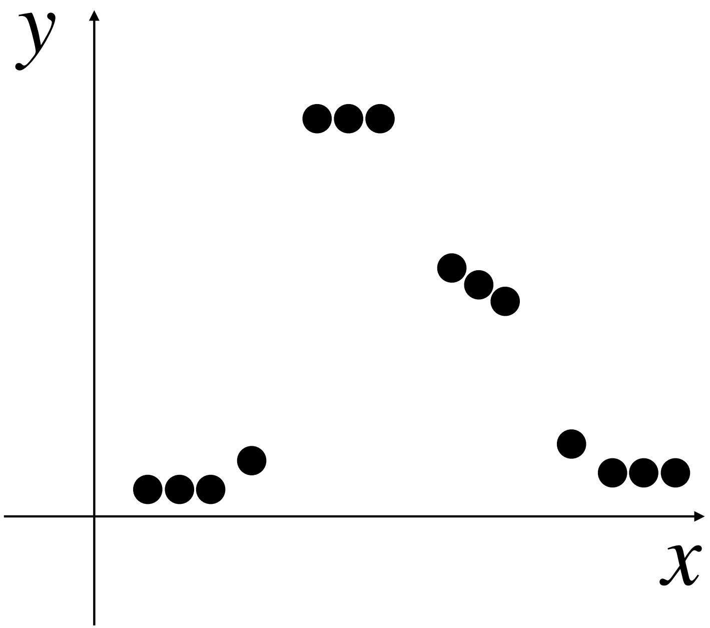
Ensemble Methods:
Bagging & Random Forest

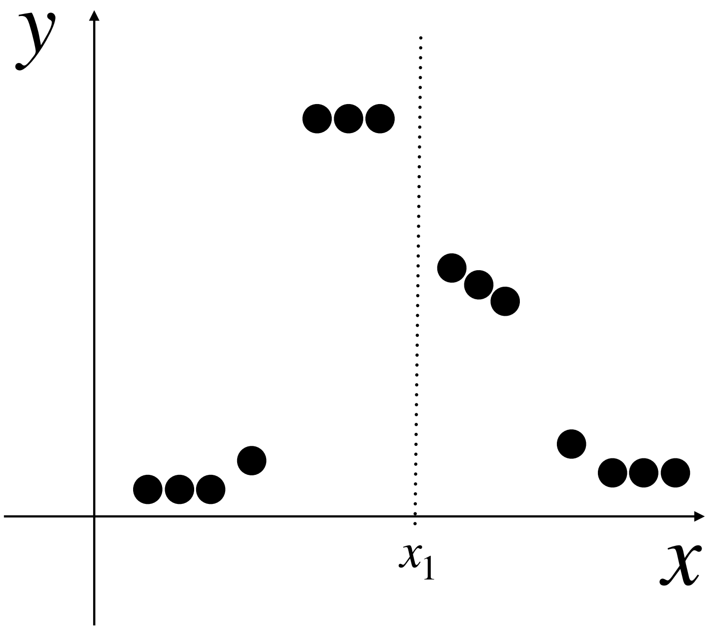
Recap on Decision (Regression) Tree

Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^{n}, \quad (x_i, y_i) \sim P$



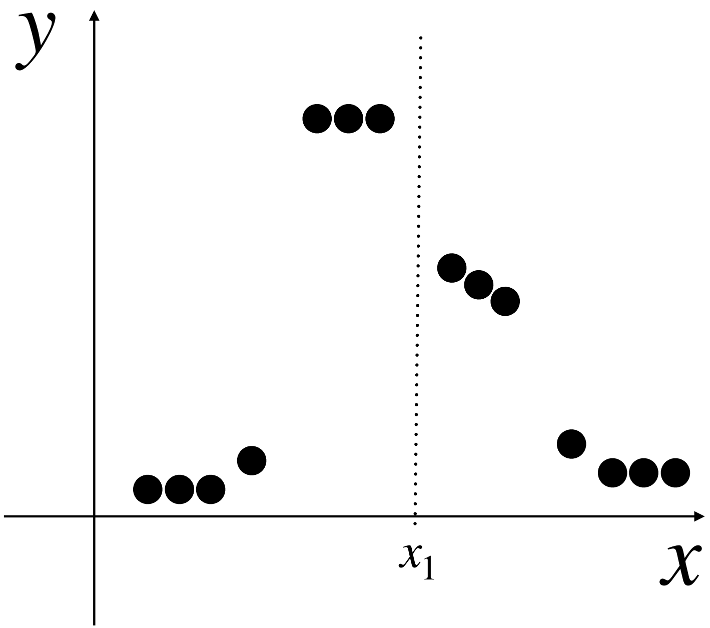
Recap on Decision (Regression) Tree

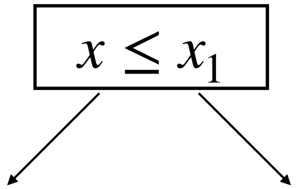
Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n, \quad (x_i, y_i) \sim P$



Recap on Decision (Regression) Tree

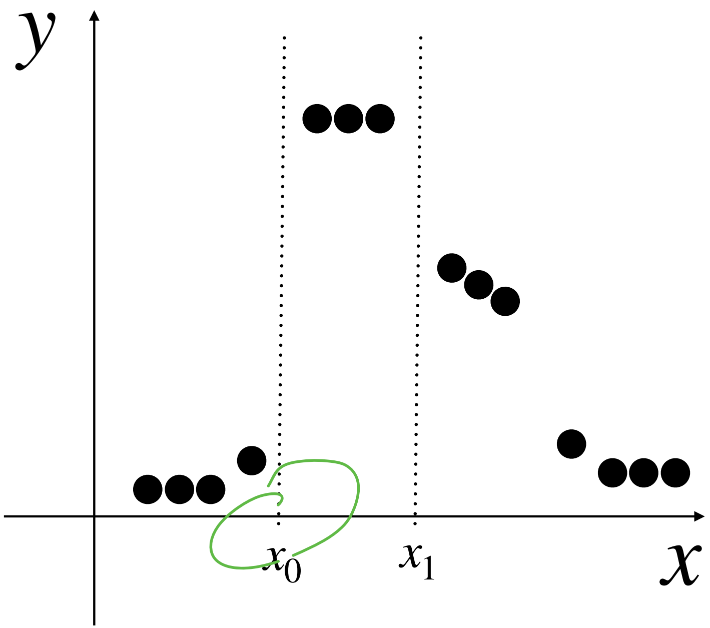
Regression dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^{n}, \quad (x_i, y_i) \sim P$

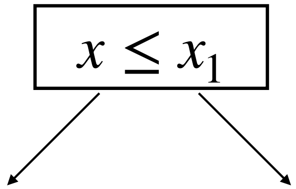




Recap on Decision (Regression) Tree

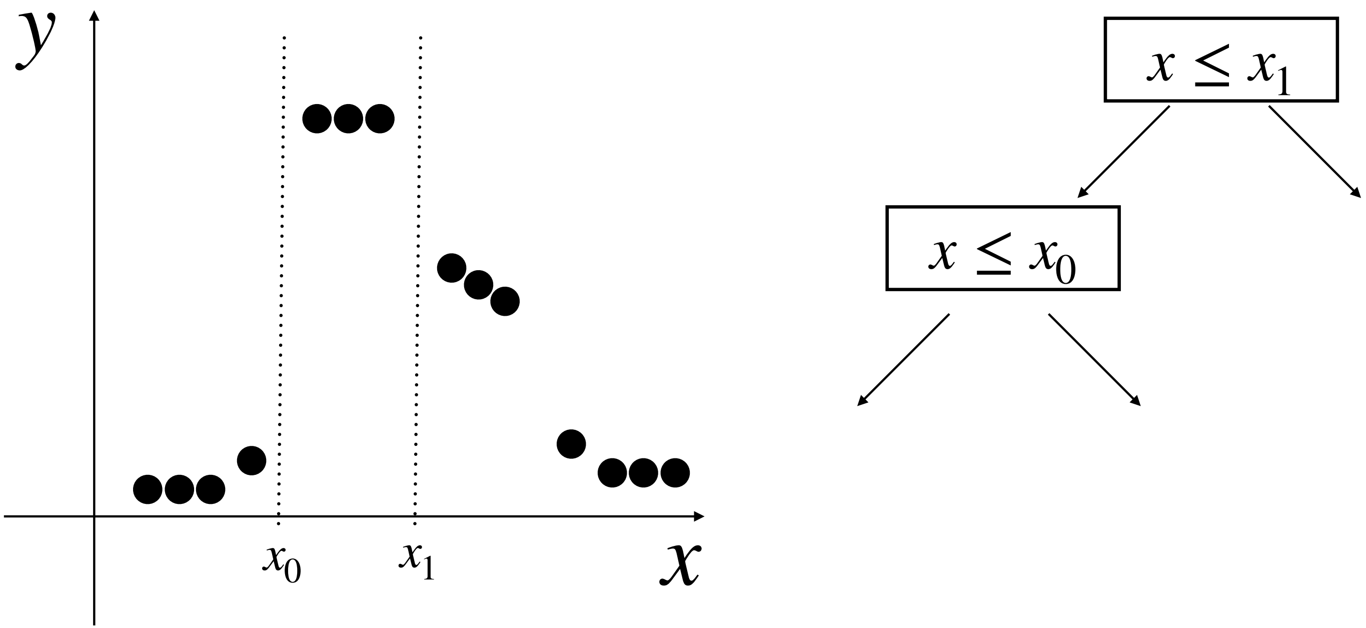
Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n, \quad (x_i, y_i) \sim P$





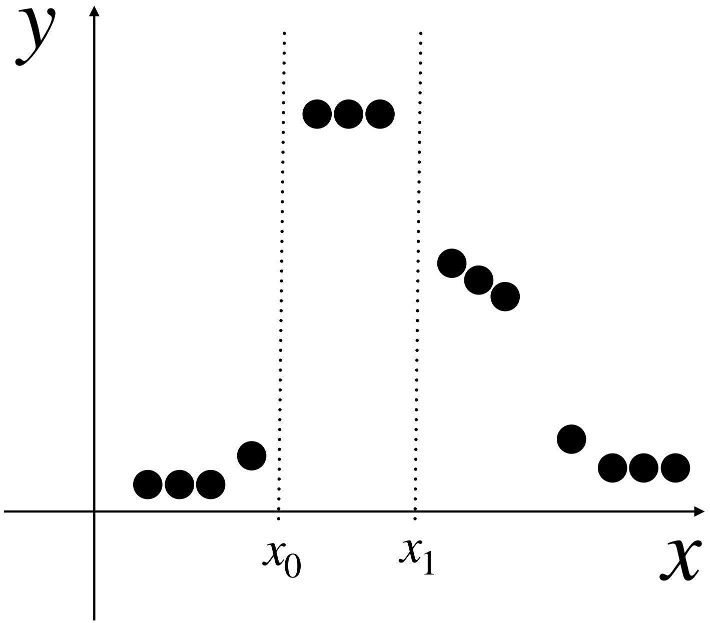
Recap on Decision (Regression) Tree

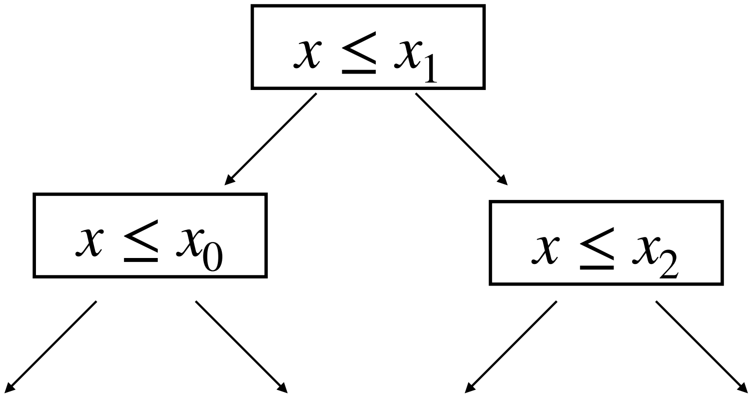
Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n, \quad (x_i, y_i) \sim P$



Recap on Decision (Regression) Tree

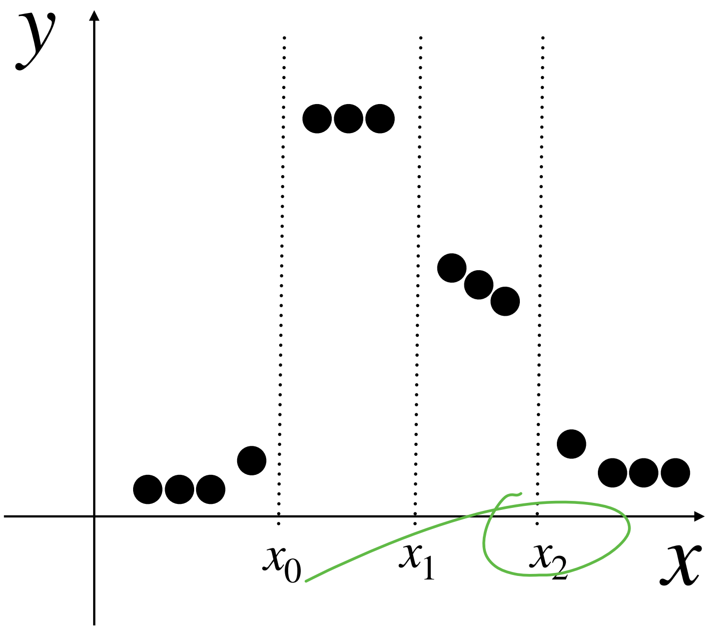
Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n, \quad (x_i, y_i) \sim P$

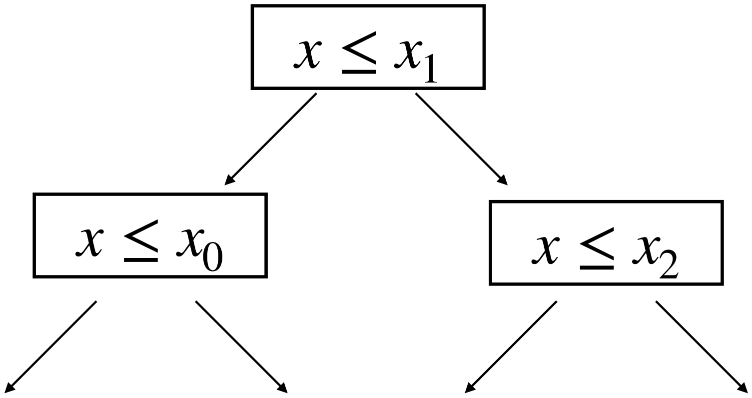




Recap on Decision (Regression) Tree

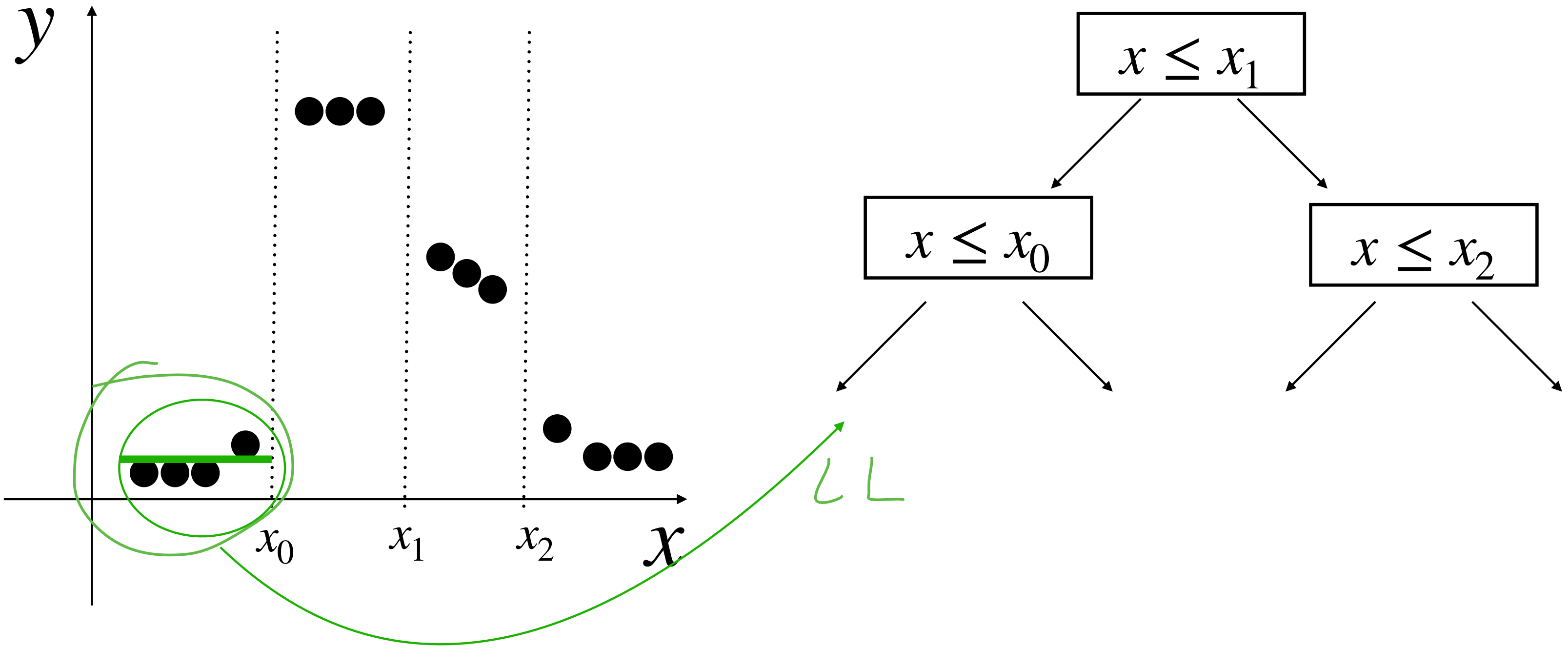
Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^{n}, \quad (x_i, y_i) \sim P$





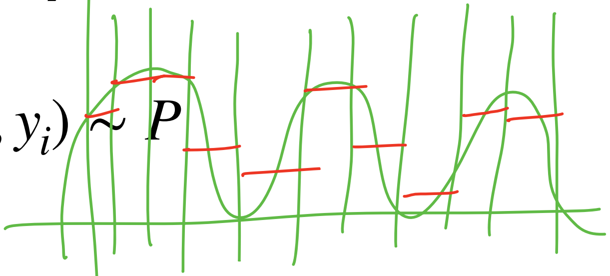
Recap on Decision (Regression) Tree

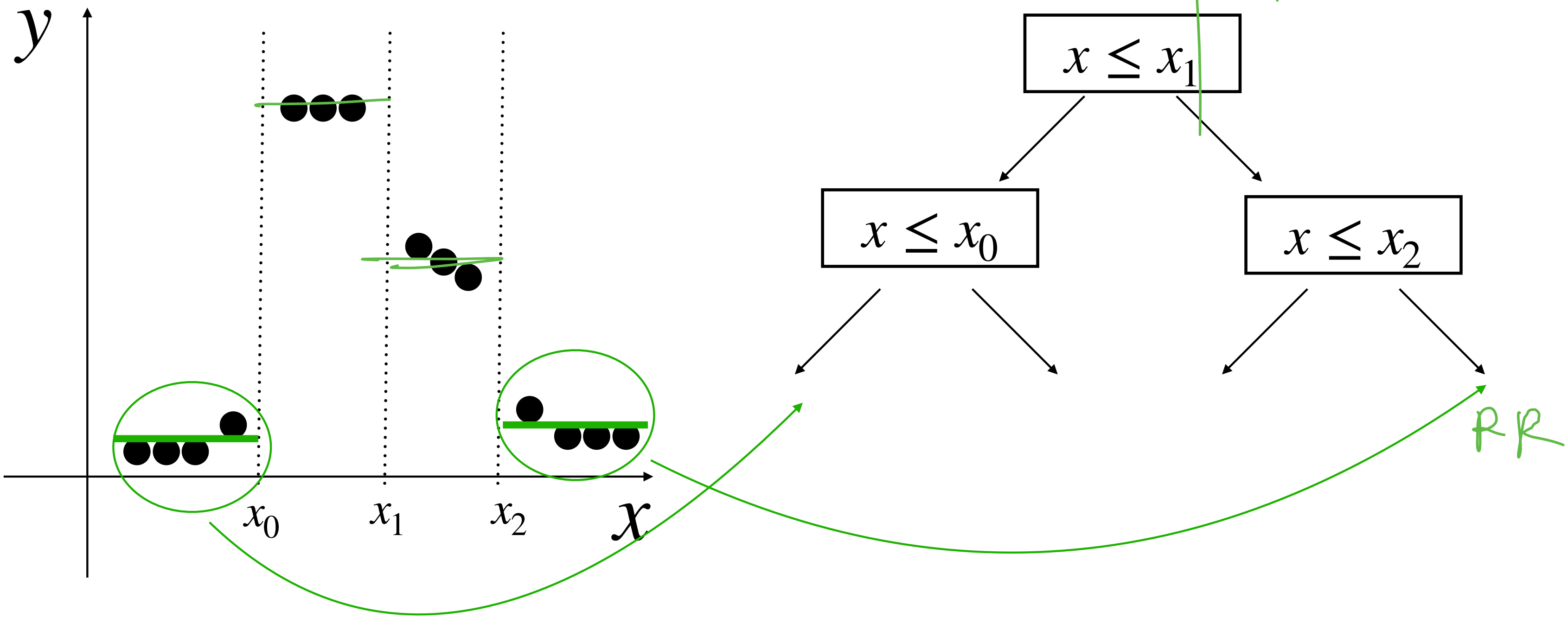
Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n, \quad (x_i, y_i) \sim P$



Recap on Decision (Regression) Tree

Regression dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^{n}, \quad (x_i, y_i) \sim P$

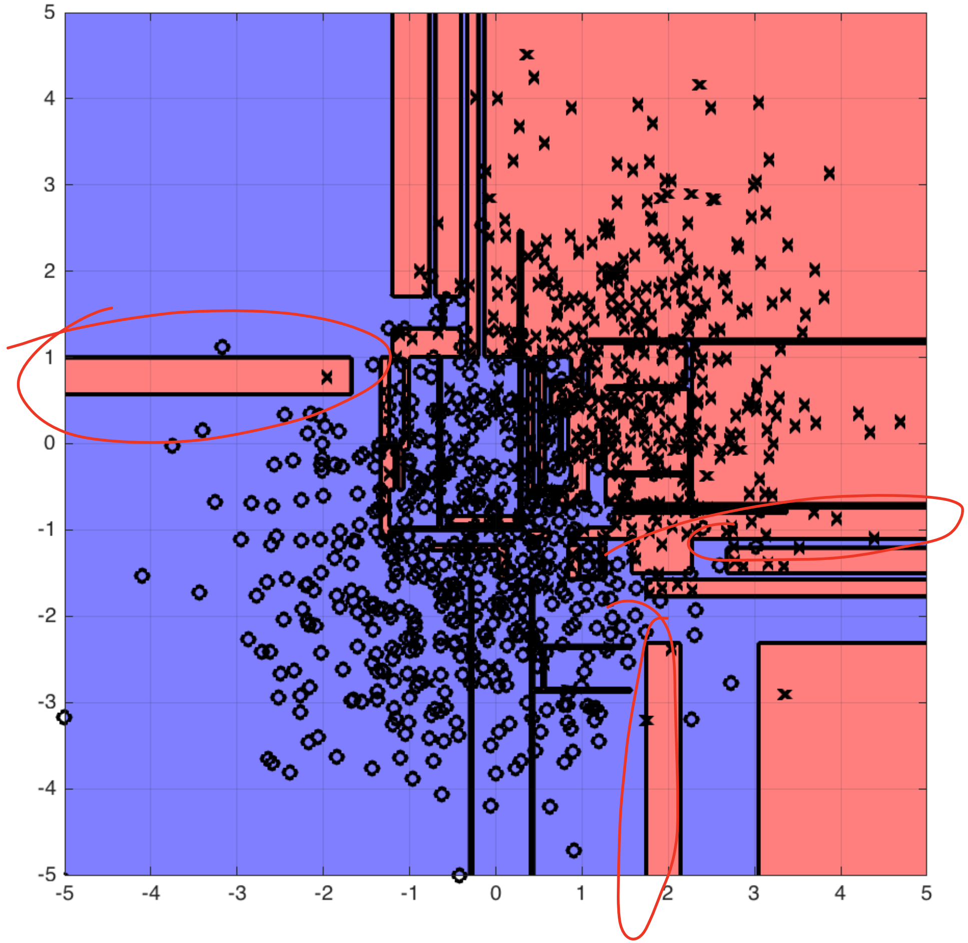




Issues of Decision Trees

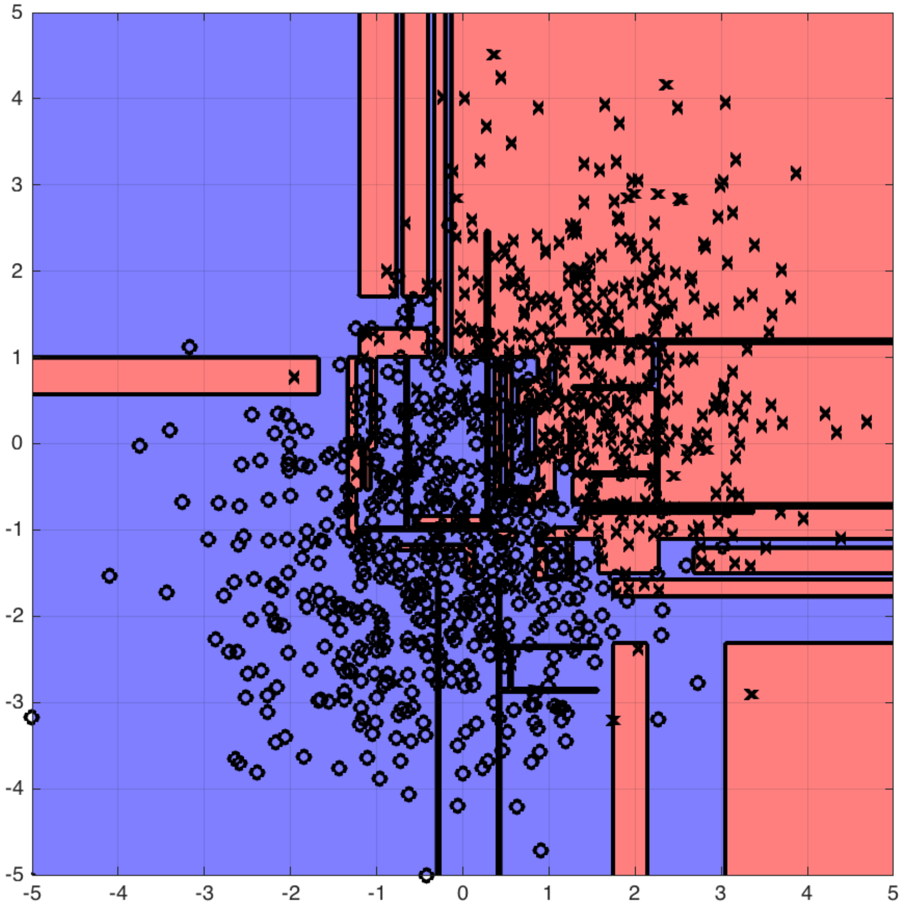
Decision Tree can have high variance, i.e., overfilling!

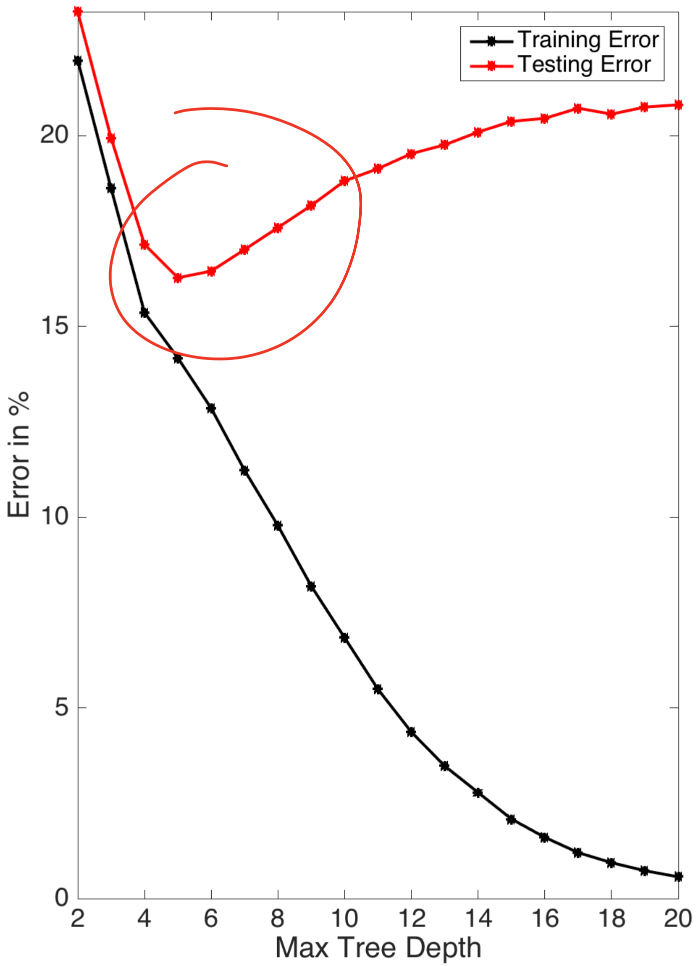
Issues of Decision Trees



Decision Tree can have high variance, i.e., overfilling!

Issues of Decision Trees





Decision Tree can have high variance, i.e., overfilling!

Common regularizations in Decision Trees

1. Minimum number of examples per leaf

No split if # of examples < threshold

Common regularizations in Decision Trees

1. Minimum number of examples per leaf

No split if # of examples < threshold

2. Maximum Depth

No split if it hits depth limit

Common regularizations in Decision Trees

1. Minimum number of examples per leaf

No split if # of examples < threshold

2. Maximum Depth

No split if it hits depth limit

3. Maximum number of nodes

Stop the tree if it hits max # of nodes

Outline of Today

1. Variance Reduction using averaging

2. Bagging: Bootstrap Aggregation

3. Random Forest

Variance Reduction via Averaging

Consider i.i.d random variables $\{x_i\}_{i=1}^{n}, \quad x_i \sim \mathcal{N}(0, \sigma^2)$

$$\text{Var}(x_i) = \sigma^2$$

Variance Reduction via Averaging

Consider i.i.d random variables $\{x_i\}_{i=1}^n, \quad x_i \sim \mathcal{N}(0,\sigma^2)$

$$\text{Var}(x_i) = \sigma^2$$

$n = 2$

Q: what is the variance of $\bar{x} = \sum_{i=1}^{n} x_i / n$

$n \to \infty$

$\bar{x} \mapsto E[x] = 0$

$$\text{Var}(\bar{x}) = \frac{\sigma^2}{n}$$

$n \to \infty \quad \text{Var}(\bar{x}) \to 0$

Variance Reduction via Averaging

Consider i.i.d random variables $\{x_i\}_{i=1}^n, \quad x_i \sim \mathcal{N}(0, \sigma^2)$

$$\text{Var}(x_i) = \sigma^2$$

Q: what is the variance of $\bar{x} = \sum_{i=1}^{n} x_i / n$

Avg significantly reduced variance!

Variance Reduction via Averaging

Consider (possibly correlated) random variables $\{x_i\}_{i=1}^{n}, \quad x_i \sim \mathcal{N}(0, \sigma^2)$

Variance Reduction via Averaging

Consider (possibly correlated) random variables $\{x_i\}_{i=1}^{n}, \quad x_i \sim \mathcal{N}(0,\sigma^2)$

$n=3 \qquad x_i \sim N(0, \sigma^2)$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \sigma^2 & \sigma_{1,2} & \sigma_{1,3} \\ \sigma_{2,1} & \sigma^2 & \sigma_{2,3} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma^2 \end{bmatrix}\right)$$

Variance Reduction via Averaging

Consider (possibly correlated) random variables $\{x_i\}_{i=1}^{n}, \quad x_i \sim \mathcal{N}(0,\sigma^2)$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \sigma^2 & \sigma_{1,2} & \sigma_{1,3} \\ \sigma_{2,1} & \sigma^2 & \sigma_{2,3} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma^2 \end{bmatrix}\right)$$

$$\sigma_{i,j} = \mathbb{E}[x_i x_j] \qquad \sigma_{ij} > 0$$

Variance Reduction via Averaging

Consider (possibly correlated) random variables $\{x_i\}_{i=1}^n, \quad x_i \sim \mathcal{N}(0,\sigma^2)$

Q: what is the variance of $\bar{x} = \sum_{i=1}^{3} x_i/3$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \sigma^2 & \sigma_{1,2} & \sigma_{1,3} \\ \sigma_{2,1} & \sigma^2 & \sigma_{2,3} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma^2 \end{bmatrix}\right)$$

$$\sigma_{i,j} = \mathbb{E}[x_i x_j]$$

Variance Reduction via Averaging

Consider (possibly correlated) random variables $\{x_i\}_{i=1}^{n}, \quad x_i \sim \mathcal{N}(0,\sigma^2)$

$$\mathrm{Var}\left(\frac{x_1+x_2+x_3}{3}\right) = \mathbb{E}\left[\left(\frac{x_1+x_2+x_3}{3}\right)^2\right]$$

Q: what is the variance of $\bar{x} = \sum_{i=1}^{3} x_i/3$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \sigma^2 & \sigma_{1,2} & \sigma_{1,3} \\ \sigma_{2,1} & \sigma^2 & \sigma_{2,3} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma^2 \end{bmatrix}\right)$$

A: $\mathrm{Var}(\bar{x}) = \sigma^2/3 + \sum_{i \neq j} \sigma_{i,j}/9$ $= 1$

$\sigma = 1$

$\sigma_{i,j} = 1$

$$\sigma_{i,j} = \mathbb{E}[x_i x_j]$$

Variance Reduction via Averaging

Consider (possibly correlated) random variables $\{x_i\}_{i=1}^{n}, \quad x_i \sim \mathcal{N}(0, \sigma^2)$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \sigma^2 & \sigma_{1,2} & \sigma_{1,3} \\ \sigma_{2,1} & \sigma^2 & \sigma_{2,3} \\ \sigma_{3,1} & \sigma_{3,2} & \sigma^2 \end{bmatrix}\right)$$

$$\sigma_{i,j} = \mathbb{E}[x_i x_j]$$

Q: what is the variance of $\bar{x} = \sum_{i=1}^{3} x_i/3$

A: $\mathrm{Var}(\bar{x}) = \sigma^2/3 + \sum_{i \neq j} \sigma_{i,j}/9$

Worst case: when these RVs are positively correlated, averaging may not reduce variance

Outline of Today

1. Variance Reduction using averaging

2. Bagging: Bootstrap Aggregation

3. Random Forest

Why Bagging

Consider train Decision Tree, i.e., $\hat{h} = \text{ID3}(\mathscr{D})$

Why Bagging

Consider train Decision Tree, i.e., $\hat{h} = \text{ID3}(\mathscr{D})$

$\hat{h}$ is a random quantity + it has high variance

Why Bagging

Consider train Decision Tree, i.e., $\hat{h} = \text{ID3}(\mathscr{D})$

$\hat{h}$ is a random quantity + it has high variance

Q: can we learn multiple $\hat{h}$ and perform averaging to reduce variance?

Why Bagging

Consider train Decision Tree, i.e., $\hat{h} = \text{ID3}(\mathscr{D})$

$\hat{h}$ is a random quantity + it has high variance

Q: can we learn multiple $\hat{h}$ and perform averaging to reduce variance?

Yes, we do this via Bootstrap

Detour: Bootstrapping

$z_i = (x_i, y_i)$

Consider dataset $\mathscr{D} = \{z_i\}_{i=1}^n$, $z_i \sim P$

Detour: Bootstrapping

Consider dataset $\mathscr{D} = \{z_i\}_{i=1}^n$, $z_i \sim P$

Let us approximate P with the following discrete distribution:

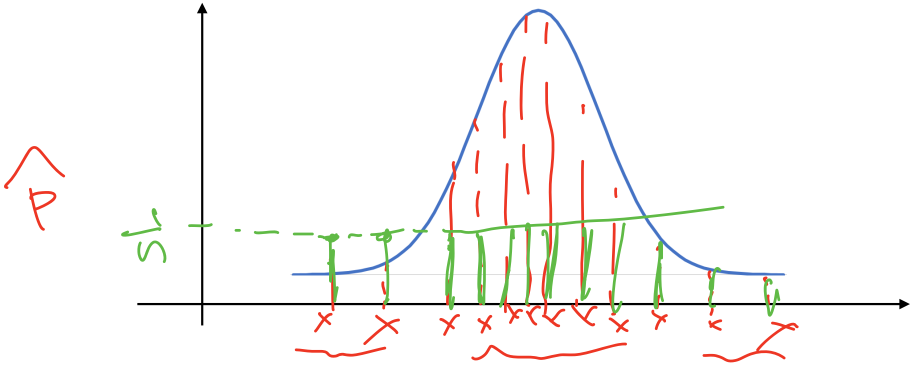
$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Detour: Bootstrapping

Consider dataset $\mathscr{D} = \{z_i\}_{i=1}^{n}, z_i \sim P$

Let us approximate P with the following discrete distribution:

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$



Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Why $\hat{P}$ can be regarded as an approximation of P?

$$E_{z \sim p}[z]$$

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Why $\hat{P}$ can be regarded as an approximation of P?

1. We can use $\hat{P}$ to approximate P's mean and variance, i.e.,

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Why $\hat{P}$ can be regarded as an approximation of P?

1. We can use $\hat{P}$ to approximate P's mean and variance, i.e.,

$$\mathbb{E}_{z \sim \hat{P}}[z] = \sum_{i=1}^{n} \frac{z_i}{n}$$

$\hat{P}$ $\varphi(z_i)$

$= \frac{1}{n}$

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Why $\hat{P}$ can be regarded as an approximation of P?

1. We can use $\hat{P}$ to approximate P's mean and variance, i.e.,

$$\mathbb{E}_{z \sim \hat{P}}[z] = \sum_{i=1}^{n} \frac{z_i}{n} \to \mathbb{E}_{z \sim P}[z]$$

LLN

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Why $\hat{P}$ can be regarded as an approximation of P?

1. We can use $\hat{P}$ to approximate P's mean and variance, i.e.,

$$\mathbb{E}_{z\sim\hat{P}}[z] = \sum_{i=1}^{n} \frac{z_i}{n} \rightarrow \mathbb{E}_{z\sim P}[z] \qquad \mathbb{E}_{z\sim\hat{P}}[z^2] = \sum_{i=1}^{n} z_i^2/n$$

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Why $\hat{P}$ can be regarded as an approximation of P?

1. We can use $\hat{P}$ to approximate P's mean and variance, i.e.,

$$\mathbb{E}_{z\sim\hat{P}}[z] = \sum_{i=1}^{n} \frac{z_i}{n} \to \mathbb{E}_{z\sim P}[z] \qquad \mathbb{E}_{z\sim\hat{P}}[z^2] = \sum_{i=1}^{n} z_i^2/n \to \mathbb{E}_{z\sim P}[z^2]$$

$$\mathbb{E}_{z\sim\hat{P}}[z^3] \longmapsto \mathbb{E}_{z\sim P}[z^3]$$

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Why $\hat{P}$ can be regarded as an approximation of P?

1. We can use $\hat{P}$ to approximate P's mean and variance, i.e.,

$$\mathbb{E}_{z\sim\hat{P}}[z] = \sum_{i=1}^{n} \frac{z_i}{n} \rightarrow \mathbb{E}_{z\sim P}[z] \qquad \mathbb{E}_{z\sim\hat{P}}[z^2] = \sum_{i=1}^{n} z_i^2/n \rightarrow \mathbb{E}_{z\sim P}[z^2]$$

2. In fact for any $f : Z \rightarrow \mathbb{R}$

f(z) ex, z³

$$\mathbb{E}_{z\sim\hat{P}}[f(z)] = \sum_{i=1}^{n} \frac{f(z_i)}{n} \rightarrow \mathbb{E}_{z\sim P}[f(z)]$$

LLN

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Booststrap: treat $\hat{P}$ as if it were the ground truth distribution P!

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Booststrap: treat $\hat{P}$ as if it were the ground truth distribution P!

Now we can draw as many samples as we want from $\hat{P}$!

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Booststrap: treat $\hat{P}$ as if it were the ground truth distribution P!

Now we can draw as many samples as we want from $\hat{P}$!

Q: What's the procedure of drawing n i.i.d samples from $\hat{P}$?

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Booststrap: treat $\hat{P}$ as if it were the ground truth distribution P!

Now we can draw as many samples as we want from $\hat{P}$!

Q: What's the procedure of drawing n i.i.d samples from $\hat{P}$?

A: sample uniform randomly from $\hat{P}$ n times **w/ replacement**

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Booststrap: treat $\hat{P}$ as if it were the ground truth distribution P!

Now we can draw as many samples as we want from $\hat{P}$!

Q: What's the procedure of drawing n i.i.d samples from $\hat{P}$?

A: sample uniform randomly from $\hat{P}$ n times **w/ replacement**

Q: after n samples, what's the probability that z_1 never being sampled?

Bootstrapping

$$\widehat{P}(z_i) = 1/n, \forall i \in [n]$$

Booststrap: treat $\hat{P}$ as if it were the ground truth distribution P!

Now we can draw as many samples as we want from $\hat{P}$!

Q: What's the procedure of drawing n i.i.d samples from $\hat{P}$?

A: sample uniform randomly from $\hat{P}$ n times **w/ replacement**

Q: after n samples, what's the probability that z_1 never being sampled?

A: $(1 - 1/n)^n \rightarrow 1/e, n \rightarrow \infty$

0.36

Bagging: Bootstrap Aggregation

Consider dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P, x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$

Bagging: Bootstrap Aggregation

Consider dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P, x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$

1. Construct $\hat{P}$, s.t., $\hat{P}(x_i, y_i) = 1/n, \forall i \in [n]$

Bagging: Bootstrap Aggregation

Consider dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P, x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$

1. Construct $\hat{P}$, s.t., $\hat{P}(x_i, y_i) = 1/n, \forall i \in [n]$

2. Treat $\hat{P}$ as the ground truth, draw k datasets $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_k$ from $\hat{P}$

$D_1 \sim \hat{P}$

$D_2 \sim \hat{P}$

$D_3 \sim \hat{P}$

$\vdots$

$D_k \sim \hat{P}$

Bagging: Bootstrap Aggregation

Consider dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P, x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$

1. Construct $\hat{P}$, s.t., $\hat{P}(x_i, y_i) = 1/n, \forall i \in [n]$

2. Treat $\hat{P}$ as the ground truth, draw k datasets $\mathscr{D}_1, \mathscr{D}_2, \ldots, \mathscr{D}_k$ from $\hat{P}$

Bootstrapped samples

Bagging: Bootstrap Aggregation

Consider dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P, x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$

1. Construct $\hat{P}$, s.t., $\hat{P}(x_i, y_i) = 1/n, \forall i \in [n]$

2. Treat $\hat{P}$ as the ground truth, draw k datasets $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_k$ from $\hat{P}$

Bootstrapped samples

3. For each $i \in [k]$, train classifier, e.g., $\hat{h}_i = \text{ID3}(\mathcal{D}_i)$

$\rightarrow \{\hat{h}_1, \ldots, \hat{h}_k\}$

Bagging: Bootstrap Aggregation

Consider dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P, x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$

1. Construct $\hat{P}$, s.t., $\hat{P}(x_i, y_i) = 1/n, \forall i \in [n]$

$|D_i|$

2. Treat $\hat{P}$ as the ground truth, draw k datasets $\mathscr{D}_1, \mathscr{D}_2, \ldots, \mathscr{D}_k$ from $\hat{P}$

Bootstrapped samples

3. For each $i \in [k]$, train classifier, e.g., $\hat{h}_i = \text{ID3}(\mathscr{D}_i)$

$\hat{h}_i(x) \to y_i \in R$

4. Averaging / Aggregation, i.e., $\bar{h} = \sum_{i=1}^k \hat{h}_i / k$

$\bar{h}(x) = \sum_{i=1}^{k} y_i / k$

Bagging: Bootstrap Aggregation

Consider dataset $\mathscr{D} = \{x_i, y_i\}_{i=1}^n$, $(x_i, y_i) \sim P, x_i \in \mathbb{R}^d, y_i \in \{-1,1\}$

1. Construct $\hat{P}$, s.t., $\hat{P}(x_i, y_i) = 1/n, \forall i \in [n]$

2. Treat $\hat{P}$ as the ground truth, draw k datasets $\mathscr{D}_1, \mathscr{D}_2, \ldots, \mathscr{D}_k$ from $\hat{P}$

Bootstrapped samples

3. For each $i \in [k]$, train classifier, e.g., $\hat{h}_i = \text{ID3}(\mathscr{D}_i)$

4. Averaging / Aggregation, i.e., $\bar{h} = \sum_{i=1}^{k} \hat{h}_i / k$

The step that reduces Var!

Bagging in Test Time

Given a test example x_{test} (binary classification)

We can use $\{\hat{h}_i\}_{i=1}^{k}$ to form a distribution over labels:

$$\hat{y} = \begin{bmatrix} p \\ 1 - p \end{bmatrix}$$

prob of +1

prob of −1

Bagging in Test Time

Given a test example x_{test}

We can use $\{\hat{h}_i\}_{i=1}^{k}$ to form a distribution over labels:

$$\hat{y} = \begin{bmatrix} p \\ 1 - p \end{bmatrix}$$

where:

$$p = \frac{\text{\# of trees predicting } +1}{k}$$

Bagging reduces variance

$$\bar{h} = \sum_{i=1}^{k} \hat{h}_i / k$$

What happens when $k \to \infty$?

Bagging reduces variance

$$\bar{h} = \sum_{i=1}^{k} \hat{h}_i / k$$

What happens when $k \to \infty$?

$\hat{h}_i = ID3(D_i) \quad D_i \sim \hat{P}$

$$\bar{h} \to \mathbb{E}_{\mathcal{D} \sim \hat{P}} \left[\mathrm{ID3}(\mathcal{D}) \right]$$

Bagging reduces variance

$$\bar{h} = \sum_{i=1}^{k} \hat{h}_i / k$$

What happens when $k \to \infty$?

$$\bar{h} \to \mathbb{E}_{\mathscr{D} \sim \hat{P}} \left[\text{ID3}(\mathscr{D}) \right]$$

$\hat{P} \to P,$ when $n \to \infty$

Bagging reduces variance

$$\bar{h} = \sum_{i=1}^{k} \hat{h}_i / k$$

What happens when $k \to \infty$?

$$\bar{h} \to \mathbb{E}_{\mathcal{D} \sim \hat{P}} \left[\text{ID3}(\mathcal{D}) \right]$$

$\downarrow$ $\hat{P} \to P,$ when $n \to \infty$

$$\mathbb{E}_{\mathcal{D} \sim P} \left[\text{ID3}(\mathcal{D}) \right]$$

Bagging reduces variance

$$\bar{h} = \sum_{i=1}^{k} \hat{h}_i / k$$

What happens when $k \to \infty$?

$$\bar{h} \to \mathbb{E}_{\mathscr{D}\sim\hat{P}}\left[\text{ID3}(\mathscr{D})\right]$$

↓ $\hat{P} \to P$, when $n \to \infty$

$$\mathbb{E}_{\mathscr{D}\sim P}\left[\text{ID3}(\mathscr{D})\right]$$

The expected decision tree (under true P)

Bagging reduces variance

$$\bar{h} = \sum_{i=1}^{k} \hat{h}_i / k$$

What happens when $k \to \infty$?

$$\bar{h} \to \mathbb{E}_{\mathcal{D} \sim \hat{P}} \left[\mathrm{ID3}(\mathcal{D}) \right]$$

↓ $\hat{P} \to P,$ when $n \to \infty$

$$\mathbb{E}_{\mathcal{D} \sim P} \left[\mathrm{ID3}(\mathcal{D}) \right]$$

The expected decision tree (under true P)

Deterministic, i.e., zero variance

Outline of Today

1. Variance Reduction using averaging

2. Bagging: Bootstrap Aggregation

3. Random Forest

Motivation of Random Forest

Consider any two hypothesis $\hat{h}_i, \hat{h}_j, i \neq j$ in Bagging

Motivation of Random Forest

Consider any two hypothesis $\hat{h}_i, \hat{h}_j, i \neq j$ in Bagging

$\hat{h}_j, \hat{h}_i$ are not independent under true distribution P

$$\hat{h}_i \leftarrow D_i \sim \hat{P}$$

$$\hat{h}_j \leftarrow D_j \sim \hat{P}$$

$$D_i \cap D_j \neq \emptyset$$

Motivation of Random Forest

Consider any two hypothesis $\hat{h}_i, \hat{h}_j, i \neq j$ in Bagging

$\hat{h}_j, \hat{h}_i$ are not independent under true distribution P

e.g., $\mathscr{D}_i, \mathscr{D}_j$ have overlap samples

Motivation of Random Forest

Consider any two hypothesis $\hat{h}_i, \hat{h}_j, i \neq j$ in Bagging

$\hat{h}_j, \hat{h}_i$ are not independent under true distribution P

e.g., $\mathscr{D}_i, \mathscr{D}_j$ have overlap samples

Recall that: $\text{Var}(\bar{x}) = \sigma^2/3 + \sum_{i \neq j} \sigma_{i,j}/9$

σ_{ij} as the correlation between $\hat{h}_i$ $\hat{h}_j$

Motivation of Random Forest

Consider any two hypothesis $\hat{h}_i, \hat{h}_j, i \neq j$ in Bagging

$\hat{h}_j, \hat{h}_i$ are not independent under true distribution P

e.g., $\mathcal{D}_i, \mathcal{D}_j$ have overlap samples

Recall that: $\text{Var}(\bar{x}) = \sigma^2/3 + \sum_{i \neq j} \sigma_{i,j}/9$

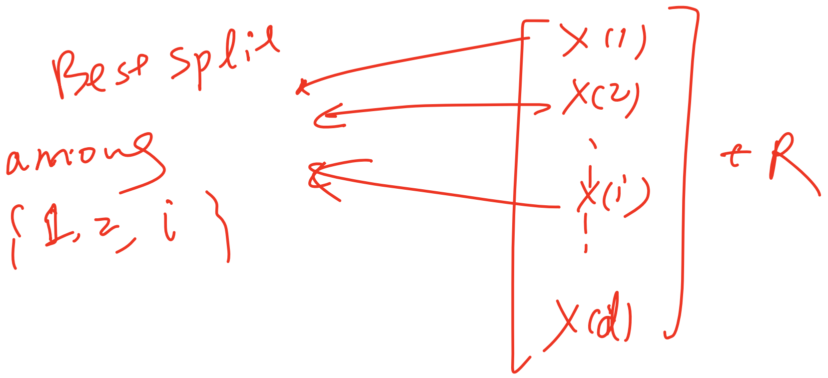
To avoid positive correlation, we want to make $\hat{h}_i, \hat{h}_j$ as independent as possible

Random Forest

Key idea:

In ID3, for every split, **randomly select** k $(k < d)$ many features, find the split **only using these k features**

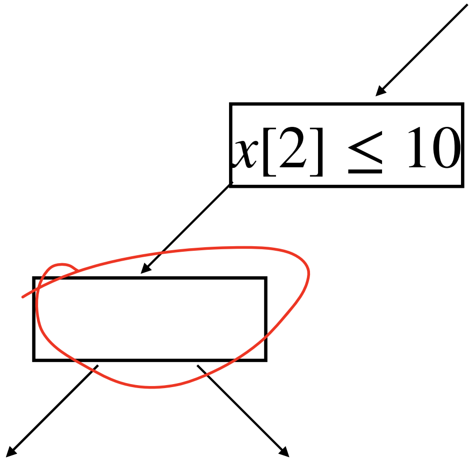
k=3



Random Forest

Key idea:

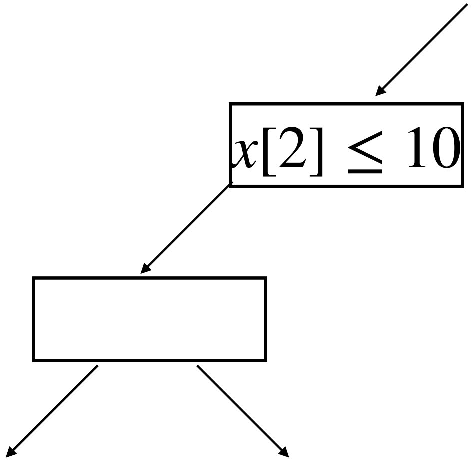
In ID3, for every split, **randomly select** k $(k < d)$ many features, find the split **only using these k features**



Random Forest

Key idea:

In ID3, for every split, **randomly select** k $(k < d)$ many features, find the split **only using these k features**

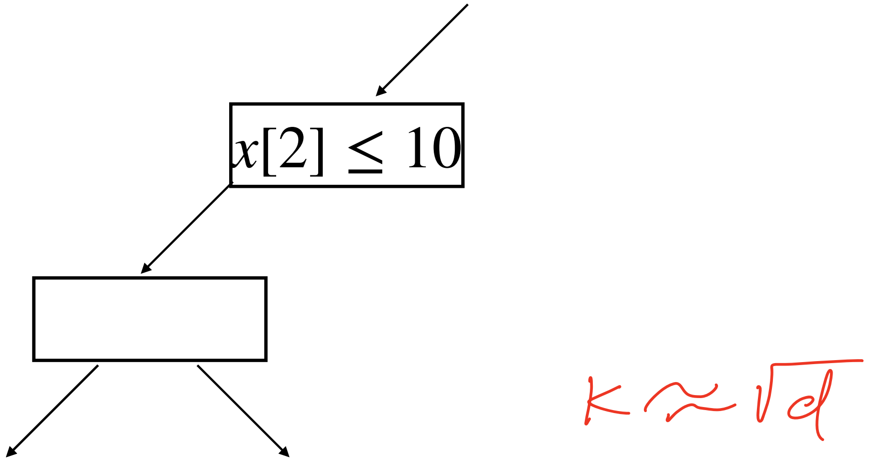


Regular ID3: looking for split in all d dimensions

Random Forest

Key idea:

In ID3, for every split, **randomly select** k $(k < d)$ many features, find the split **only using these k features**



Regular ID3: looking for split in all d dimensions

ID3 in RF: looking for split in k randomly picked dimensions

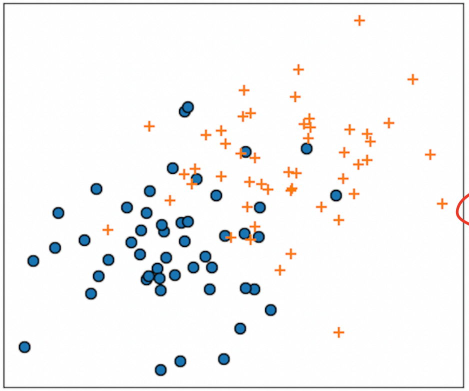
Benefit of Random Forest

By always randomly selecting subset of features for **every tree, and every split**:

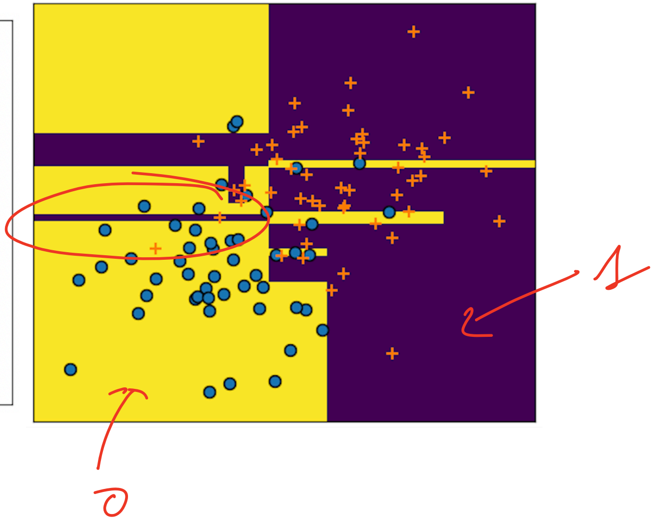
We further reduce the correlation between $\hat{h}_i$ & $\hat{h}_j$

D_i D_j

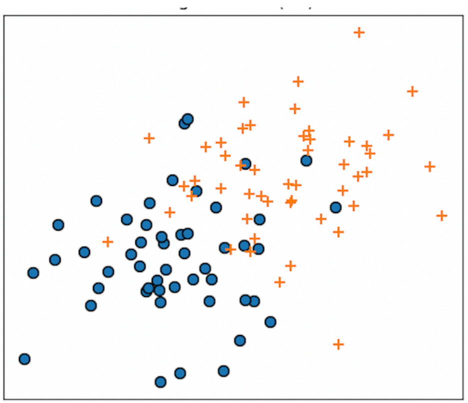
Demo of Random Forest



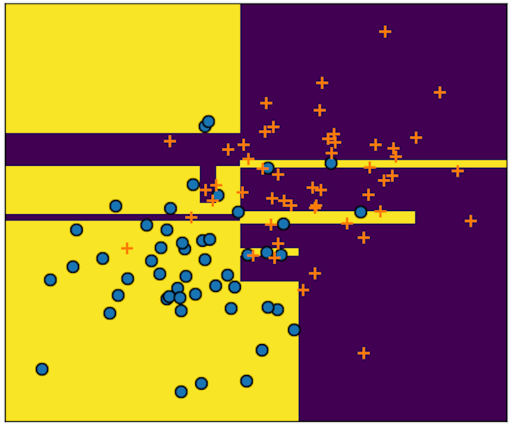
DT w/ Depth 10



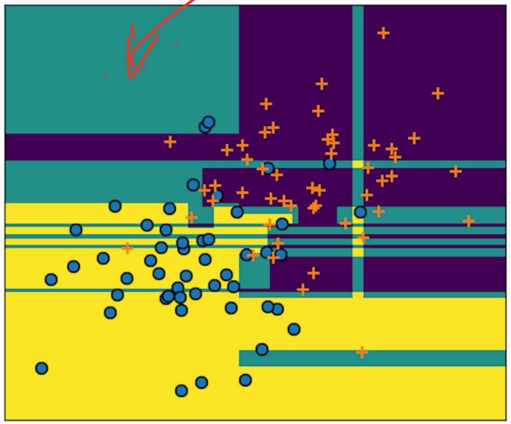
Demo of Random Forest



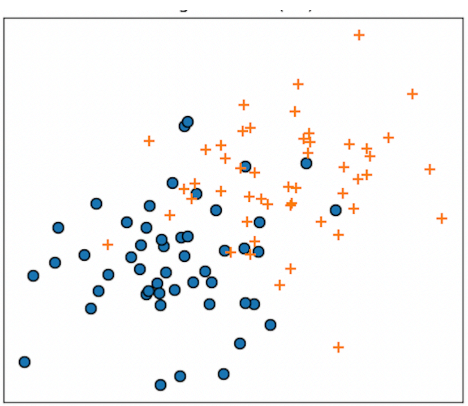
DT w/ Depth 10



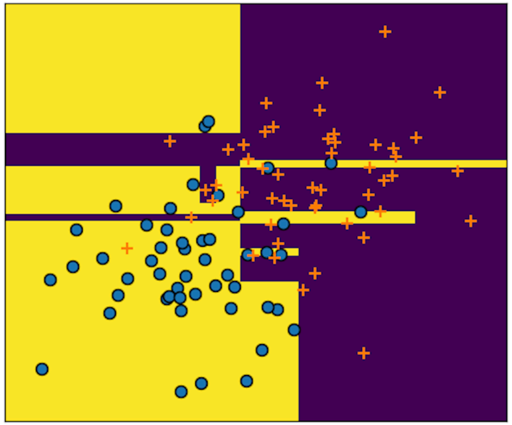
RF w/ 2 trees



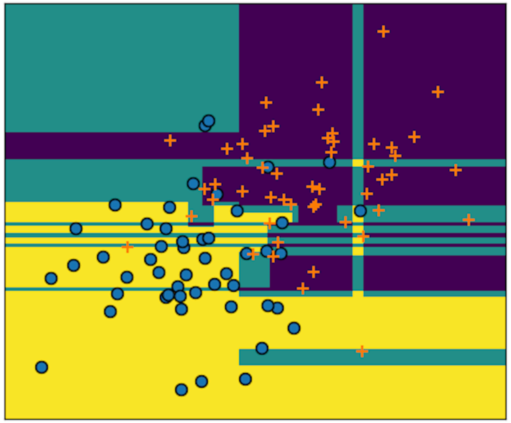
Demo of Random Forest



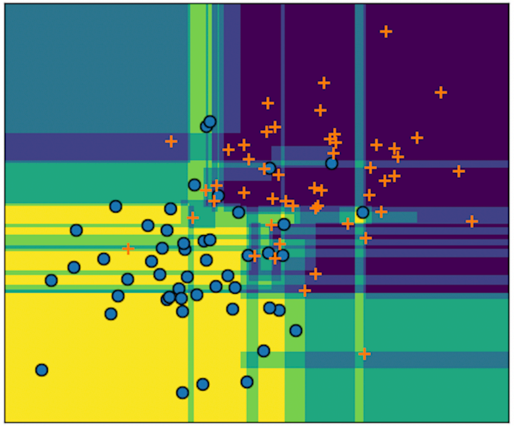
DT w/ Depth 10



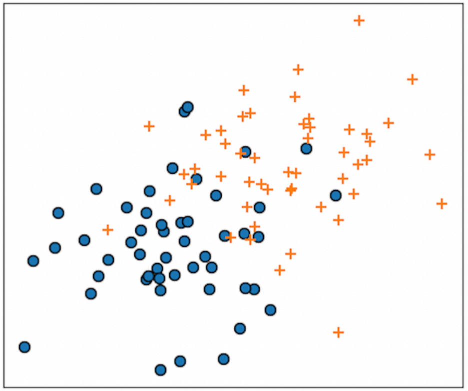
RF w/ 2 trees



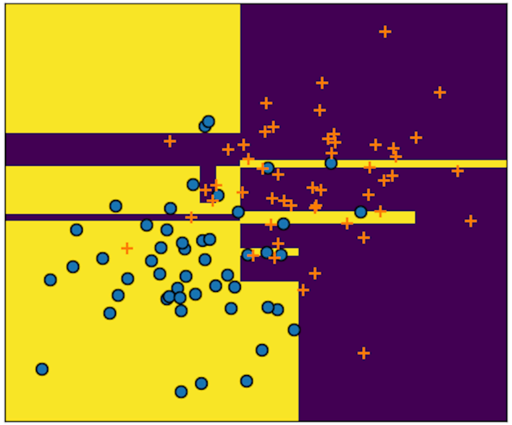
RF w/ 5 trees



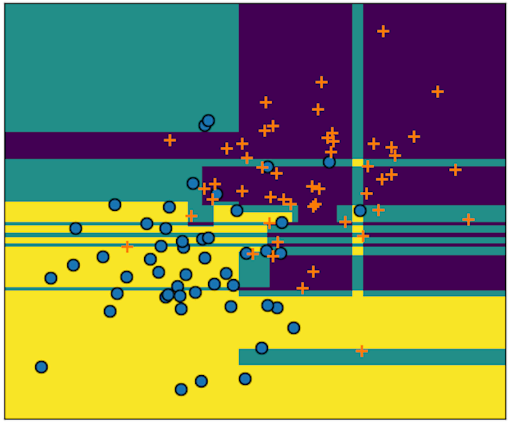
Demo of Random Forest



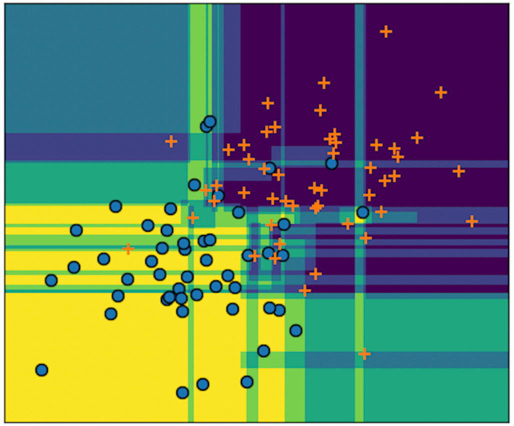
DT w/ Depth 10



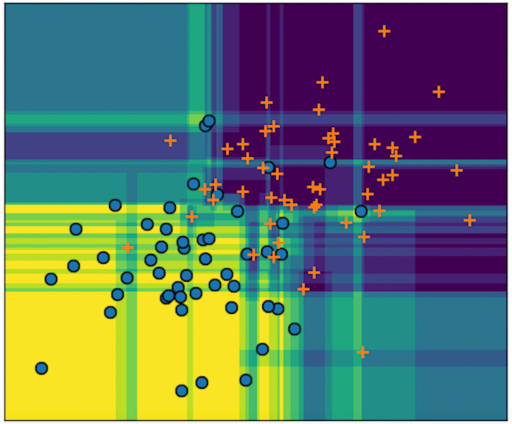
RF w/ 2 trees



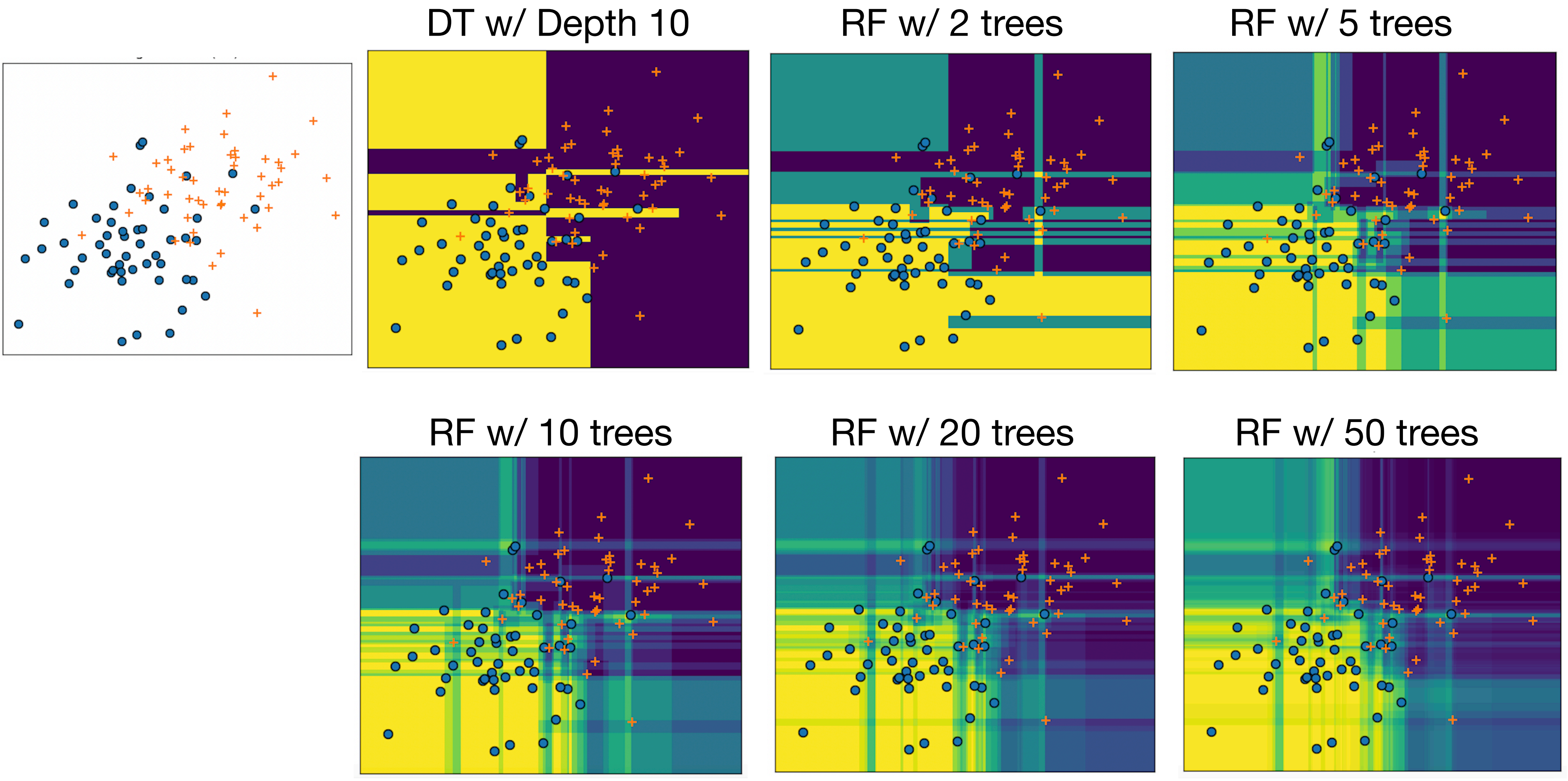
RF w/ 5 trees



RF w/ 10 trees



Demo of Random Forest



Summary for today

An approach to reduce the variance of our classifier:

Summary for today

An approach to reduce the variance of our classifier:

1. Create datasets via bootstrapping + train classifiers on them + averaging

Summary for today

An approach to reduce the variance of our classifier:

1. Create datasets via bootstrapping + train classifiers on them + averaging

2. To further reduce correlation between classifiers, RF randomly selects subset of dimensions for every split.