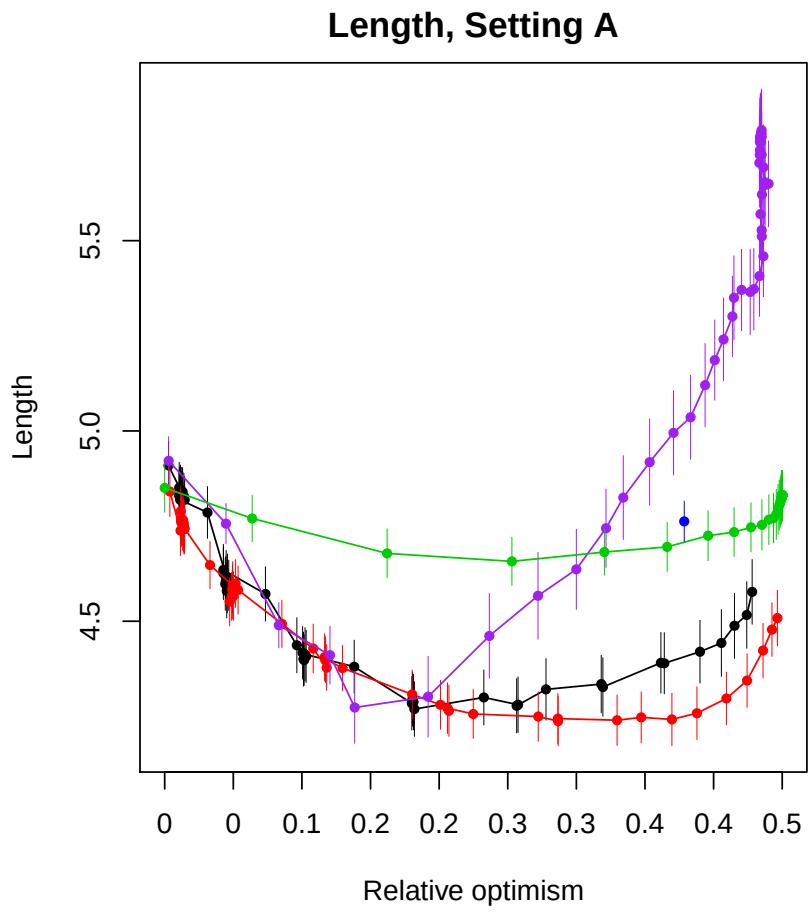
κ from restricted isometry condition, s = sparsity, $\rho_n = (1/d)^c$.

High Dimensional Examples

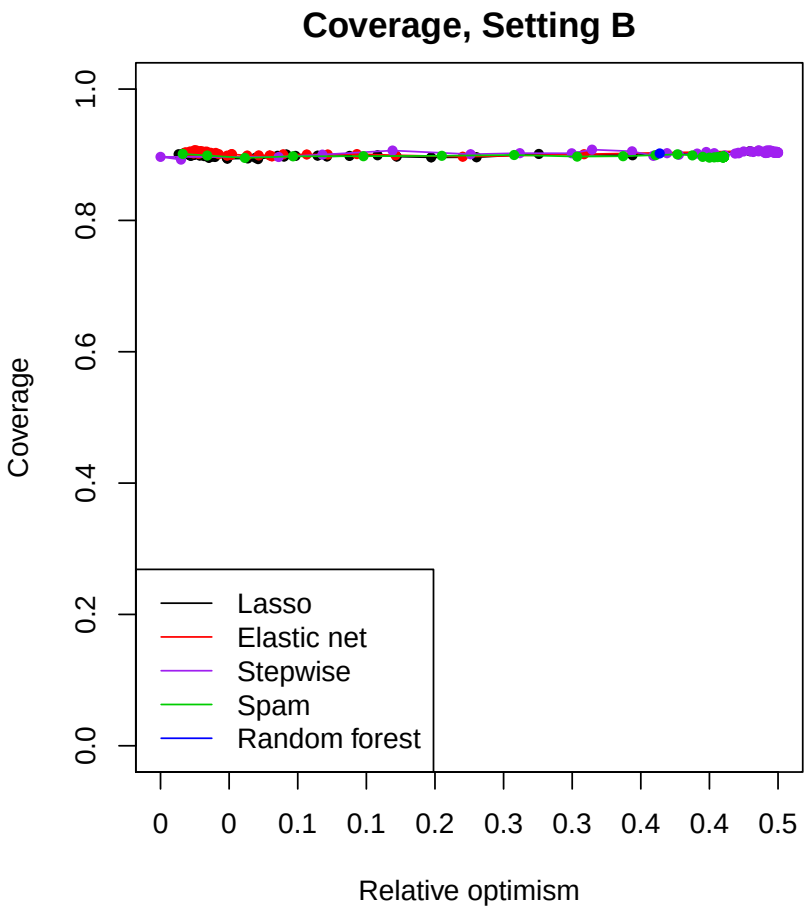
Amazing accuracy in very high dimensions

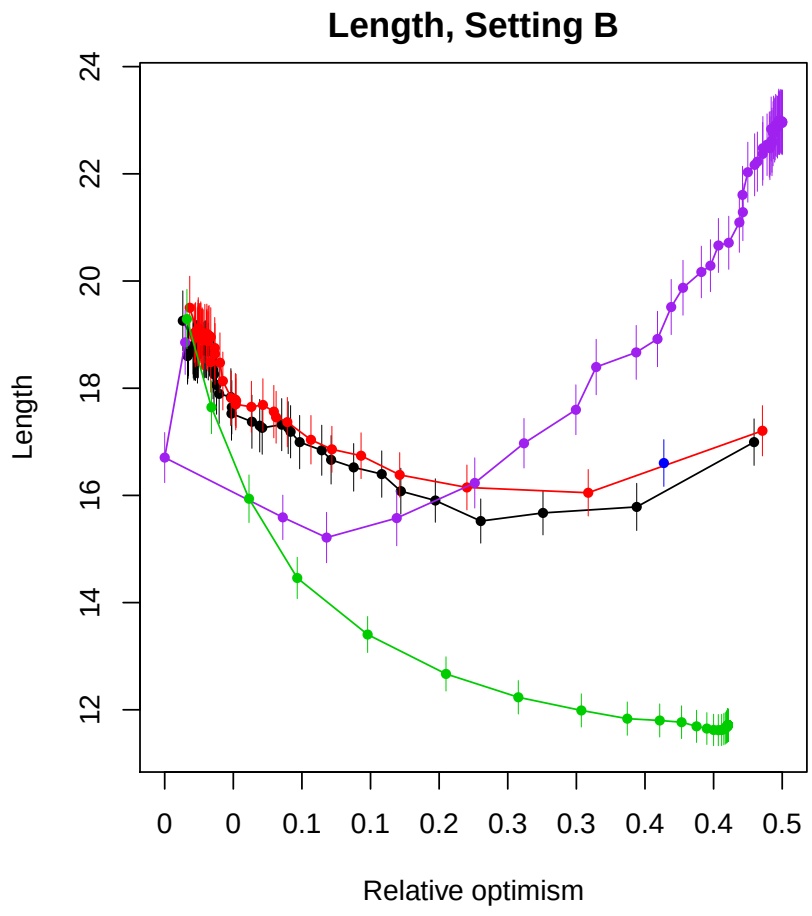
Example: $n = 200, d = 2,000$; linear and Normal



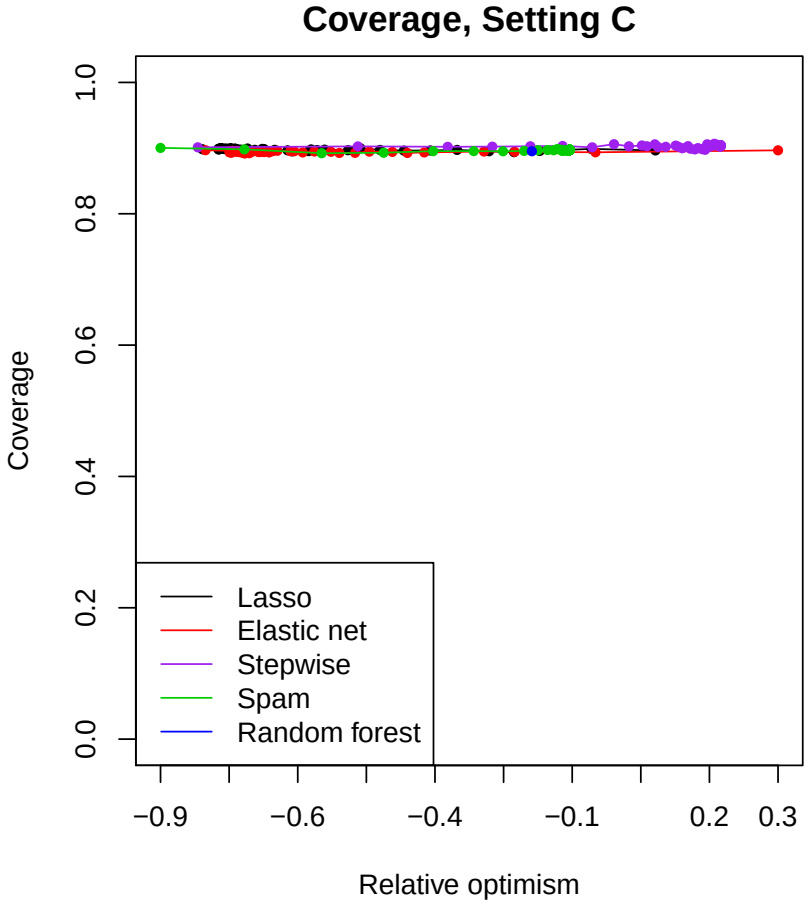


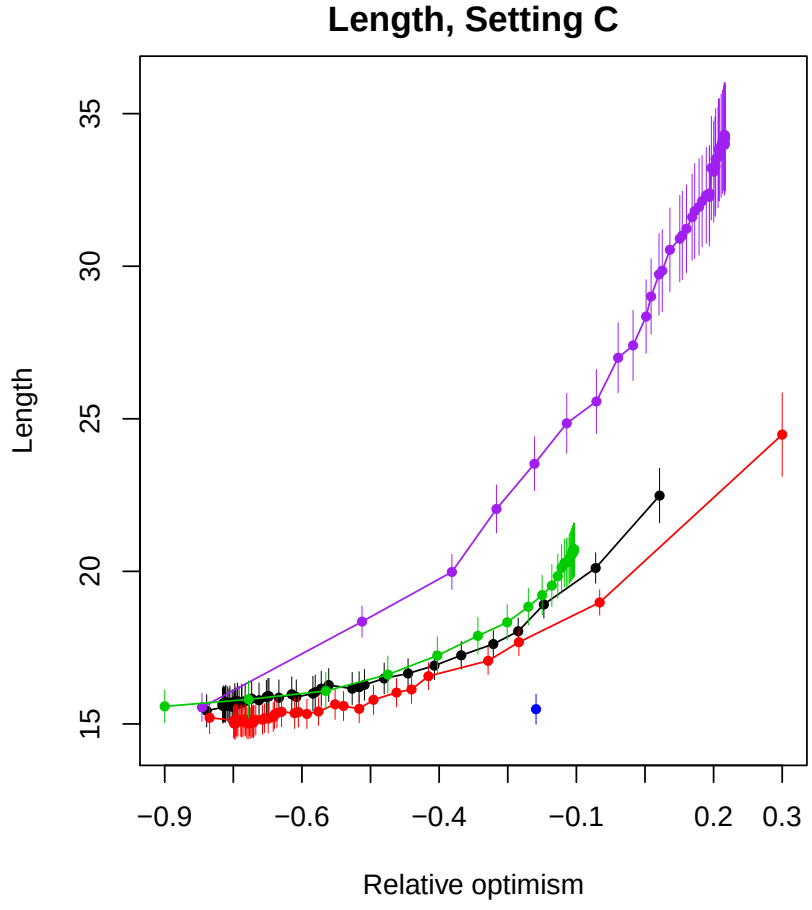
Example: $n = 200, d = 2,000$; nonlinear and heavy-tailed





Example: $n = 200, d = 2,000$; linear, correlated, heteroskedastic, heavy-tailed





Sliced Regression

[1] joint work with Yotam Hechtlinger, Nic Dalmasso, Alessandro Rinaldo

Sliced Regression

Careful choice of conformal score matters.[1]

[1] joint work with Yotam Hechtlinger, Nic Dalmasso, Alessandro Rinaldo

Sliced Regression

Careful choice of conformal score matters.[1]

Use residuals based on slicing i.e. $\widehat{p}(x|dy)$.

[1] joint work with Yotam Hechtlinger, Nic Dalmasso, Alessandro Rinaldo

Sliced Regression

Careful choice of conformal score matters.[1]

Use residuals based on slicing i.e. $\widehat{p}(x|dy)$.

Similar to sliced inverse regression but we do NOT assume the usual index model $Y = f(\beta^T X) + \epsilon$.

[1] joint work with Yotam Hechtlinger, Nic Dalmasso, Alessandro Rinaldo

Sliced Regression

Careful choice of conformal score matters.[1]

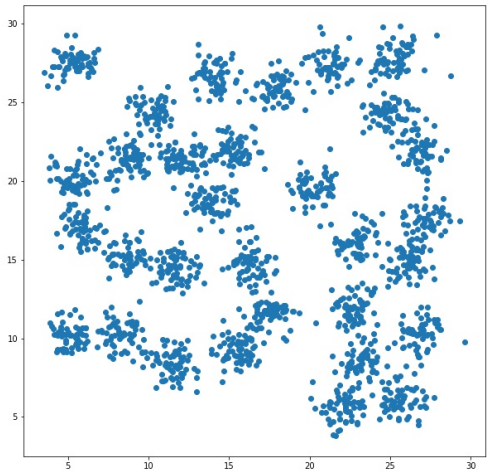
Use residuals based on slicing i.e. $\widehat{p}(x|dy)$.

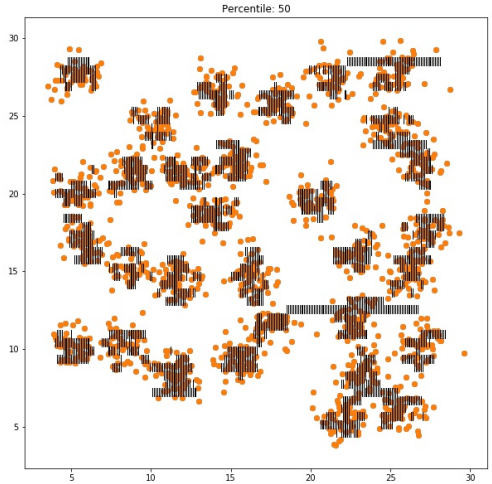
Similar to sliced inverse regression but we do NOT assume the usual index model $Y = f(\beta^T X) + \epsilon$.

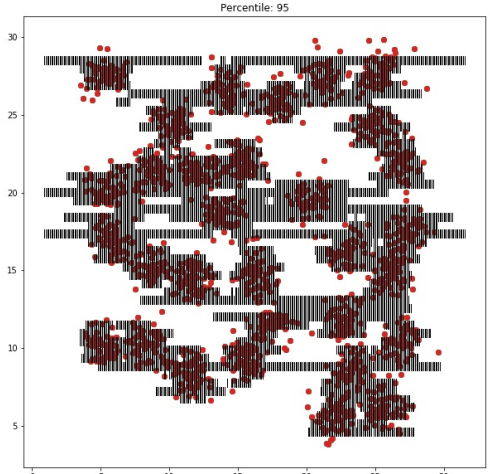
This choice encourages outputting $C_n = \emptyset$ when x is unusual.

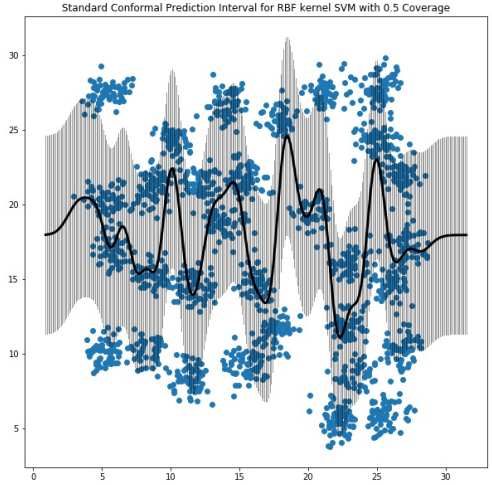
[1] joint work with Yotam Hechtlinger, Nic Dalmasso, Alessandro Rinaldo

The D31 Data




Percentile: 50


Percentile: 95


Standard Conformal Prediction Interval for RBF kernel SVM with 0.5 Coverage

Choice of Features: Efficiency not Validity

Part of choosing score is the choice of covariates.

Choice of Features: Efficiency not Validity

Part of choosing score is the choice of covariates.

The Merck molecular activity dataset.

Choice of Features: Efficiency not Validity

Part of choosing score is the choice of covariates.

The Merck molecular activity dataset.

n = 14,875 observations
d = 5,464 features.

Choice of Features: Efficiency not Validity

Part of choosing score is the choice of covariates.

The Merck molecular activity dataset.

n = 14,875 observations
d = 5,464 features.

Sliced conformal regression.

Choice of Features: Efficiency not Validity

Part of choosing score is the choice of covariates.

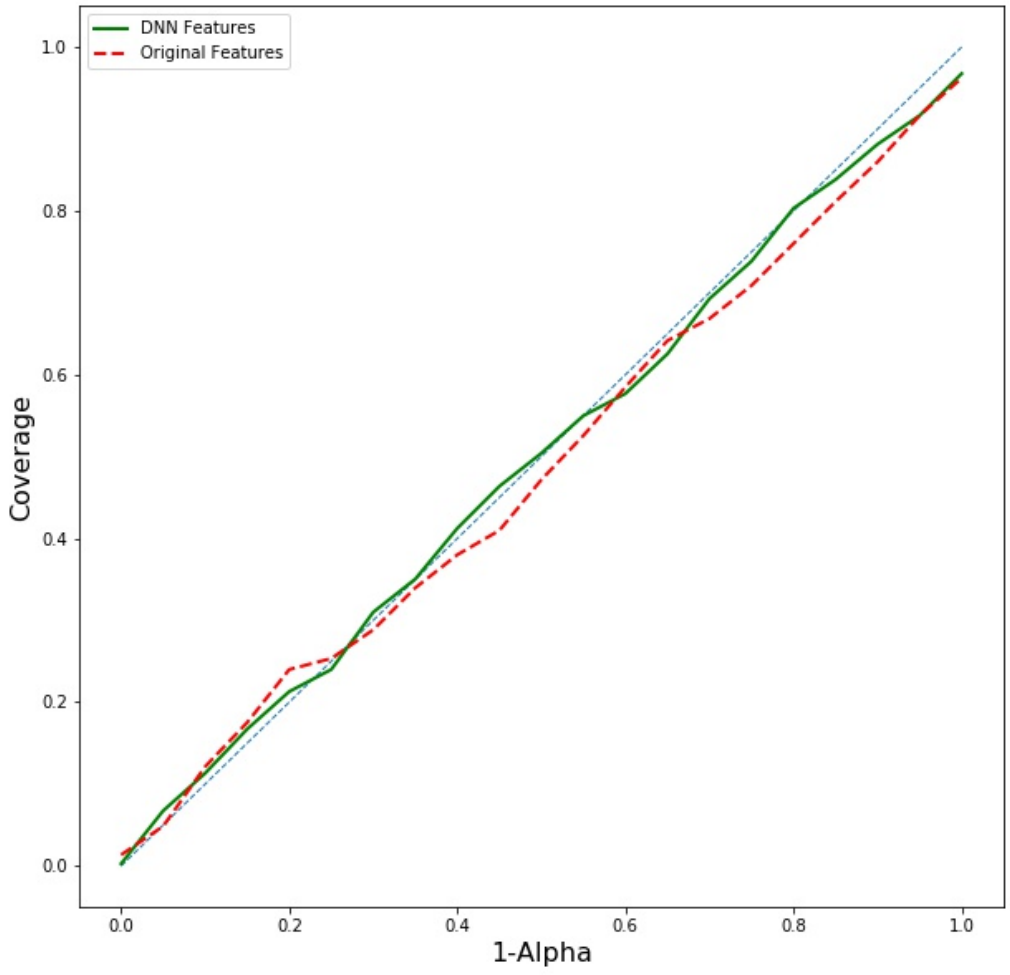
The Merck molecular activity dataset.

n = 14,875 observations
d = 5,464 features.

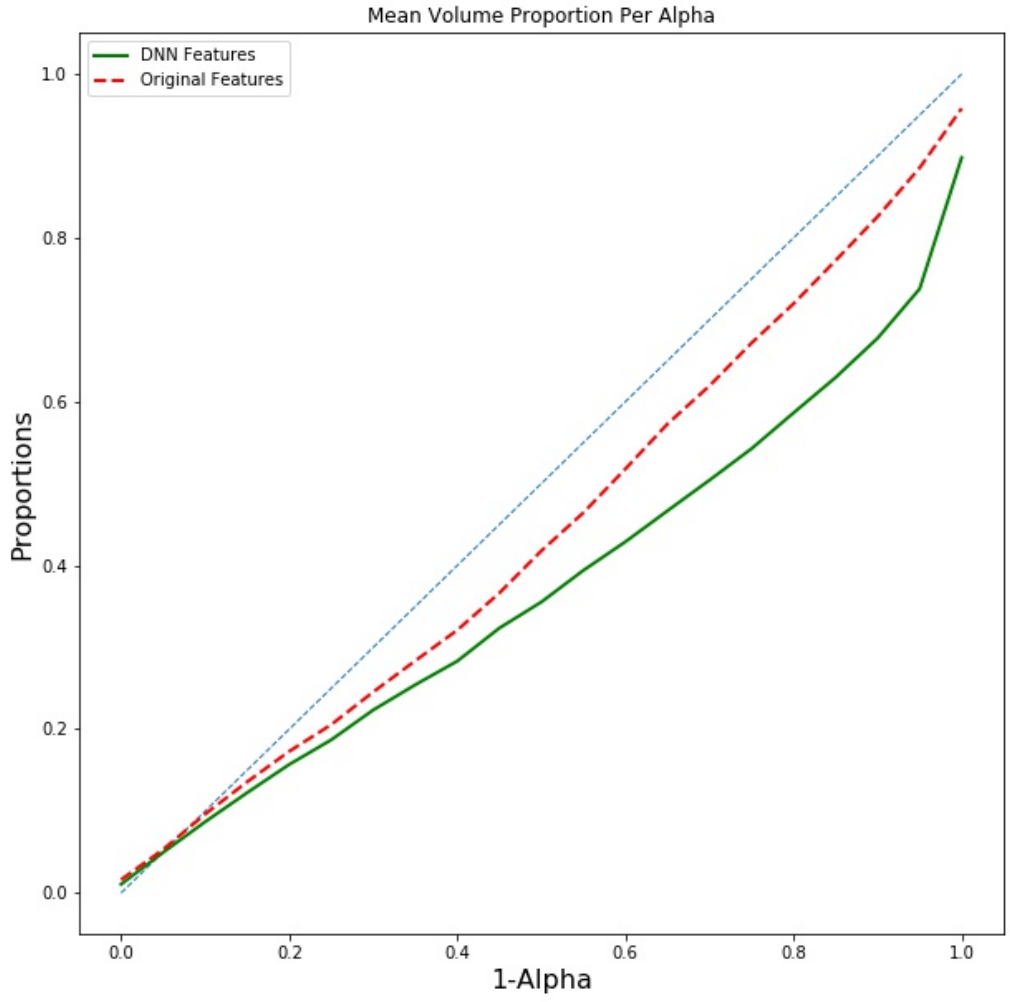
Sliced conformal regression.

Use features as given
or
extract features from deep learning?

Choice of Features: Efficiency not Validity



Choice of Features: Efficiency not Validity



Optimal Multiclass Prediction: (Sadinle, Lei and W. 2018)

Optimal Multiclass Prediction: (Sadinle, Lei and W. 2018)

Suppose that $Y \in \{1, \ldots, k\}$. k can be large.

Optimal Multiclass Prediction: (Sadinle, Lei and W. 2018)

Suppose that $Y \in \{1, \ldots, k\}$. k can be large.

Want to minimize $\mathbb{E}|C(X)|$ subject to
$P(Y \in C(X)|Y = y) \geq 1 - \alpha_y$.
Call $\mathbb{E}|C(X)|$ the ambiguity.

Optimal Multiclass Prediction: (Sadinle, Lei and W. 2018)

Suppose that $Y \in \{1, \ldots, k\}$. k can be large.

Want to minimize $\mathbb{E}|C(X)|$ subject to
$P(Y \in C(X)|Y = y) \geq 1 - \alpha_y$.
Call $\mathbb{E}|C(X)|$ the ambiguity.

Solution (NP lemma) is

$$C^*(x) = \{x : \ p(y|x) \geq t_y\}$$

Estimating the Optimal Classifier

Estimating the Optimal Classifier

Plug-in estimator $\widehat{C} = \{x : \ p(y|x) \geq \widehat{t}_y\}$:

Estimating the Optimal Classifier

Plug-in estimator $\widehat{C} = \{x : \ p(y|x) \geq \widehat{t}_y\}$:

With prob at least $1 - k\delta_n - (1/n)$,

$$P(\widehat{C} \Delta C^*) \preceq \epsilon_n^\gamma + \sqrt{\frac{\log n}{n}}$$

where δ_n and ϵ_n are model specific.

Estimating the Optimal Classifier

Plug-in estimator $\widehat{C} = \{x : \ p(y|x) \geq \widehat{t}_y\}$:

With prob at least $1 - k\delta_n - (1/n)$,

$$P(\widehat{C} \Delta C^*) \preceq \epsilon_n^\gamma + \sqrt{\frac{\log n}{n}}$$

where δ_n and ϵ_n are model specific.

Example: kNN, $p(y|x)$ Lipschitz: $\delta_n = 1/n$ and $\epsilon_n \asymp (\log n/n)^{1/(2d)}$.

Estimating the Optimal Classifier

Plug-in estimator $\widehat{C} = \{x : \ p(y|x) \geq \widehat{t}_y\}$:

With prob at least $1 - k\delta_n - (1/n)$,

$$P(\widehat{C} \Delta C^*) \preceq \epsilon_n^\gamma + \sqrt{\frac{\log n}{n}}$$

where δ_n and ϵ_n are model specific.

Example: kNN, $p(y|x)$ Lipschitz: $\delta_n = 1/n$ and $\epsilon_n \asymp (\log n/n)^{1/(2d)}$.

Example: sparse logistic regression, with incoherence assumption:

$$\epsilon \asymp (\log d/n)^{1/4}, \qquad \delta_n \asymp \frac{1}{d} + ||\beta||_0 \sqrt{\log d/n}$$

Estimating the Optimal Classifier

Plug-in estimator $\widehat{C} = \{x : \ p(y|x) \geq \widehat{t}_y\}$:

With prob at least $1 - k\delta_n - (1/n)$,

$$P(\widehat{C} \Delta C^*) \preceq \epsilon_n^\gamma + \sqrt{\frac{\log n}{n}}$$

where δ_n and ϵ_n are model specific.

Example: kNN, $p(y|x)$ Lipschitz: $\delta_n = 1/n$ and $\epsilon_n \asymp (\log n/n)^{1/(2d)}$.

Example: sparse logistic regression, with incoherence assumption:

$$\epsilon \asymp (\log d/n)^{1/4}, \qquad \delta_n \asymp \frac{1}{d} + ||\beta||_0 \sqrt{\log d/n}$$

but validity does not depend on these assumptions.

(a) “3” and “5”.

(b) “4” and “9”.

(c) “7”, “8” and “9”.

(d) “0”, “2” and “3”.

(e) “1” and “4”.

(f) “3” and “5”.

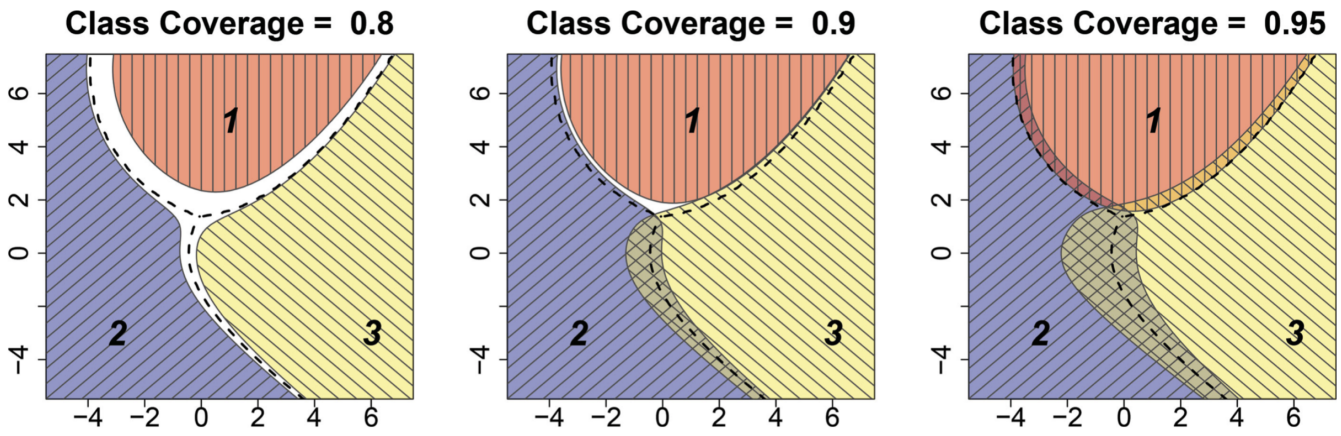
Price of Optimality

Price of Optimality

Sometimes, $C(x) = \emptyset$.

Price of Optimality

Sometimes, $C(x) = \emptyset$.



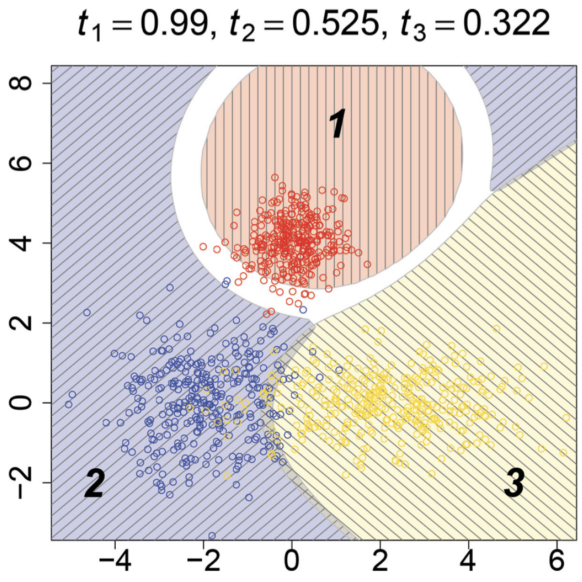
Null Predictions

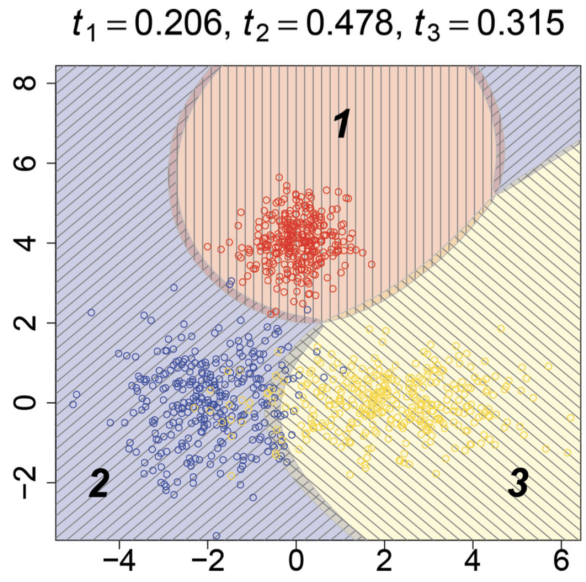
Could try to minimize $\mathbb{E}|C(X)|$ subject to
$P(Y \in C(X)|Y = y) \geq 1 - \alpha_y$ and subject to $C(x) \neq \emptyset$.
Difficult in general.

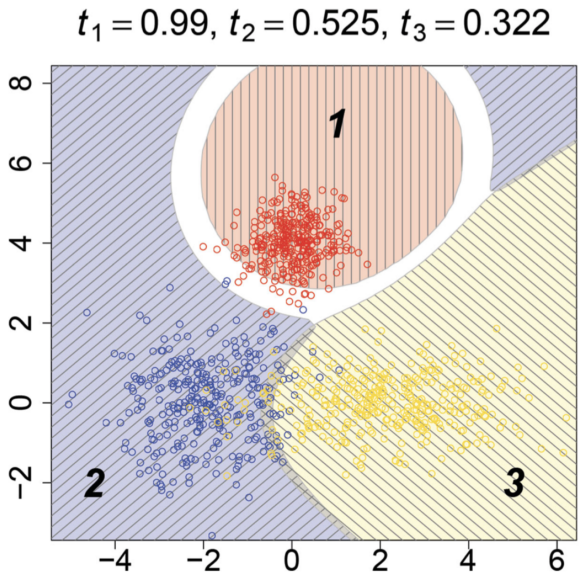
Null Predictions

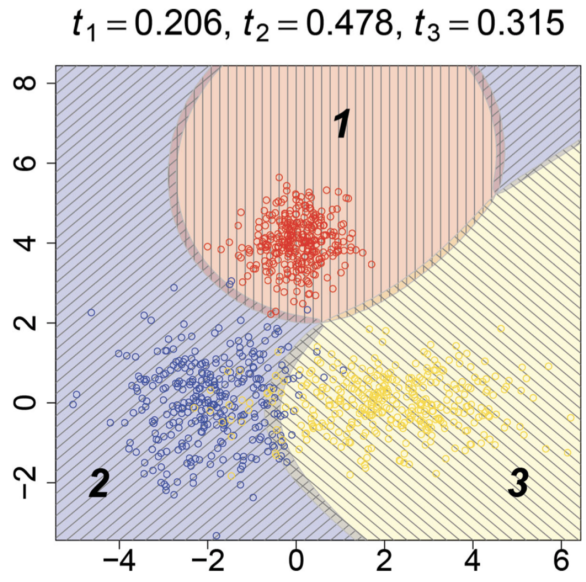
Could try to minimize $\mathbb{E}|C(X)|$ subject to
$P(Y \in C(X)|Y = y) \geq 1 - \alpha_y$ and subject to $C(x) \neq \emptyset$.
Difficult in general.

Accretive completion: gradually decrease each t_y (greedily)while minimizing ambiguity $\mathbb{E}|C(x)|$.


$t_1 = 0.99,\ t_2 = 0.525,\ t_3 = 0.322$
1
2
3
8
6
4
2
0
−2
−4
−2
0
2
4
6


$t_1 = 0.206,\ t_2 = 0.478,\ t_3 = 0.315$
1
2
3
8
6
4
2
0
−2
−4
−2
0
2
4
6





... but maybe we should encourage empty set predictions ....

Cautious Deep Learning

If k is large, and X is ambiguous, $C(X)$ will be large. This is a feature.

Cautious Deep Learning

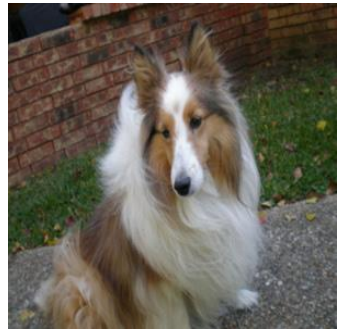
If k is large, and X is ambiguous, $C(X)$ will be large. This is a feature.

If we use $\widehat{p}(x|y)$ as a score, and x is unusual (i.e. $\widehat{p}(x|y)$ small for all y) we will encourage $C(x) = \emptyset$. This is a feature not a bug!

Deep Learning

TL: Sea Snake
Prediction: Null Set

TL: Alp
Prediction: Ski



TL: Shetland Sheepdog
Prediction: Shetland Sheepdog, Collie, Toilet Paper

TL: Soup Bowl
Prediction: Face Powder, Soup Bowl, Tray

TL: Cradle
Prediction: Sleeping Bag

TL: Garter Snake
Prediction: Null Set

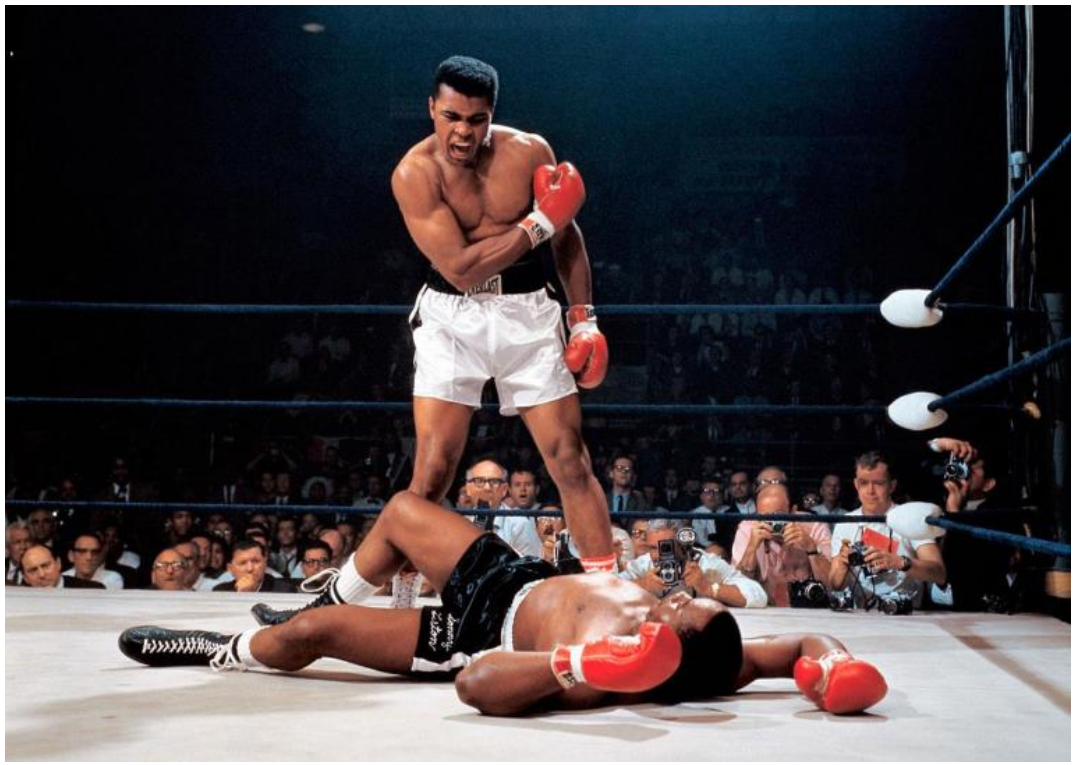
TL: Porcupine
Prediction: Porcupine, Quill

TL: Bakery
Prediction: Null Set

Deep Learning

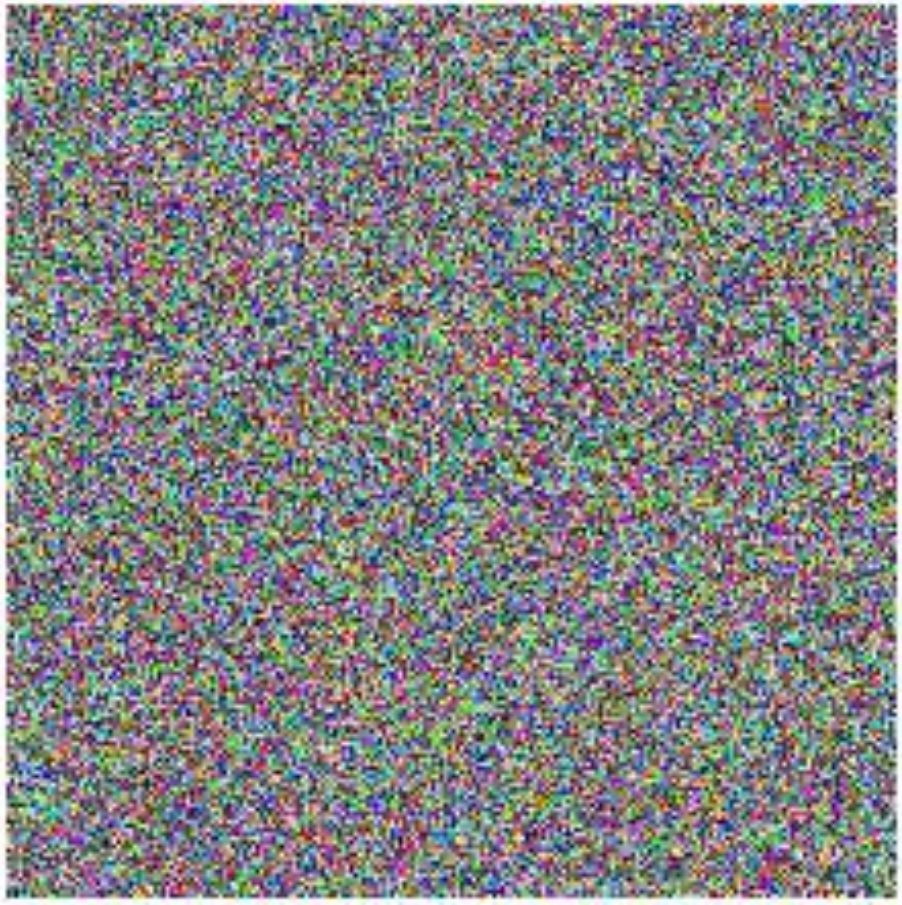
Our Method (α=.5):	Inception-v4 Model:
• Null Set	• Coil (0.910) • Hay (0.008) • Maze (0.005)

Deep Learning



Our Method (α=.55):	Inception-v4 Model:
• Null Set	• Volleyball (0.388) • Tennis Ball (0.160) • Racket (0.157)

Deep Learning



Our Method (α=.5):	Inception-v4 Model:
• Null Set	• Kite (0.137) • Bee Eater (0.033) • Missle (0.031)

Back To Inference: Combining LOCO With Conformal Prediction

Back To Inference: Combining LOCO With Conformal Prediction

Use conformal set C to assess effects of dropping covariates.

Back To Inference: Combining LOCO With Conformal Prediction

Use conformal set C to assess effects of dropping covariates.

LOCO: Drop $X(j)$ to get $\widehat{m}_{(-j)}$ and let

$$\Delta_j(x, y) = |y - \widehat{m}_{(-j)}(x)| - |y - \widehat{m}(x)|$$

Back To Inference: Combining LOCO With Conformal Prediction

Use conformal set C to assess effects of dropping covariates.

LOCO: Drop $X(j)$ to get $\widehat{m}_{(-j)}$ and let

$$\Delta_j(x, y) = |y - \widehat{m}_{(-j)}(x)| - |y - \widehat{m}(x)|$$

$$W_j(x) = \{\Delta_j(x, y) : \ y \in C\}.$$

Back To Inference: Combining LOCO With Conformal Prediction

Use conformal set C to assess effects of dropping covariates.

LOCO: Drop $X(j)$ to get $\widehat{m}_{(-j)}$ and let

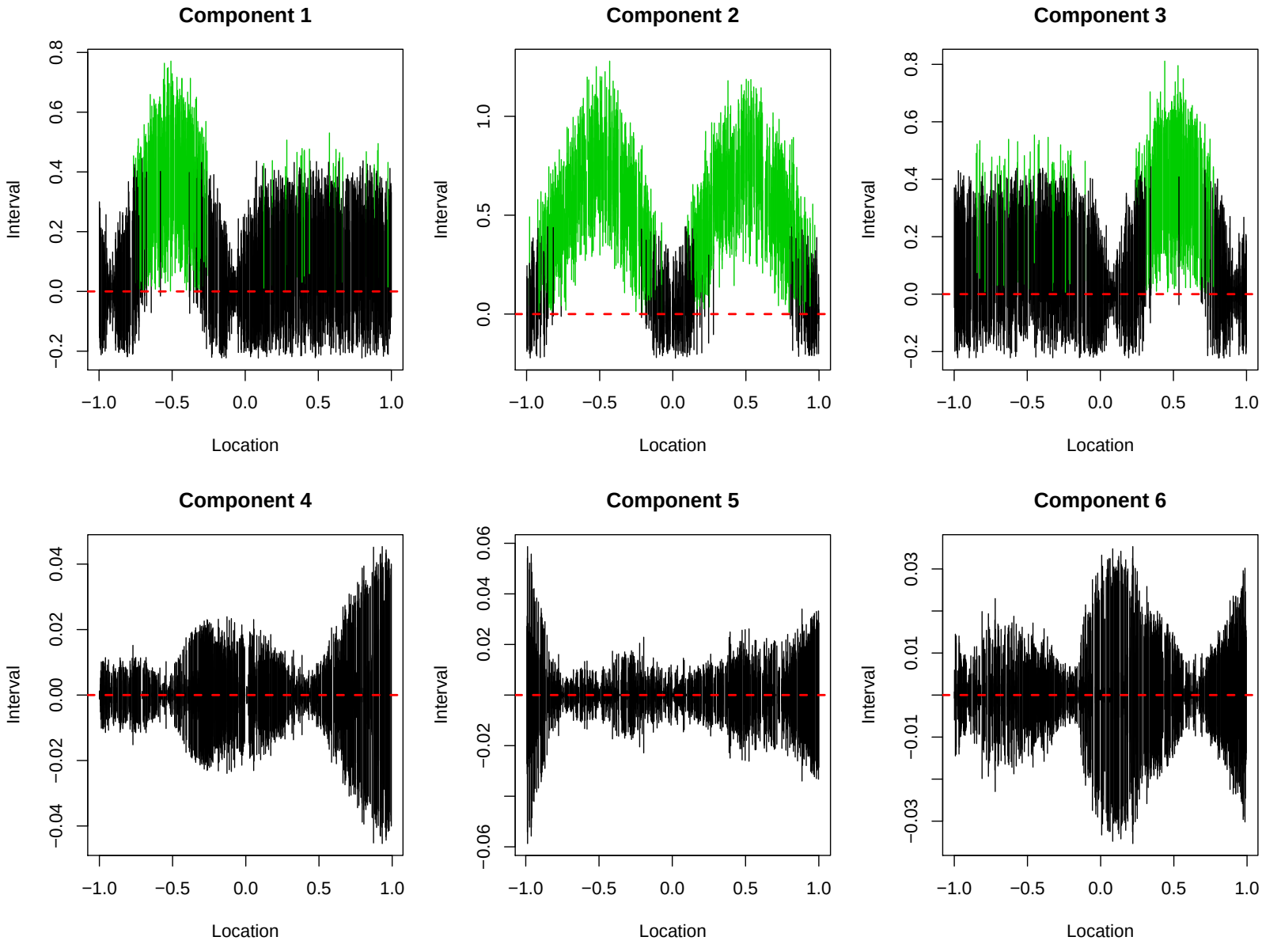
$$\Delta_j(x, y) = |y - \widehat{m}_{(-j)}(x)| - |y - \widehat{m}(x)|$$

$$W_j(x) = \{\Delta_j(x, y) : \ y \in C\}.$$

Then

$$\inf_P P(\Delta_j(X, Y) \in C(X) \text{ for all } j) \geq 1 - \alpha.$$

Nonparametric Additive Model: $f_4 = f_5 = f_6 = 0$



Part II: Clustering

By taking a predictive view of clustering we solve several problems:

Part II: Clustering

By taking a predictive view of clustering we solve several problems:

- choose tuning parameters

Part II: Clustering

By taking a predictive view of clustering we solve several problems:

- choose tuning parameters
- automatically merge some clusters

Part II: Clustering

By taking a predictive view of clustering we solve several problems:

- choose tuning parameters
- automatically merge some clusters
- replace Voronoi diagram with spheres and ellipsoids (more natural)

Part II: Clustering

By taking a predictive view of clustering we solve several problems:

- choose tuning parameters
- automatically merge some clusters
- replace Voronoi diagram with spheres and ellipsoids (more natural)
- get a prediction coverage gaurantee

Predictive Clustering

Basic idea:

Predictive Clustering

Basic idea:

1. perform clustering.

Predictive Clustering

Basic idea:

1. perform clustering.
2. define cluster-based conformal score (residual)

Predictive Clustering

Basic idea:

1. perform clustering.
2. define cluster-based conformal score (residual)
3. Get conformal set.

Predictive Clustering

Basic idea:

1. perform clustering.
2. define cluster-based conformal score (residual)
3. Get conformal set.
4. Choose tuning parameters to minimize size of conformal prediction set.

Predictive Clustering

Basic idea:

1. perform clustering.
2. define cluster-based conformal score (residual)
3. Get conformal set.
4. Choose tuning parameters to minimize size of conformal prediction set.

This implicitly creates a bias-variance tradeoff that is missing in clustering.

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Run k-means on $\mathcal{Y}_1$ to get centers $c_1, \ldots, c_k$.

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Run k-means on $\mathcal{Y}_1$ to get centers $c_1, \ldots, c_k$.
3. For the data in $\mathcal{Y}_2$ compute the (non-augmented) residuals

$$R_i = \min_j ||Y_i - c_{j(i)}||$$

where $c_{j(i)}$ is the closest center to Y_i.

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Run k-means on $\mathcal{Y}_1$ to get centers $c_1, \ldots, c_k$.
3. For the data in $\mathcal{Y}_2$ compute the (non-augmented) residuals

$$R_i = \min_j ||Y_i - c_{j(i)}||$$

   where $c_{j(i)}$ is the closest center to Y_i.
4. Let t_α be the $1 - \alpha$ quantile of the residuals.

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Run k-means on $\mathcal{Y}_1$ to get centers $c_1, \ldots, c_k$.
3. For the data in $\mathcal{Y}_2$ compute the (non-augmented) residuals

$$R_i = \min_j ||Y_i - c_{j(i)}||$$

   where $c_{j(i)}$ is the closest center to Y_i.
4. Let t_α be the $1 - \alpha$ quantile of the residuals.
5. Let $\mathcal{C}_k = \bigcup_j B(c_j, t_\alpha)$.

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Run k-means on $\mathcal{Y}_1$ to get centers $c_1, \ldots, c_k$.
3. For the data in $\mathcal{Y}_2$ compute the (non-augmented) residuals

$$R_i = \min_j ||Y_i - c_{j(i)}||$$

   where $c_{j(i)}$ is the closest center to Y_i.
4. Let t_α be the $1 - \alpha$ quantile of the residuals.
5. Let $\mathcal{C}_k = \bigcup_j B(c_j, t_\alpha)$.
6. Choose $\widehat{k}$ to minimize Lebesgue measure. (Has a min!)

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Run k-means on $\mathcal{Y}_1$ to get centers $c_1, \ldots, c_k$.
3. For the data in $\mathcal{Y}_2$ compute the (non-augmented) residuals

$$R_i = \min_j ||Y_i - c_{j(i)}||$$

   where $c_{j(i)}$ is the closest center to Y_i.
4. Let t_α be the $1 - \alpha$ quantile of the residuals.
5. Let $\mathcal{C}_k = \bigcup_j B(c_j, t_\alpha)$.
6. Choose $\widehat{k}$ to minimize Lebesgue measure. (Has a min!)
7. Return: $\mathcal{C}_{\widehat{k}} = \bigcup_j B(c_j, t_\alpha)$

From k-means to k-spheres

1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Run k-means on $\mathcal{Y}_1$ to get centers $c_1, \ldots, c_k$.
3. For the data in $\mathcal{Y}_2$ compute the (non-augmented) residuals

$$R_i = \min_j ||Y_i - c_{j(i)}||$$

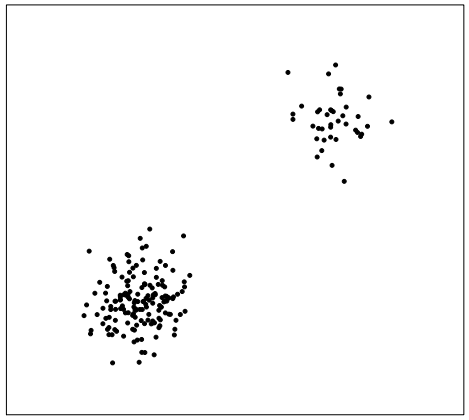
   where $c_{j(i)}$ is the closest center to Y_i.
4. Let t_α be the $1-\alpha$ quantile of the residuals.
5. Let $\mathcal{C}_k = \bigcup_j B(c_j, t_\alpha)$.
6. Choose $\widehat{k}$ to minimize Lebesgue measure. (Has a min!)
7. Return: $\mathcal{C}_{\widehat{k}} = \bigcup_j B(c_j, t_\alpha)$

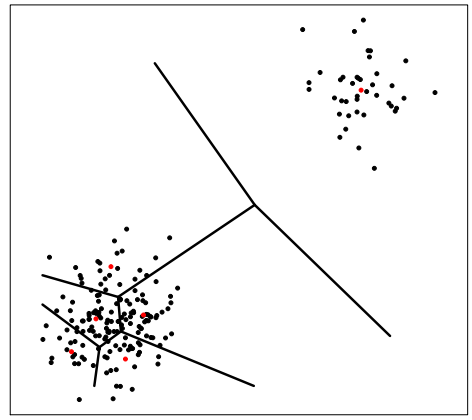
Then

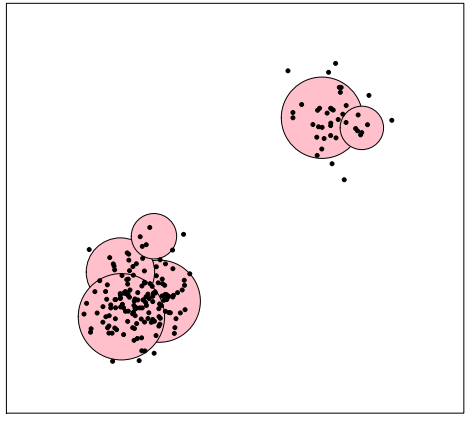
$$\inf_P P(Y \in \mathcal{C}) \geq 1 - \alpha$$

Clusters $C_1, \ldots, C_r$ are the connected components of $\mathcal{C}$.

Example







Top left: Data. Top right: $k = 6$-means. Bottom left: Repaired by our method.

Improved Residuals

We can choose any residuals for conformal prediction.

$$R_i = \min_j \frac{||Y_i - c_j||^2}{\widehat{\sigma}_j^2} + 2d \log \widehat{\sigma}_j - 2 \log \widehat{\pi}_j.$$

Then

$$\mathcal{C} = \bigcup_j B(c_j, r_j)$$

where

$$r_j = \widehat{\sigma}_j \sqrt{[t_\alpha + 2 \log \widehat{\pi}_j - 2d \log \widehat{\sigma}_j]_+}.$$

Improved k-means Prediction

Improved k-means Prediction

Smaller prediction sets.

Improved k-means Prediction

Smaller prediction sets.

Better approximation to density level sets.

Improved k-means Prediction

Smaller prediction sets.

Better approximation to density level sets.

Robust to outliers: Huber contamination model

$$P = (1 - \epsilon)P_0 + \epsilon Q$$

Improved k-means Prediction

Smaller prediction sets.

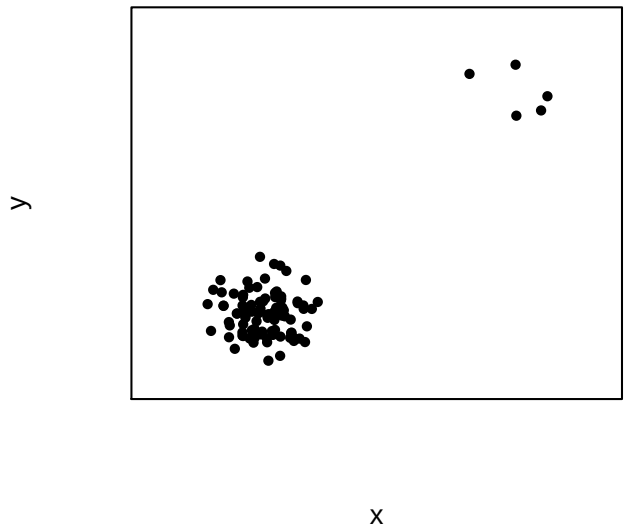
Better approximation to density level sets.

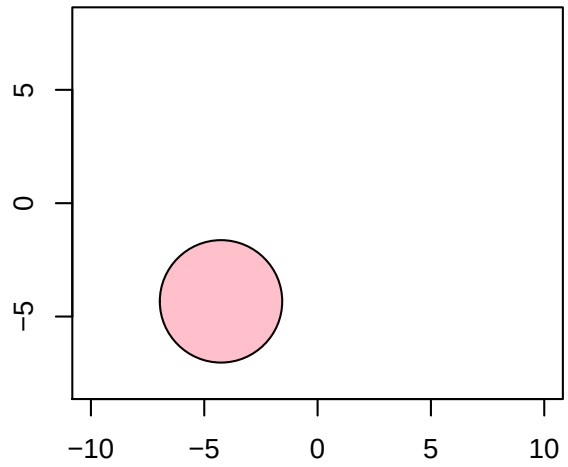
Robust to outliers: Huber contamination model

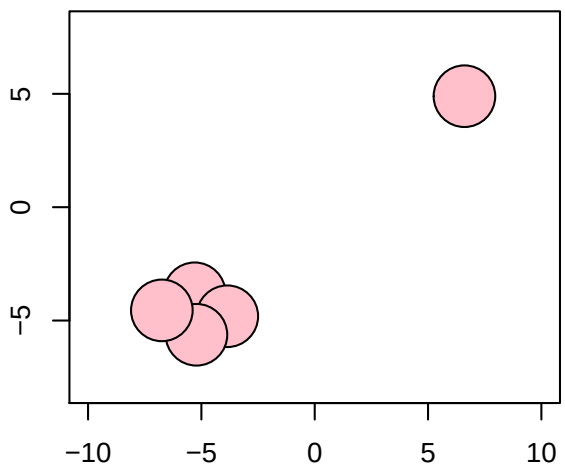
$$P = (1 - \epsilon)P_0 + \epsilon Q$$

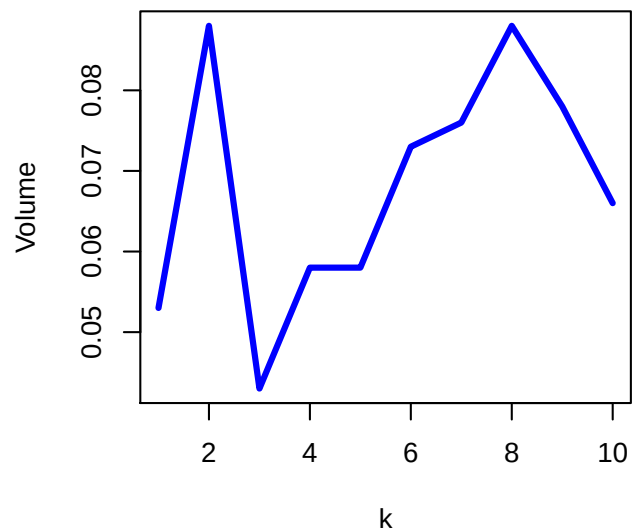
Valid coverage and smaller prediction set.

Example: Outliers, standard k-means

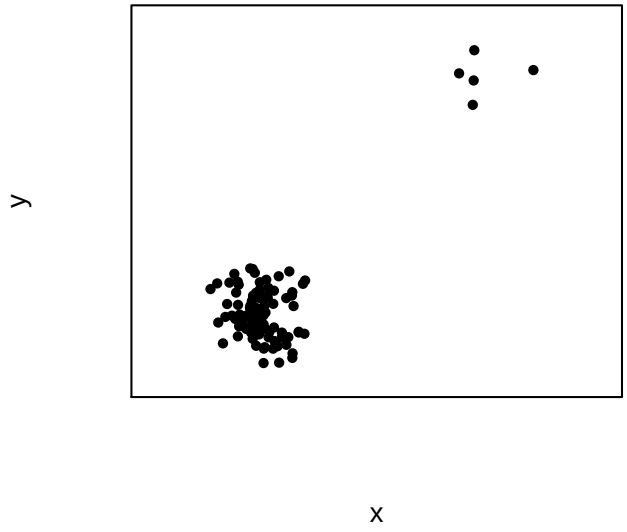


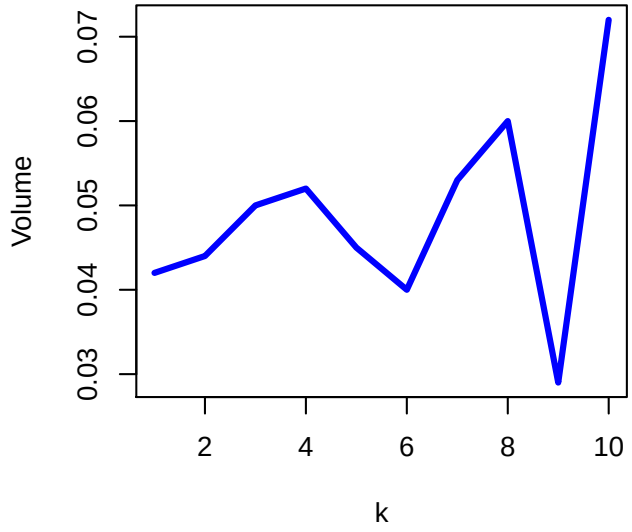


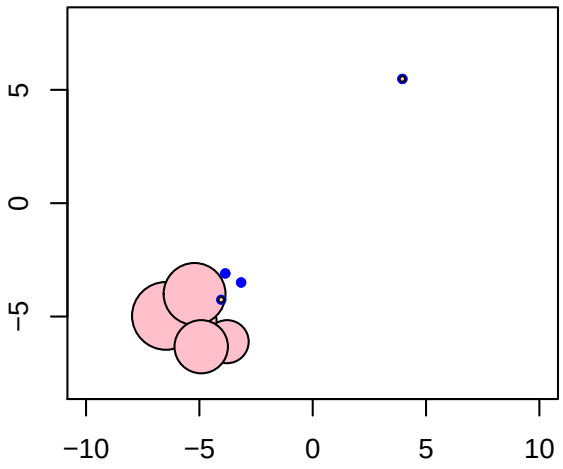




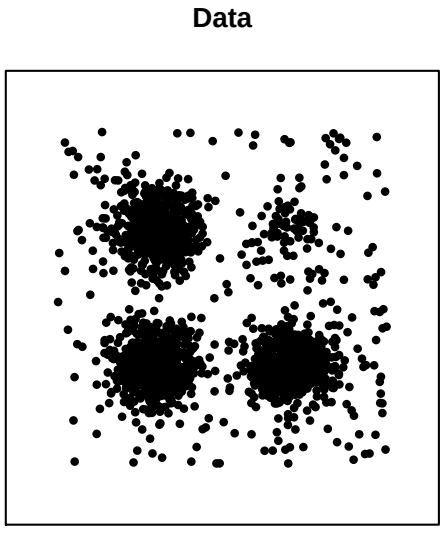
Example: Outliers, improved k-means

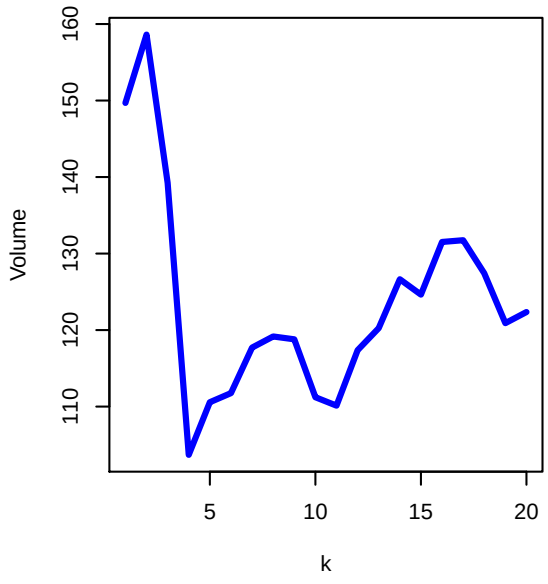


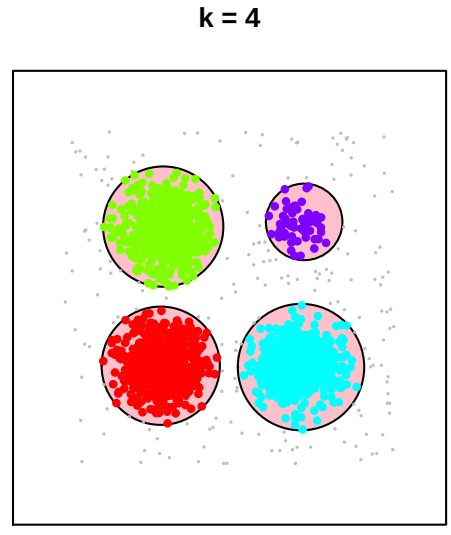




Example

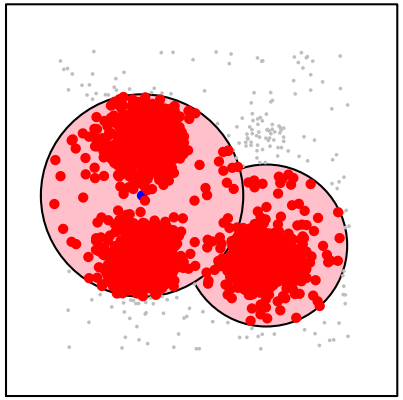




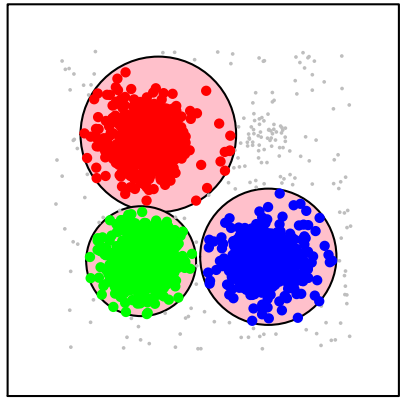


Example

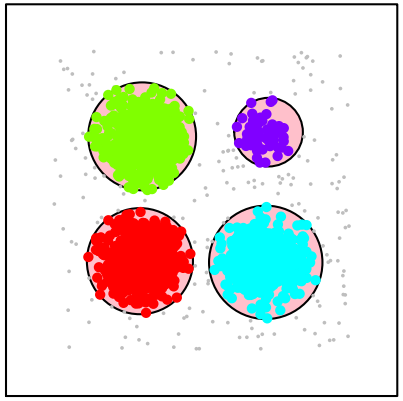
k = 2



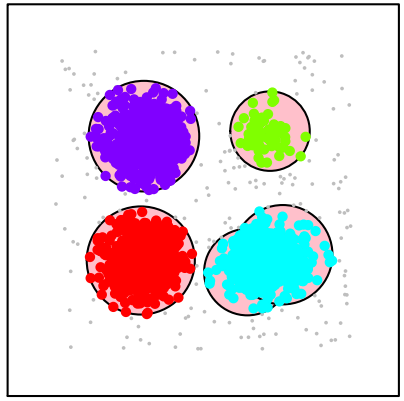
k = 3



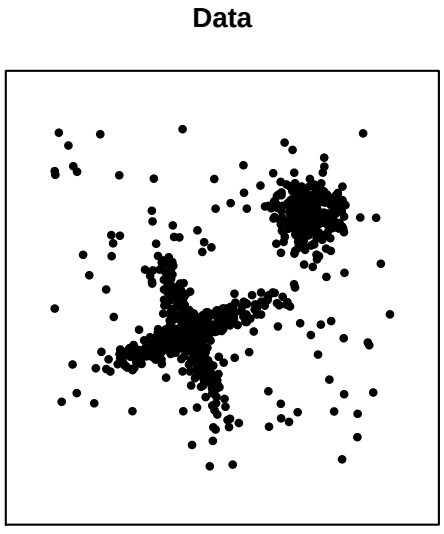
k = 4

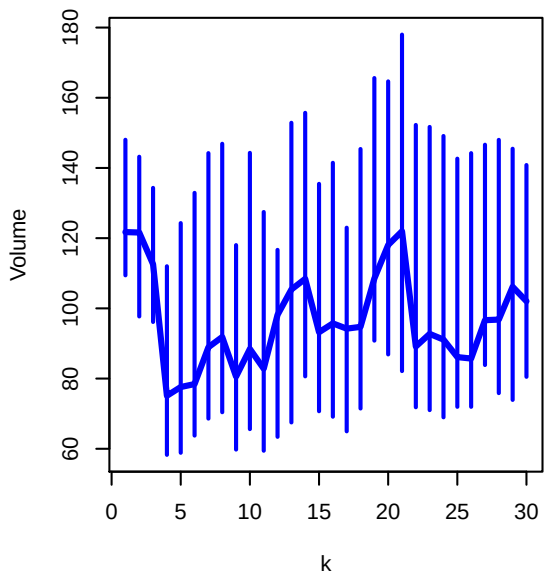


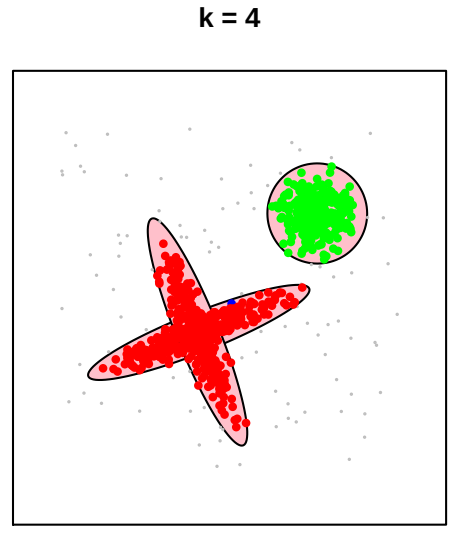
k = 5



Example







Conformal Density Clustering

Let $C_1, \ldots, C_r$ be the connected components of

$$L = \{y : \widehat{p}_h(y) > t\}.$$

Problems:
1. How to choose h?
2. How to choose t?
3. How to find $C_1, \ldots, C_r$?

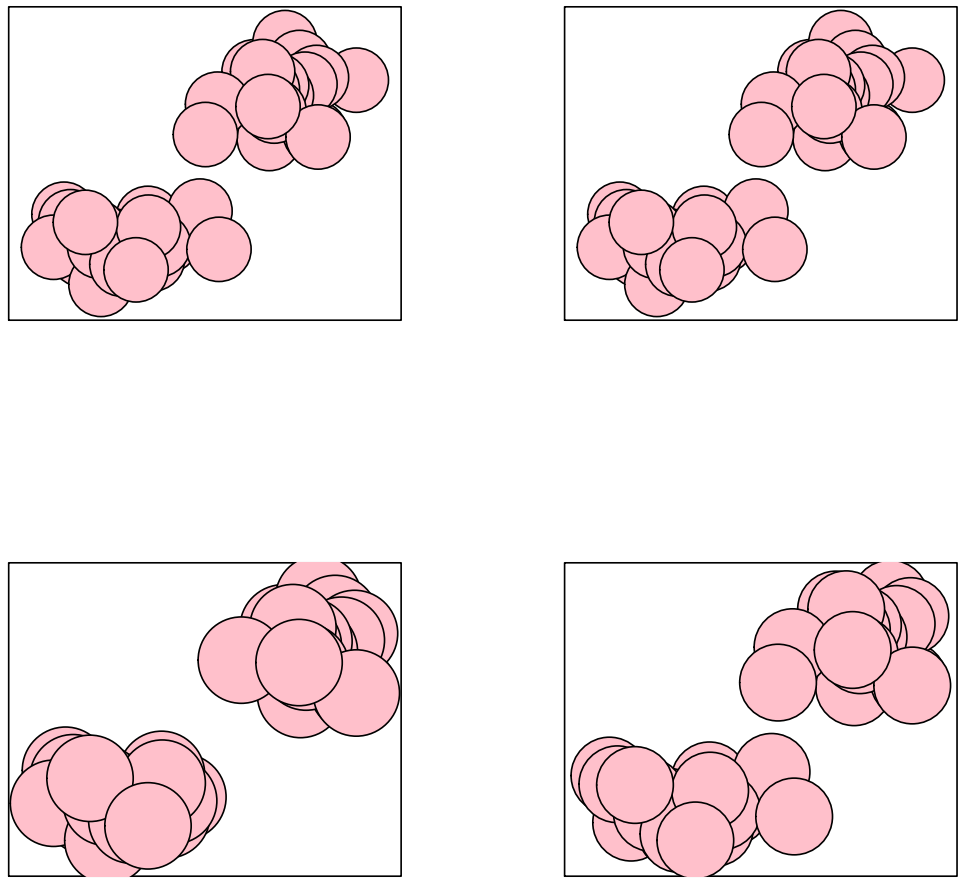
All solved by conformal clustering.

Density Level Set Conformal Clustering

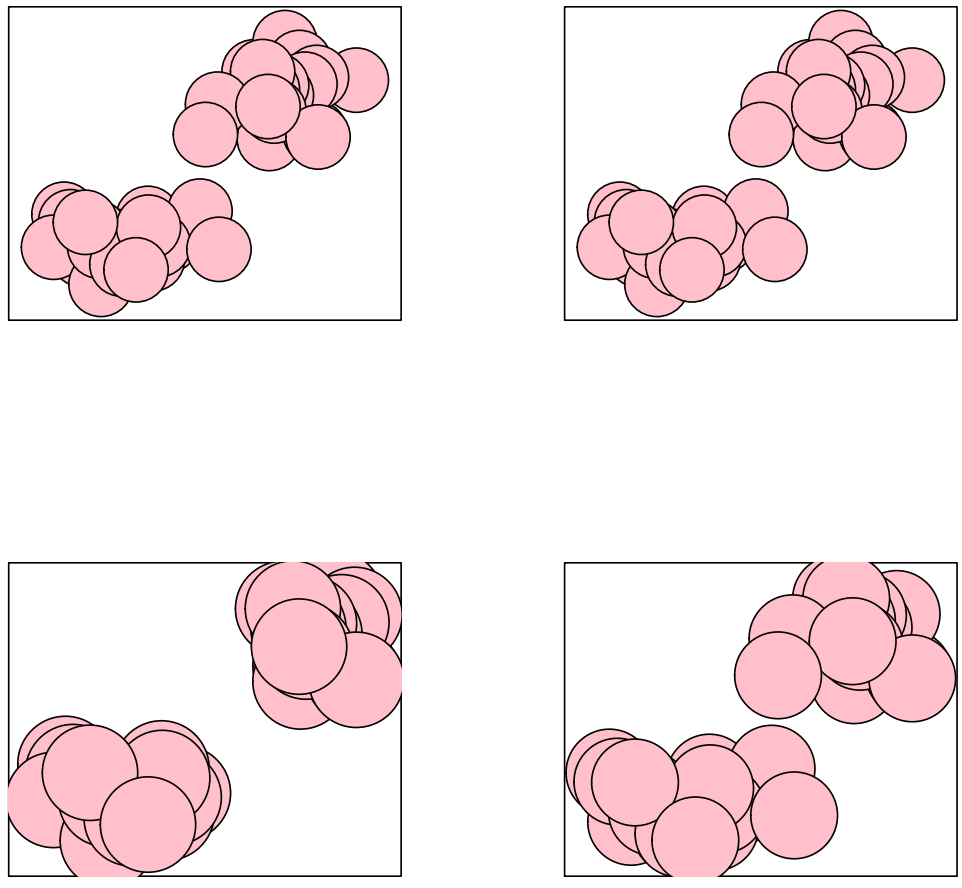
1. Split the data into two halves $\mathcal{Y}_1$ and $\mathcal{Y}_2$.
2. Estimate a density $\widehat{p}$ from $\mathcal{Y}_1$.
3. For a given $t > 0$, let $\mathcal{Y}_1^{(t)} := \{Y_i \in \mathcal{Y}_1 : \widehat{p}(Y_i) \geq L\}$.
4. Compute the residuals $R_i := R(Y_i)$ for $Y_i \in \mathcal{Y}_2$, where $R(y) = \min_{Y_j \in \mathcal{Y}_1^{(t)}} \|y - Y_j\|$.
5. Let t_α be the $1 - \alpha$ quantile of the residuals.
6. Let $C = \bigcup_{Y_j \in \mathcal{Y}_1^{(t)}} B(Y_j, t_\alpha)$.

Choose h and L to minimize volume.
Conformalization makes this very robust to h and L because t_α adapts.

Example: Small L, 4 bandwidths



Example: Large L, 4 bandwidths



Part III: Random Effects (with Robin Dunn)

Random distributions (subjects)

$$P_1, \ldots, P_k \sim \Pi$$

and observe data $\mathcal{D}_j$ drawn from P_j.

Part III: Random Effects (with Robin Dunn)

Random distributions (subjects)

$$P_1, \ldots, P_k \sim \Pi$$

and observe data $\mathcal{D}_j$ drawn from P_j.

Problem 1: Predict $Y \sim P_{k+1}$ (new subject).
Problem 2: Predict $Y \sim P_j$ for some $1 \leq j \leq k$; new observation on existing subject. (Shrinkage.)

Part III: Random Effects (with Robin Dunn)

Random distributions (subjects)

$$P_1, \ldots, P_k \sim \Pi$$

and observe data $\mathcal{D}_j$ drawn from P_j.

Problem 1: Predict $Y \sim P_{k+1}$ (new subject).
Problem 2: Predict $Y \sim P_j$ for some $1 \leq j \leq k$; new observation on existing subject. (Shrinkage.)

I'll focus on Problem 1.

Random Effects: Method 1

Random Effects: Method 1

Select one observation randomly from each group. (These are iid).

Random Effects: Method 1

Select one observation randomly from each group. (These are iid).

Apply any conformal procedure at level α/N.

Random Effects: Method 1

Select one observation randomly from each group. (These are iid).

Apply any conformal procedure at level α/N.

Repeat N times.

Random Effects: Method 1

Select one observation randomly from each group. (These are iid).

Apply any conformal procedure at level α/N.

Repeat N times.

Set $C = \bigcap_{j=1}^{N} C_j$.

Random Effects: Method 1

Select one observation randomly from each group. (These are iid).

Apply any conformal procedure at level α/N.

Repeat N times.

Set $C = \bigcap_{j=1}^{N} C_j$.

This is always valid. Optimal N seems to be $N = 1$.

Random Effects: Method 2. Double Conformal Inference

Use a parametric working model. (Does not need to be correct).
Apply conformal prediction to get C_j for group j at level $1-\delta$.
In fact, these are iid random sets such that $\int_{C_j} dP_j = 1 - \delta + o_P(1)$.

Random Effects: Method 2. Double Conformal Inference

Use a parametric working model. (Does not need to be correct).
Apply conformal prediction to get C_j for group j at level $1-\delta$.
In fact, these are iid random sets such that $\int_{C_j} dP_j = 1 - \delta + o_P(1)$.

We have m iid random sets $C_1, \ldots, C_m$.

Random Effects: Method 2. Double Conformal Inference

Use a parametric working model. (Does not need to be correct).
Apply conformal prediction to get C_j for group j at level $1-\delta$.
In fact, these are iid random sets such that $\int_{C_j} dP_j = 1 - \delta + o_P(1)$.

We have m iid random sets $C_1, \ldots, C_m$.

Apply conformal inference to the sets at level β to get $\mathcal{C}$ such that

$$P(C_{m+1} \in \mathcal{C}) \geq 1 - \beta.$$

Random Effects: Method 2. Double Conformal Inference

Use a parametric working model. (Does not need to be correct).
Apply conformal prediction to get C_j for group j at level $1-\delta$.
In fact, these are iid random sets such that $\int_{C_j} dP_j = 1-\delta+o_P(1)$.

We have m iid random sets $C_1, \ldots, C_m$.

Apply conformal inference to the sets at level β to get $\mathcal{C}$ such that

$$P(C_{m+1} \in \mathcal{C}) \geq 1-\beta.$$

If $\delta+\beta \leq \alpha$ then

$$P(Y \in \mathcal{C}) \geq 1-\alpha.$$

Random Effects: Method 2. Double Conformal Inference

Use a parametric working model. (Does not need to be correct).
Apply conformal prediction to get C_j for group j at level $1-\delta$.
In fact, these are iid random sets such that $\int_{C_j} dP_j = 1-\delta+o_P(1)$.

We have m iid random sets $C_1, \ldots, C_m$.

Apply conformal inference to the sets at level β to get $\mathcal{C}$ such that

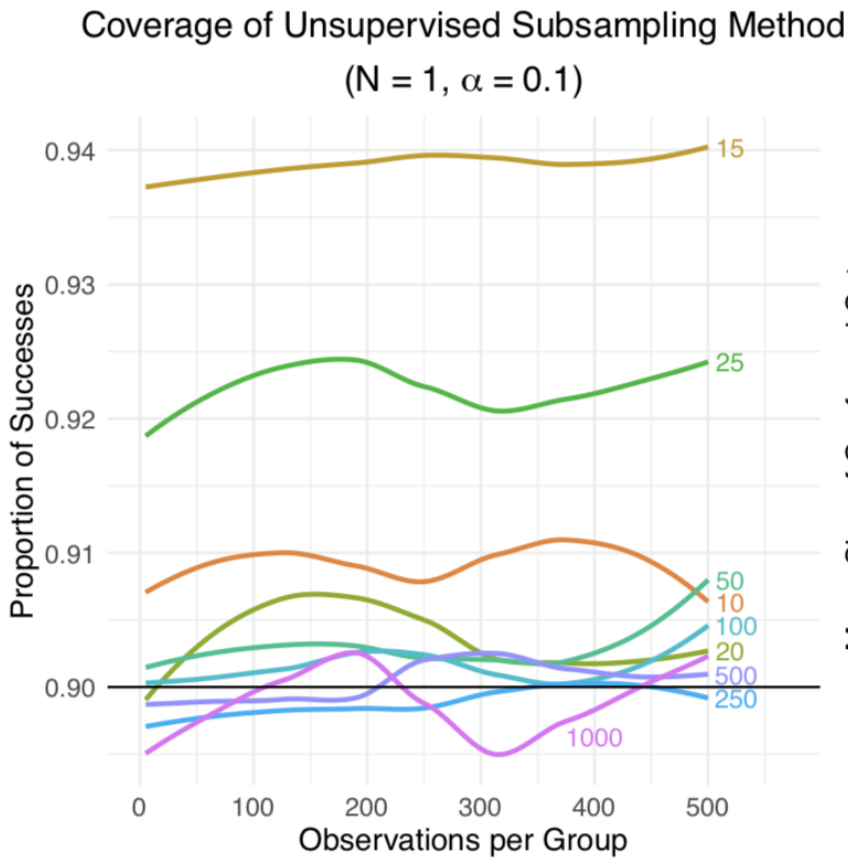
$$P(C_{m+1} \in \mathcal{C}) \geq 1-\beta.$$

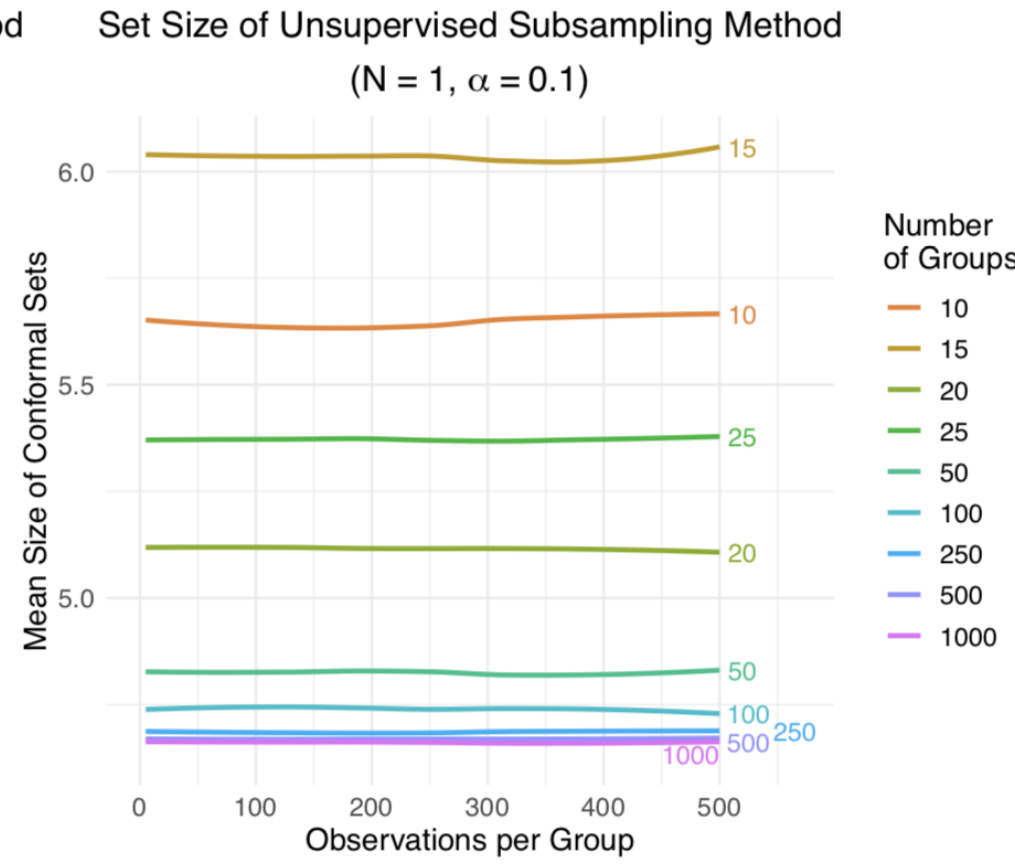
If $\delta+\beta \leq \alpha$ then

$$P(Y \in \mathcal{C}) \geq 1-\alpha.$$

Seems elegant but subsampling works better.

Random Effects





Summary

Summary

In the last few years, we have pursued the goal of model free inference.

Summary

In the last few years, we have pursued the goal of model free inference.

This philosophy can be applied to many problems:
high-dimensional regression, deep learning, random effects, ...

Summary

In the last few years, we have pursued the goal of model free inference.

This philosophy can be applied to many problems:
high-dimensional regression, deep learning, random effects, ...

Conformalizing any optimal procedure preserves its properties without sacrificing general validity.

Summary

In the last few years, we have pursued the goal of model free inference.

This philosophy can be applied to many problems:
high-dimensional regression, deep learning, random effects, ...

Conformalizing any optimal procedure preserves its properties without sacrificing general validity.

Especially important in high-dimensional problems where assumptions are hard to check.

Summary

In the last few years, we have pursued the goal of model free inference.

This philosophy can be applied to many problems:
high-dimensional regression, deep learning, random effects, ...

Conformalizing any optimal procedure preserves its properties without sacrificing general validity.

Especially important in high-dimensional problems where assumptions are hard to check.

Can we get rid of the iid assumption?

Summary

In the last few years, we have pursued the goal of model free inference.

This philosophy can be applied to many problems:
high-dimensional regression, deep learning, random effects, ...

Conformalizing any optimal procedure preserves its properties without sacrificing general validity.

Especially important in high-dimensional problems where assumptions are hard to check.

Can we get rid of the iid assumption?

Code: https://github.com/ryantibs/conformal

Summary

In the last few years, we have pursued the goal of model free inference.

This philosophy can be applied to many problems:
high-dimensional regression, deep learning, random effects, ...

Conformalizing any optimal procedure preserves its properties without sacrificing general validity.

Especially important in high-dimensional problems where assumptions are hard to check.

Can we get rid of the iid assumption?

Code: https://github.com/ryantibs/conformal

THE END