EECE5645
Parallel Processing for Data Analytics

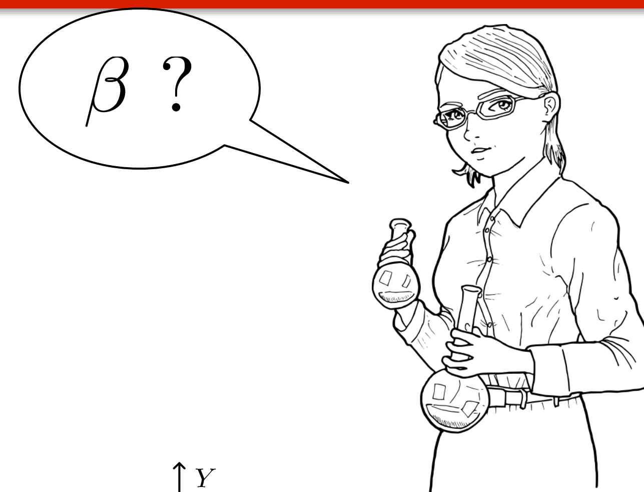
Lecture 10: Feature Selection

Linear Regression

$x_i \in \mathbb{R}^d$ $\qquad$ $y_i \in \mathbb{R}$

n

Height	Weight	Age	Blood Pressure
1.8m	40K	19ys	3.1
1.9m	55K	34ys	1.2
⋮			
1.5m	90K	24ys	2.5

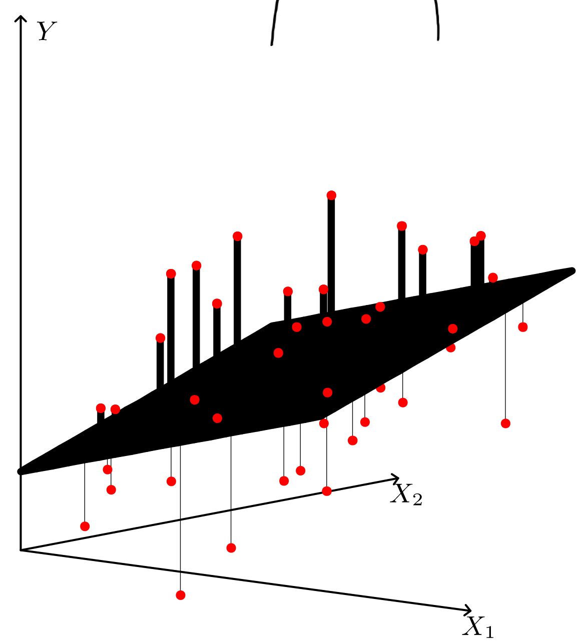


$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

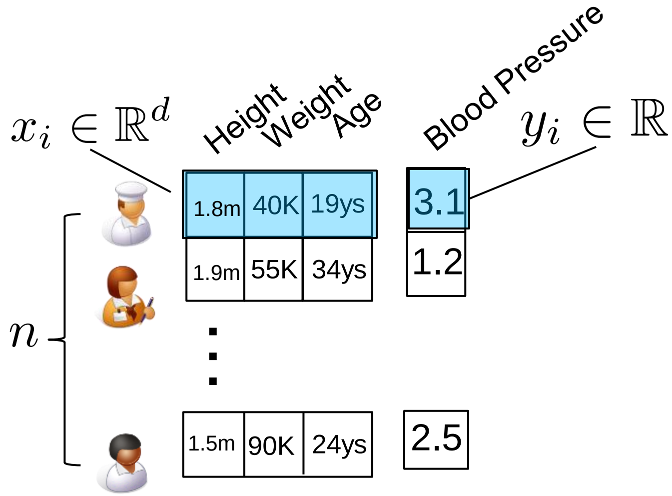
Learn by minimizing Residual-Sum-of-Squares (RSS):

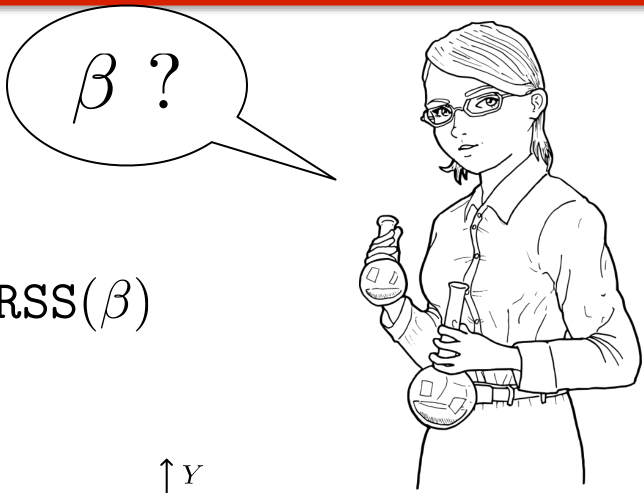
$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \mathrm{RSS}(\beta) = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \langle \beta, x_i \rangle)^2$$

$$= (X^T X)^{-1} X^T y$$

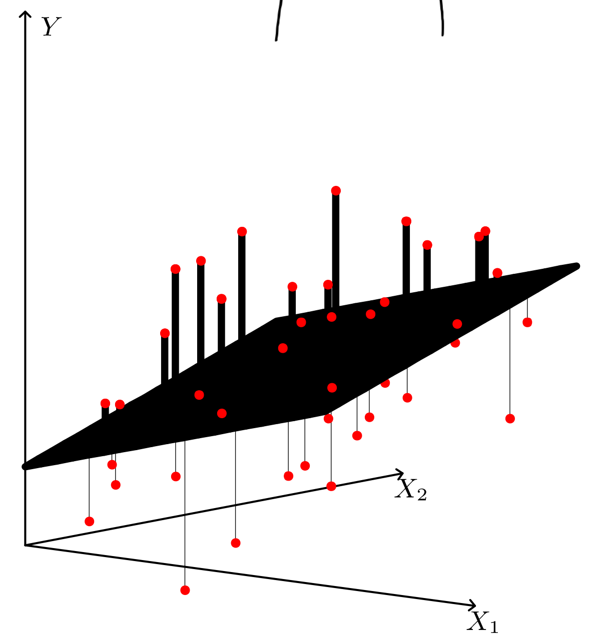


Linear Regression





$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \text{RSS}(\beta)$$

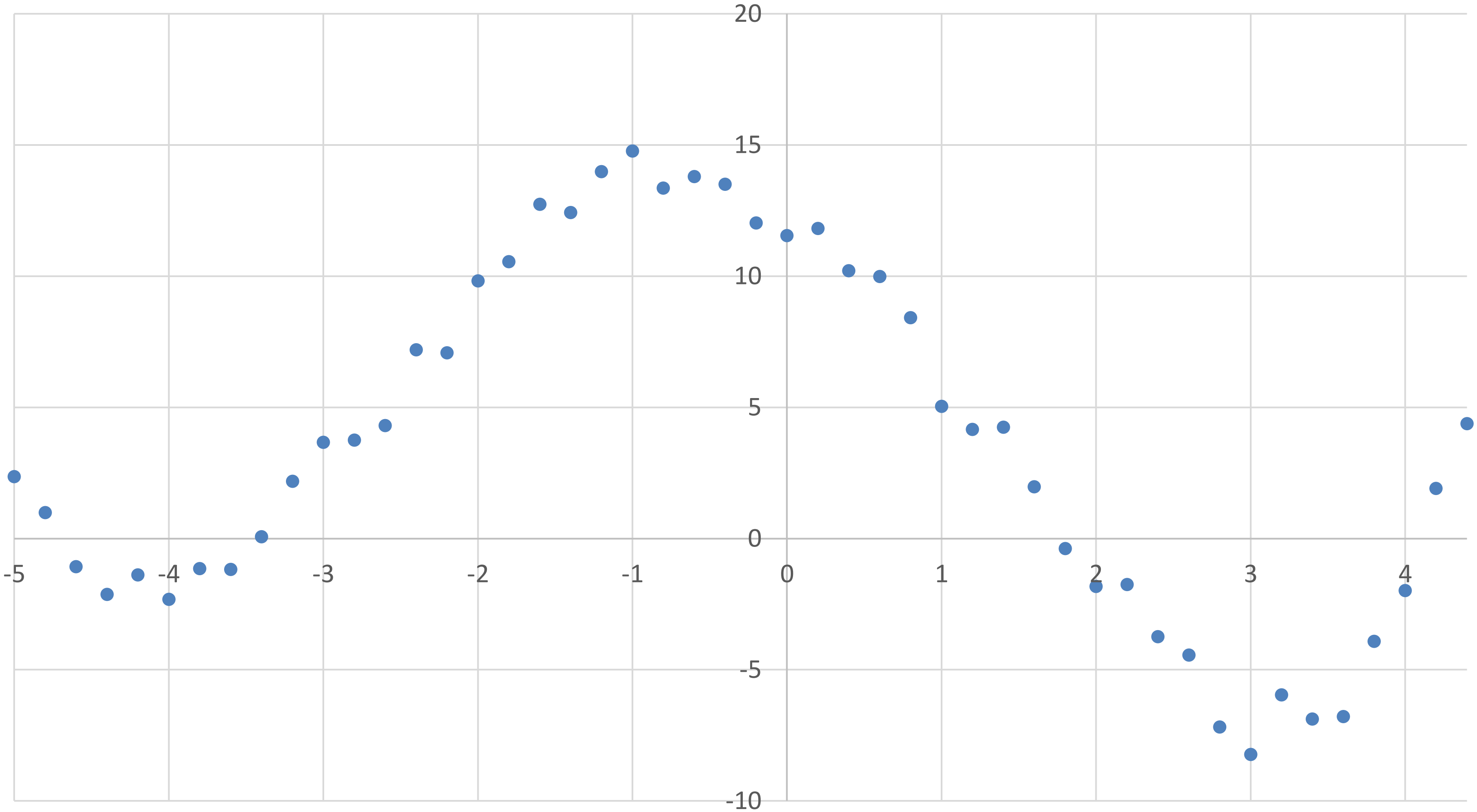


$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

Expected Prediction Error:

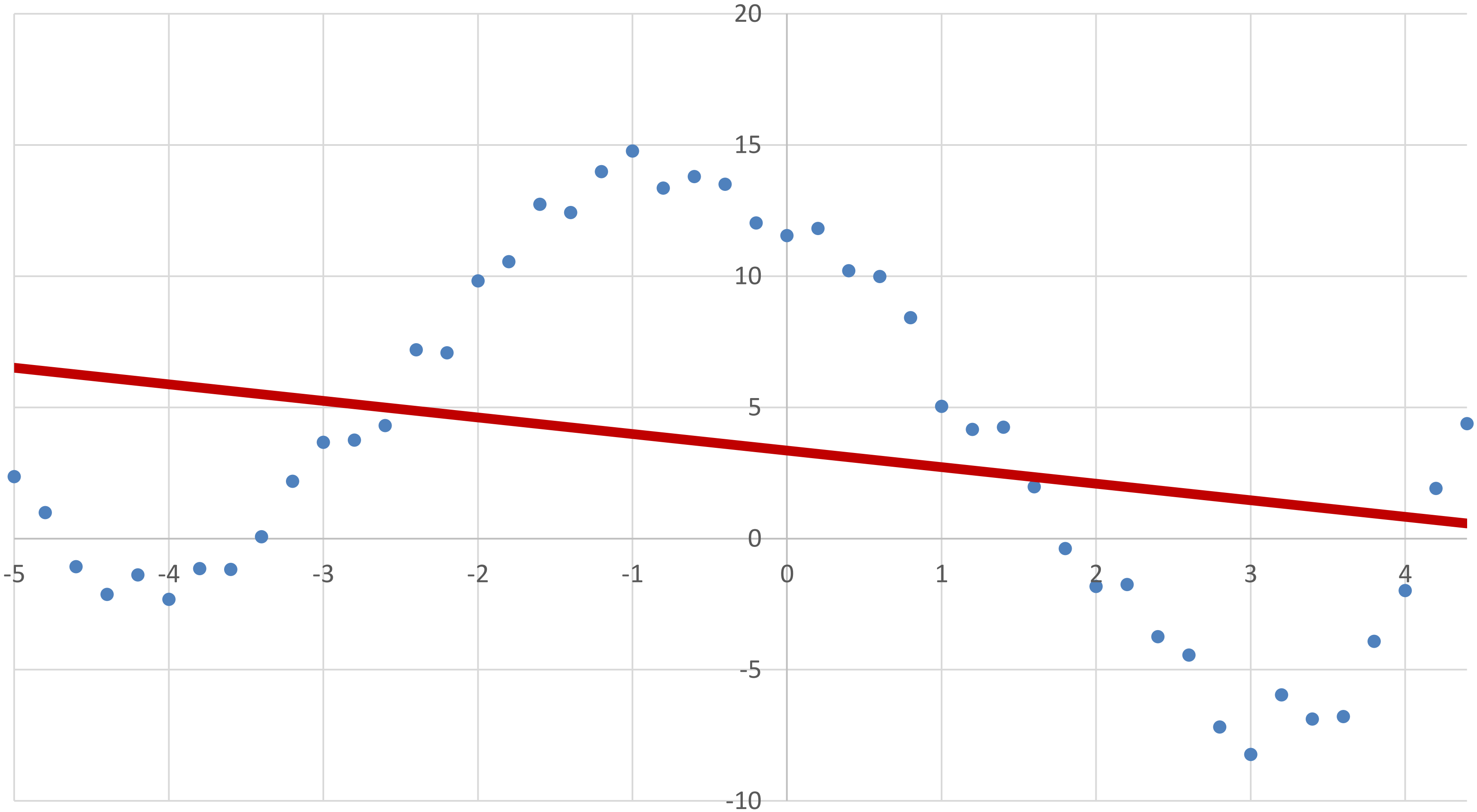
$$\text{EPE} \quad \approx \sigma^2 + \sigma^2 \frac{d}{n}$$

What if Data is Not Linear?

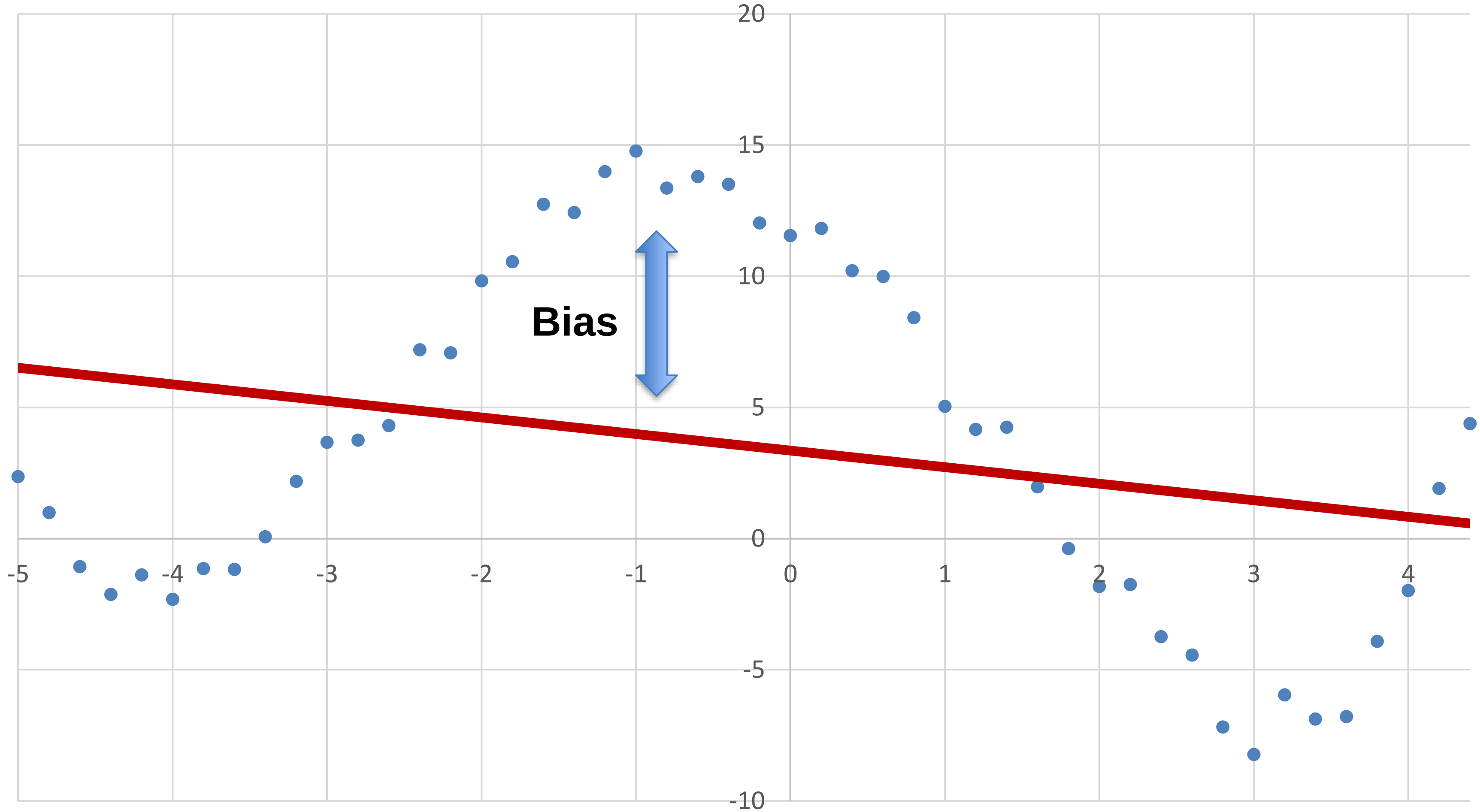


Northeastern

What if Data is Not Linear?

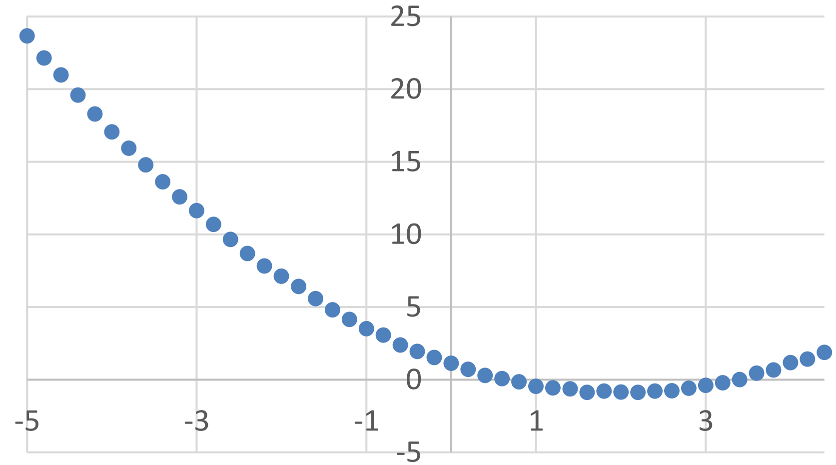


What if Data is Not Linear?



Fitting Quadratic Functions

- Suppose f is quadratic:

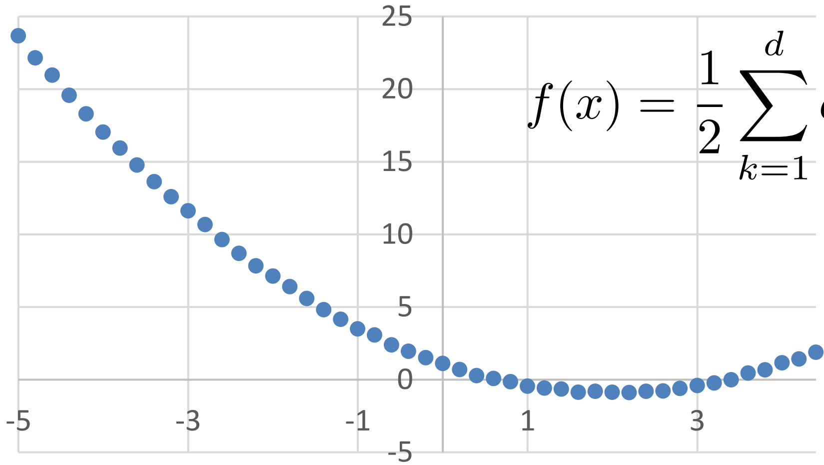


$$f(x) = \frac{1}{2}x^\top Q x + b^\top x + c$$

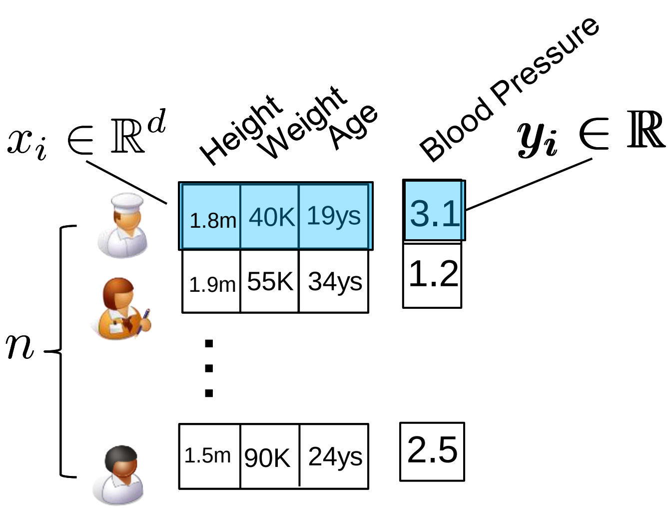
$$= \frac{1}{2}\sum_{k=1}^{d}\sum_{k'=1}^{d} q_{kk'}x_k x_{k'} + \sum_{k=1}^{d} b_k x_k + c$$

$$= \frac{1}{2}\sum_{k=1}^{d} q_{kk}x_k^2 + \sum_{k=1}^{d}\sum_{k'>k}^{d} q_{kk'}x_k x_{k'} + \sum_{k=1}^{d} b_k x_k + c$$

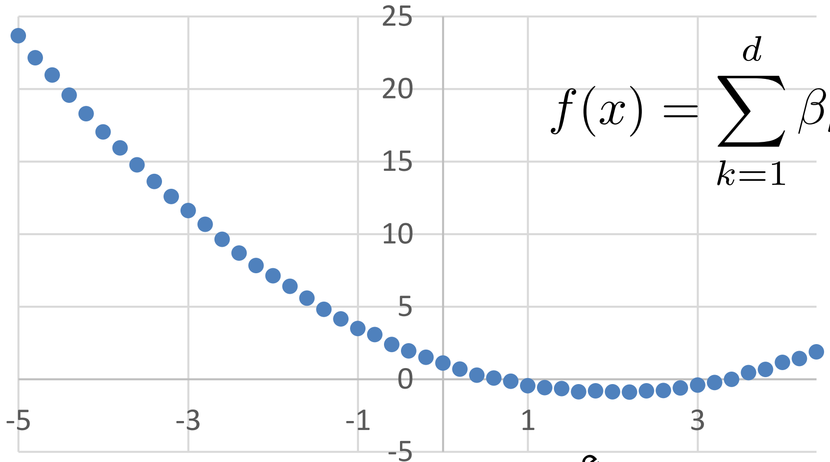
Fitting Quadratic Functions



$$f(x) = \frac{1}{2}\sum_{k=1}^{d} q_{kk} x_k^2 + \sum_{k=1}^{d}\sum_{k'>k}^{d} q_{kk'} x_k x_{k'} + \sum_{k=1}^{d} b_k x_k + c$$

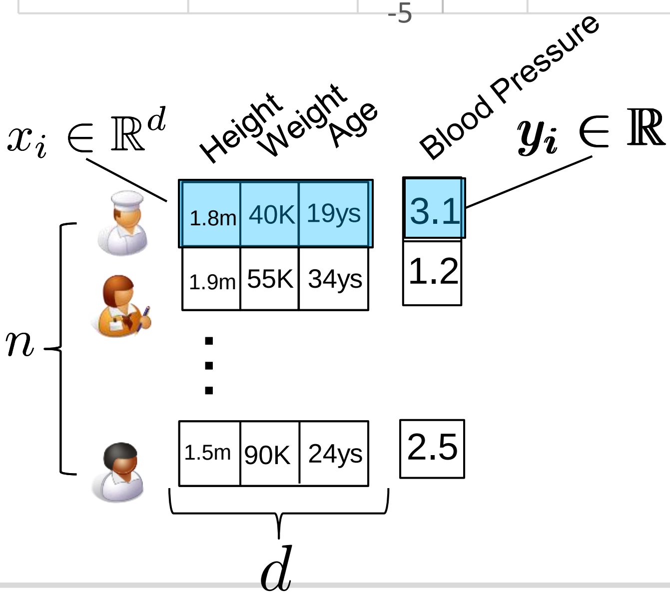


Fitting Quadratic Functions

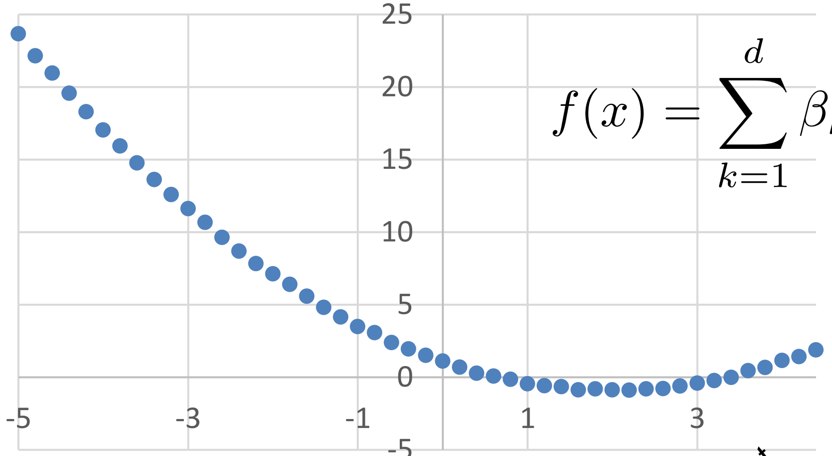


$$f(x) = \sum_{k=1}^{d} \beta_{kk} x_k^2 + \sum_{k=1}^{d} \sum_{k'>k}^{d} \beta_{kk'} x_k x_{k'} + \sum_{k=1}^{d} \beta_k x_k + \beta_0$$

$$\beta \in \mathbb{R}^{\frac{d(d-1)}{2}+2d+1}$$

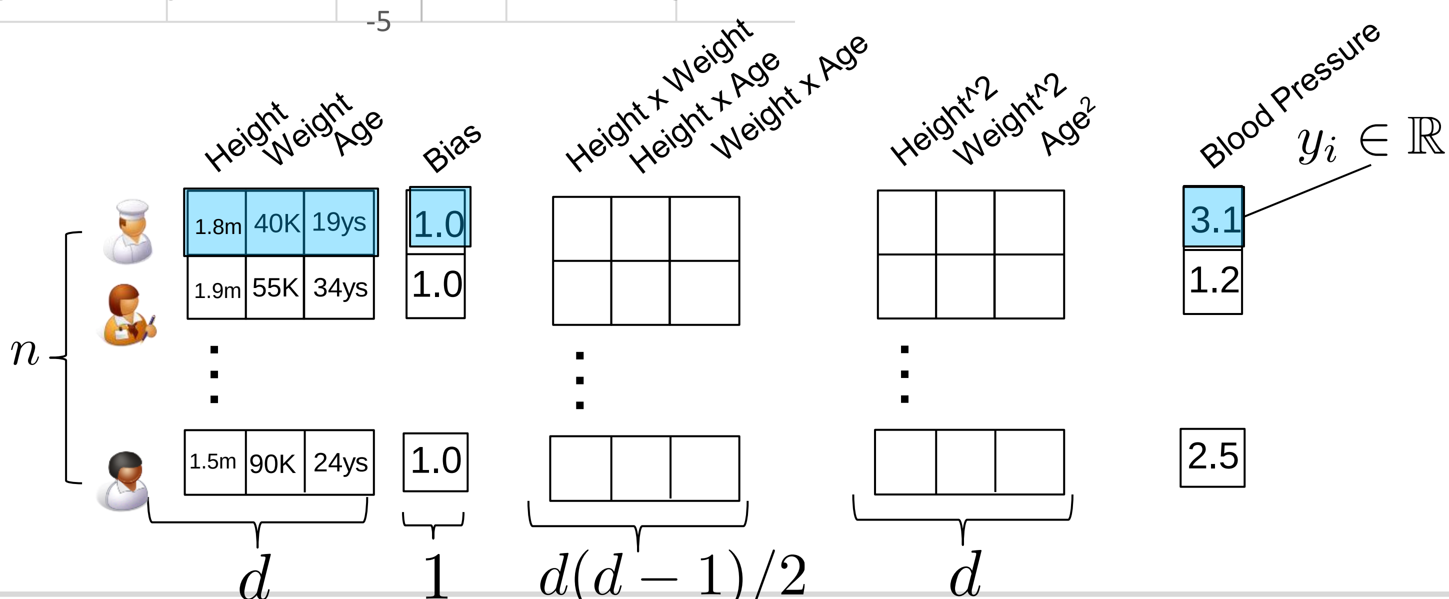


Fitting Quadratic Functions



$$f(x) = \sum_{k=1}^{d} \beta_{kk} x_k^2 + \sum_{k=1}^{d} \sum_{k'>k}^{d} \beta_{kk'} x_k x_{k'} + \sum_{k=1}^{d} \beta_k x_k + \beta_0$$

Learn $\beta \in \mathbb{R}^{\frac{d(d-1)}{2}+2d+1}$ **through LSE!**

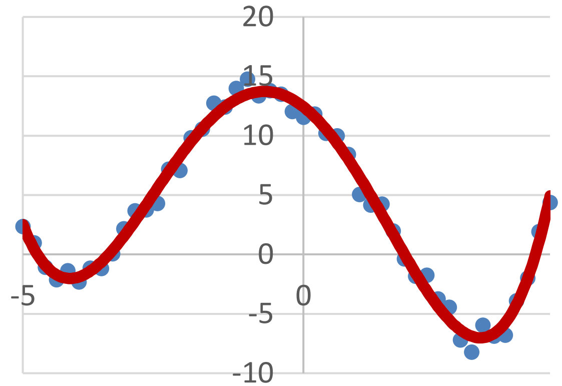


Northeastern

Fitting Polynomials

To learn a polynomial f of degree

$$k = 2, 3, \ldots :$$



- Produce new features containing all monomials:

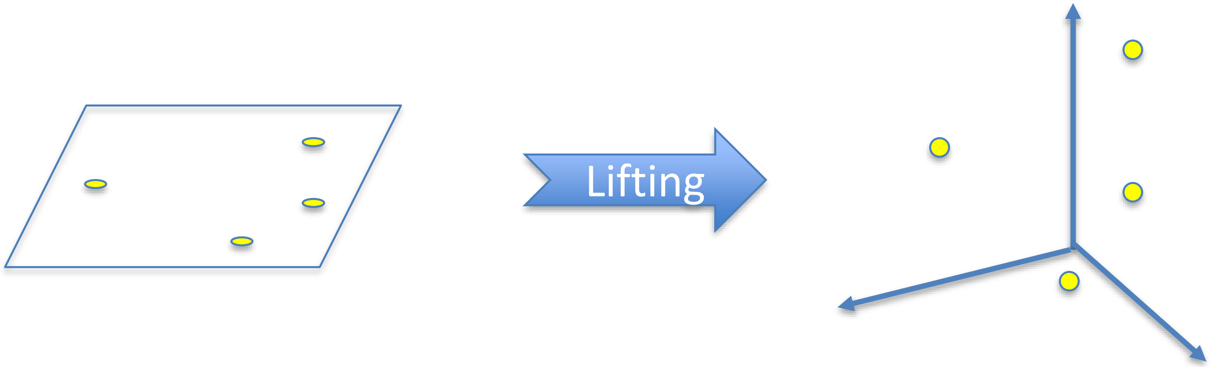
$$\prod_{i=1}^{d} x_i^{k_i} = x_1^{k_1} x_2^{k_2} \ldots x_d^{k_d}$$

where $k_1 + k_2 + \ldots + k_d \leq k$.

- Perform linear regression on resulting new set of features

Lifting

- Affine in $\mathbb{R}^d$ = Linear in $\mathbb{R}^{d+1}$
- Quadratic in $\mathbb{R}^d$ = Linear in $\mathbb{R}^{d(d-1)/2+2d+1} = \mathbb{R}^{O(d^2)}$
- Polynomial of degree k in $\mathbb{R}^d$ = Linear in $\mathbb{R}^{O(d^k)}$



Northeastern

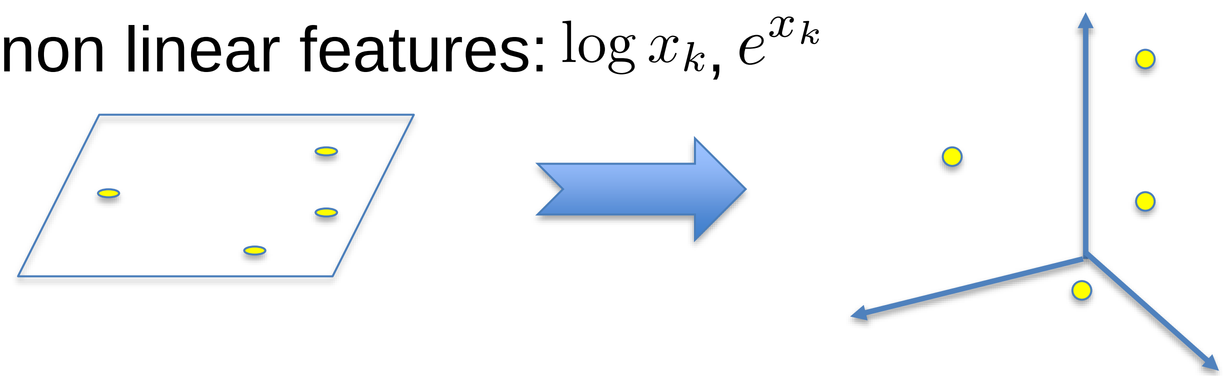
Different Basis Functions

Coefficients to be learned

Known **basis** functions

$$f(x) = \sum_{\ell=1}^{m} \beta_\ell f_\ell(x)$$

- Polynomials: basis functions are monomials
- Periodic functions: $\sin(\ell \frac{x}{T}), \cos(\ell \frac{x}{T}), \ell \in \mathbb{N}, x \in [0, T]$
- Other non linear features: $\log x_k$, e^{x_k}



$(x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$ $\qquad$ $(f_1(x), f_2(x), \ldots, f_m(x)) \in \mathbb{R}^m$

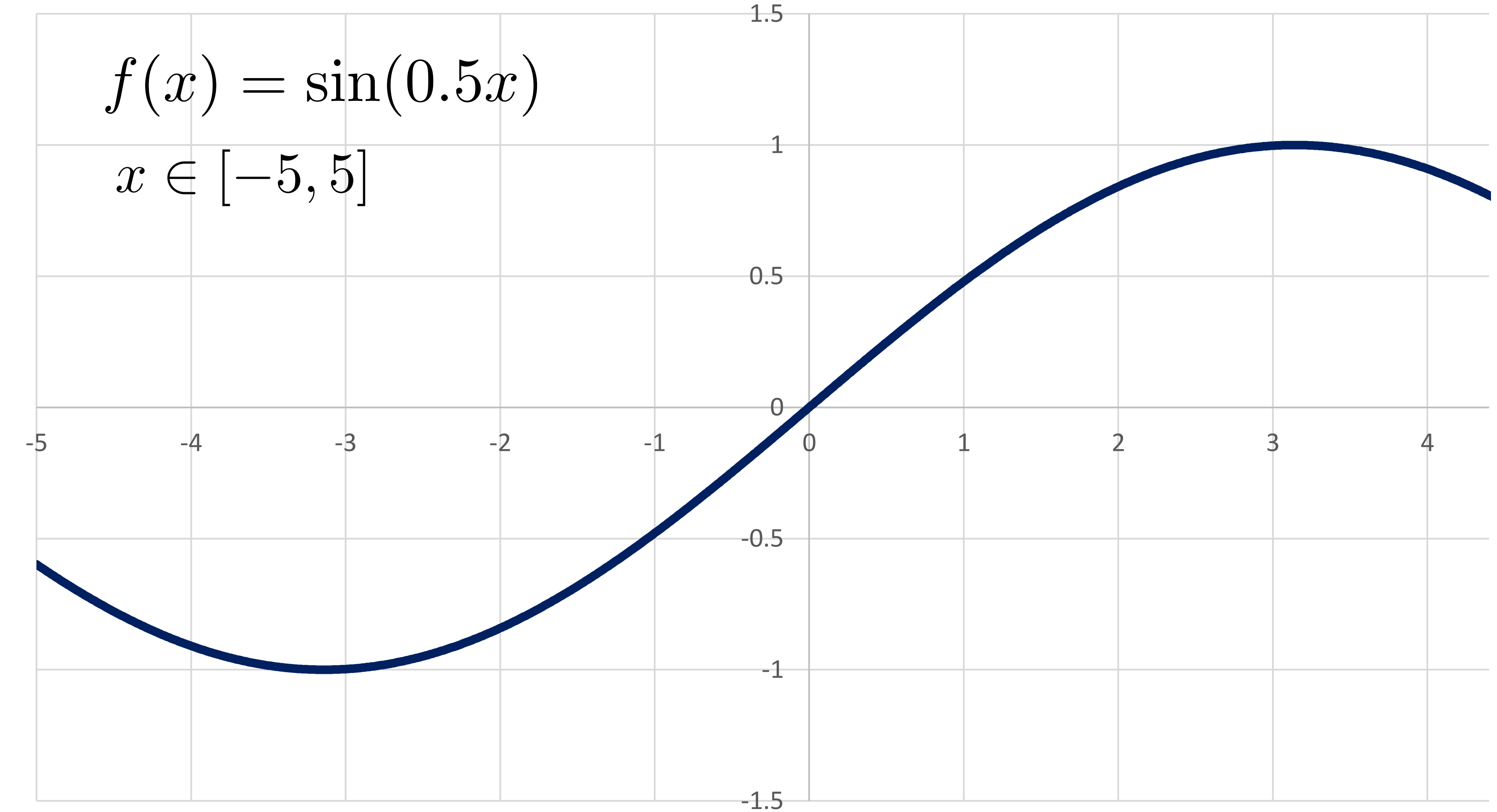
Stone-Weierstrass Theorem

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a continuous function defined over a closed and bounded set $A \subset \mathbb{R}^d$. Then, for any $\delta > 0$, there exists a polynomial $p : \mathbb{R}^d \to \mathbb{R}$ such that:

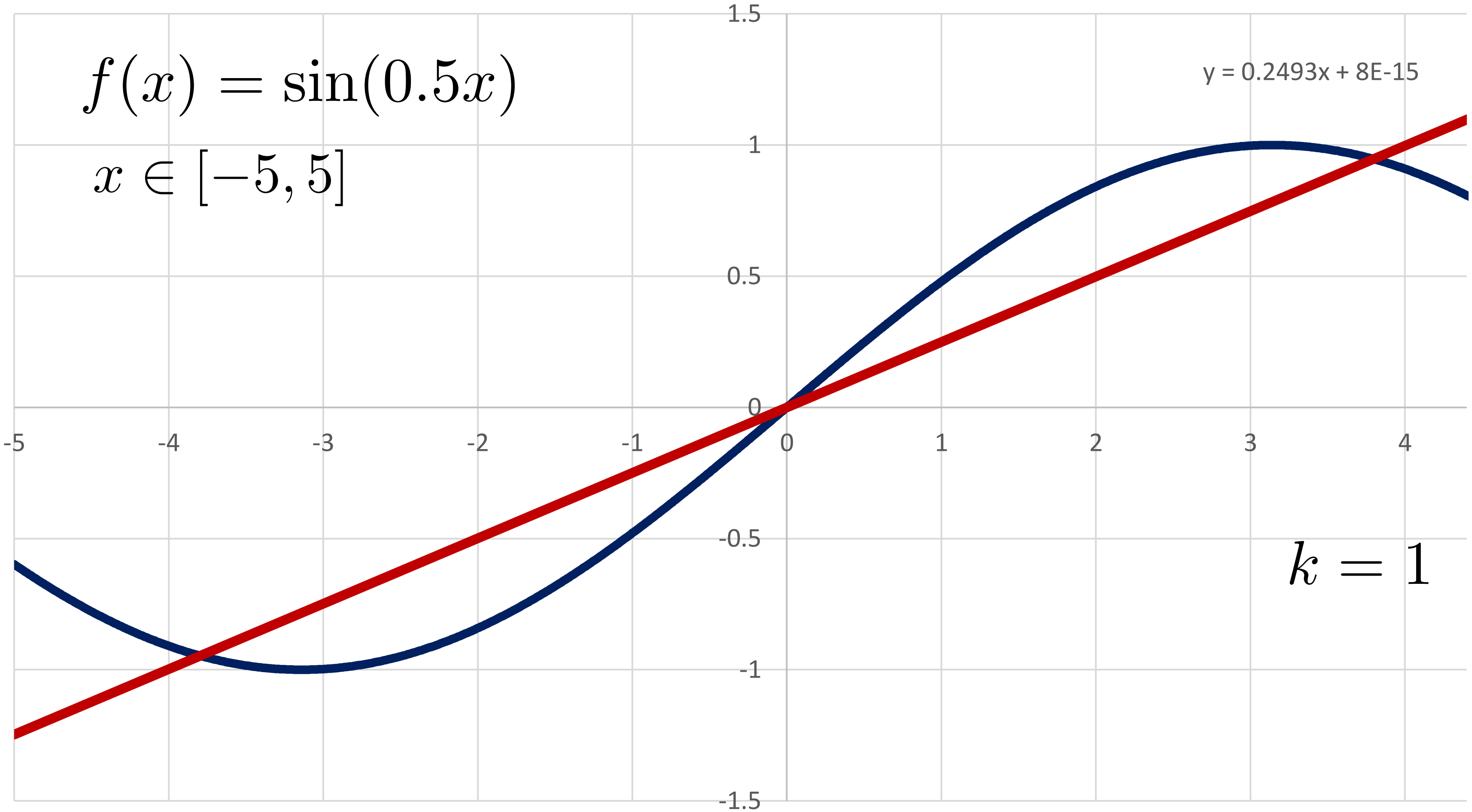
$$|f(x) - p(x)| \leq \delta$$

for all $x \in A$.

Stone-Weierstrass in Action

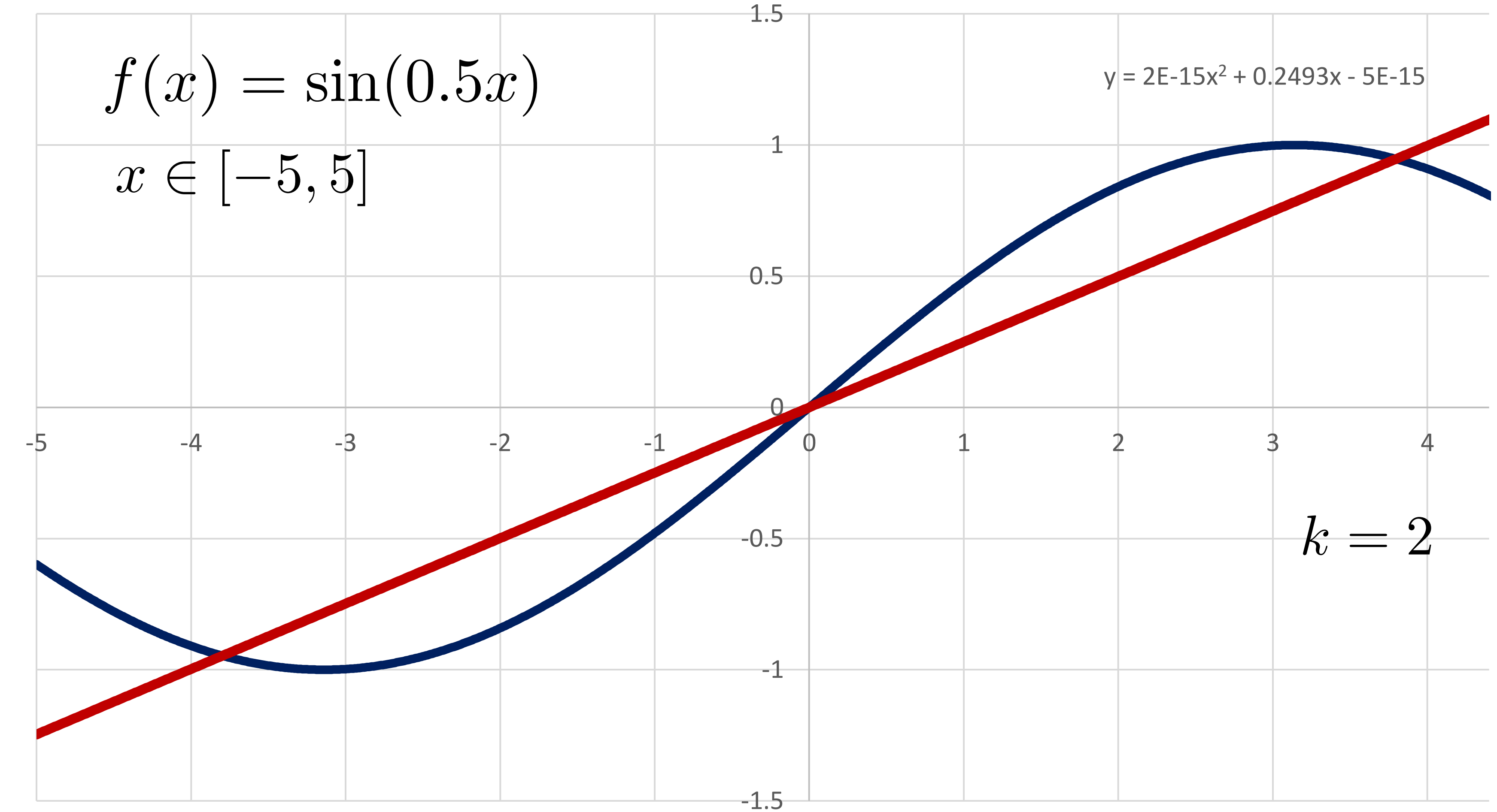


Stone-Weierstrass in Action

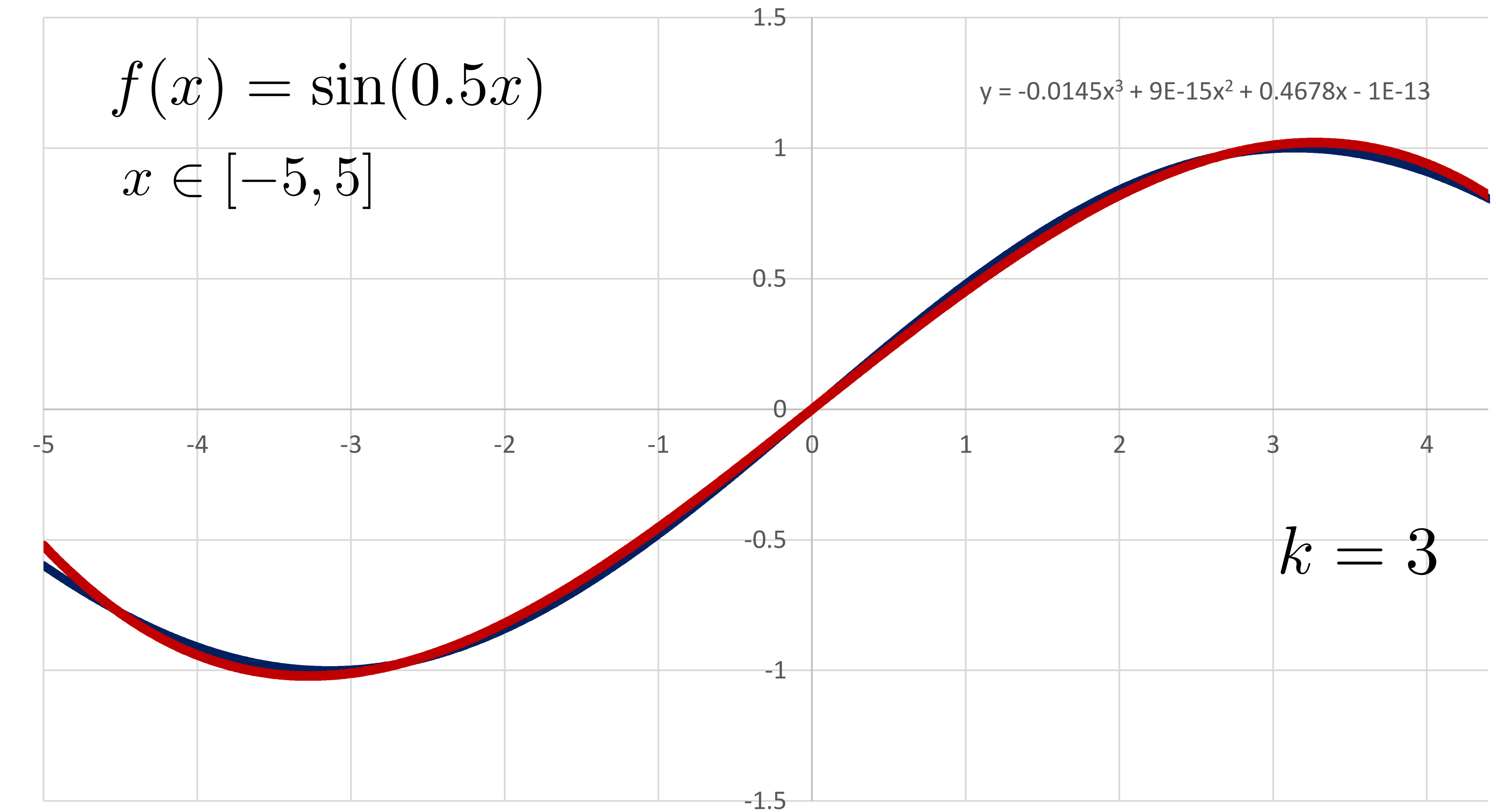


Northeastern

Stone-Weierstrass in Action

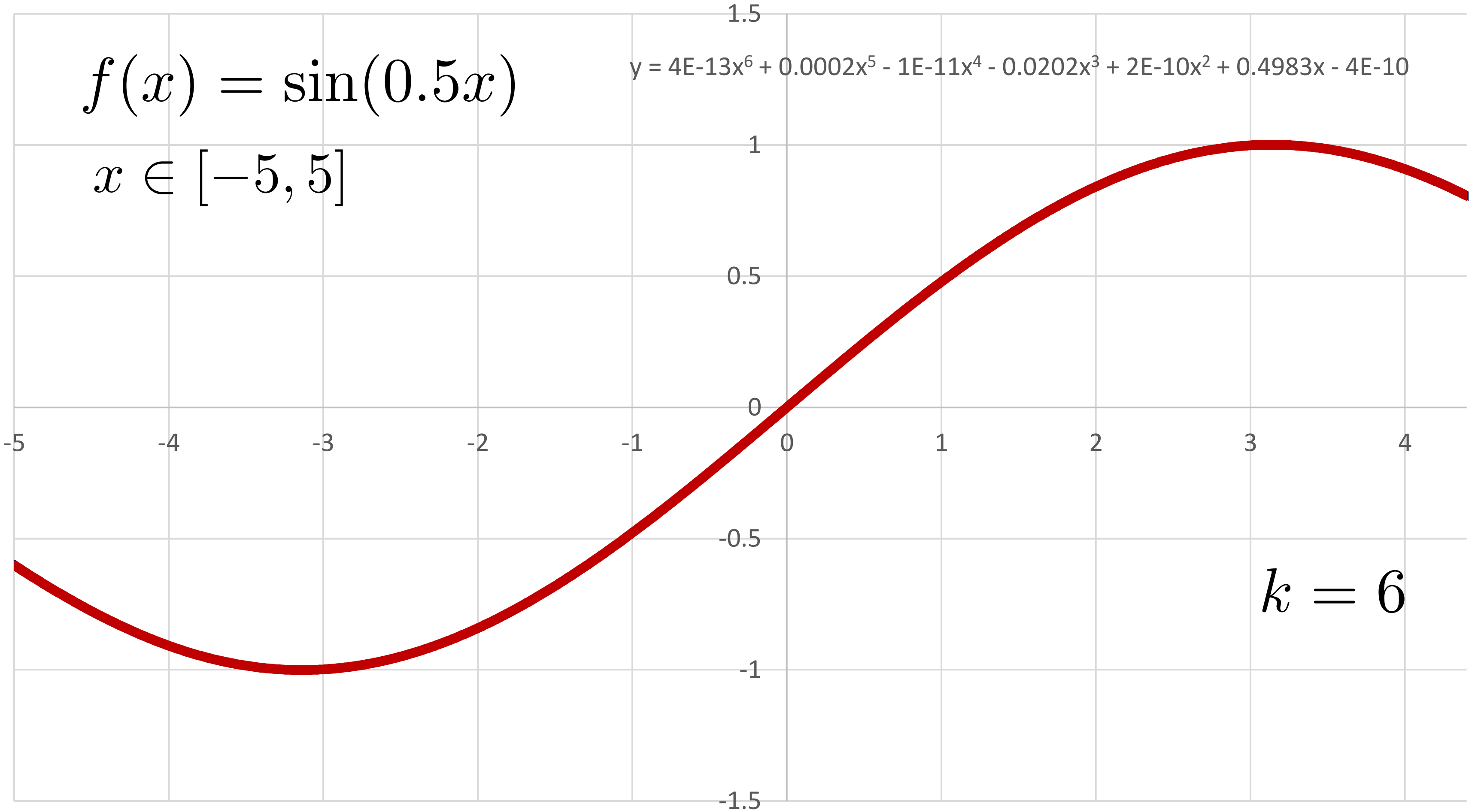


Stone-Weierstrass in Action

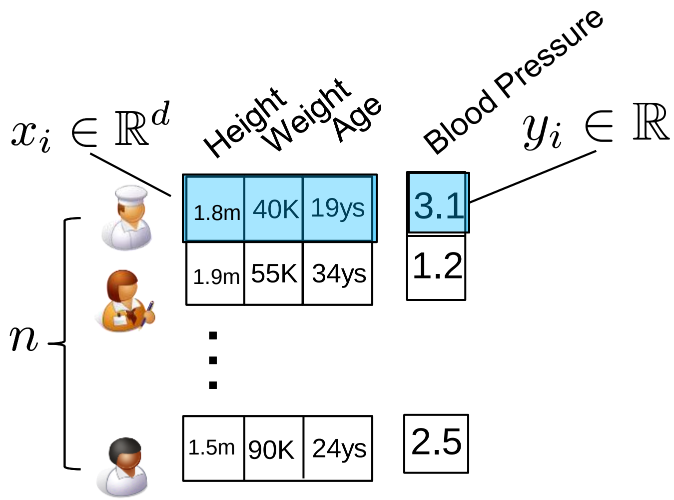


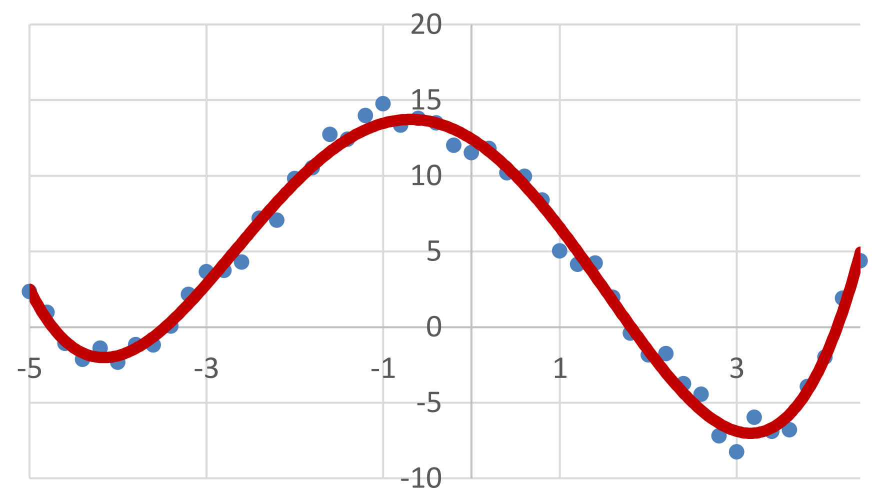
Northeastern

Stone-Weierstrass in Action



What Does This Imply?





$$y_i = f(x_i) + \varepsilon_i, \quad i = 1, \ldots, n$$

Suppose features x_i are in in $[0, 100]^d$, and $f : \mathbb{R}^d \to \mathbb{R}$ is a continuous function.

Then, we can learn a polynomial that is **arbitrarily close to** f using linear regression!

Wait…what?

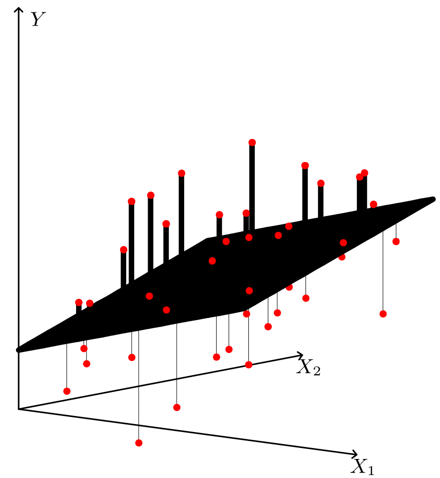
kNN

- $f : \mathbb{R}^d \to \mathbb{R}$ is continuous

Curse of dimensionality

Linear Regression

- $f : \mathbb{R}^d \to \mathbb{R}$ is linear.



No curse: we have assumed it away!

Wait…what?

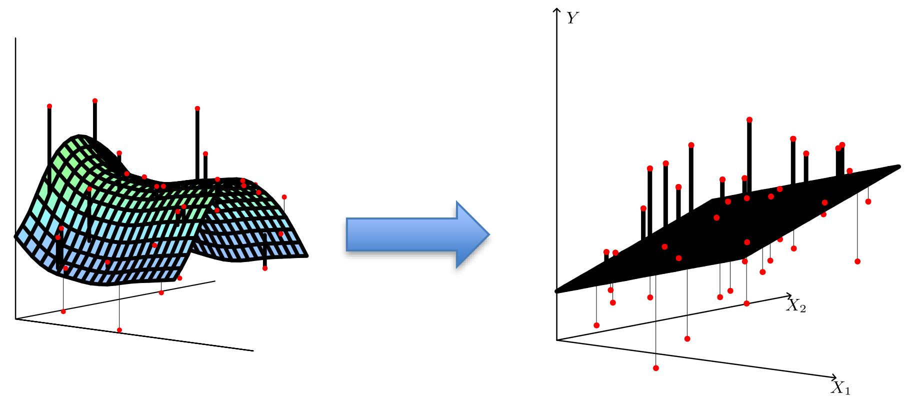
kNN

- ❑ $f : \mathbb{R}^d \to \mathbb{R}$ is continuous

Curse of dimensionality

Linear Regression

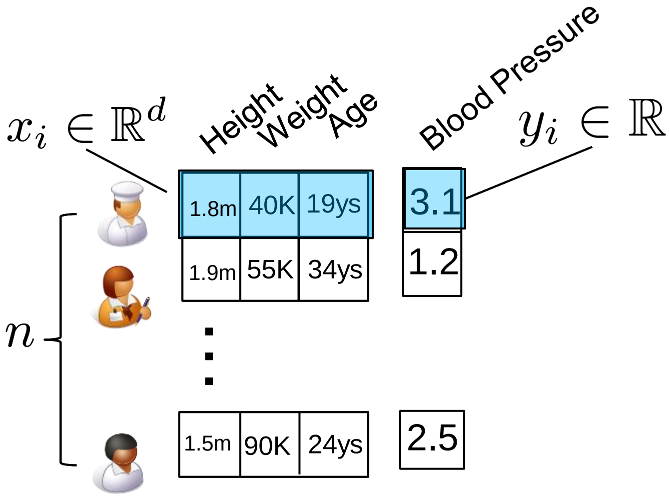
- ❑ $f : \mathbb{R}^d \to \mathbb{R}$ is continuous



Learn f through appropriate lifting.

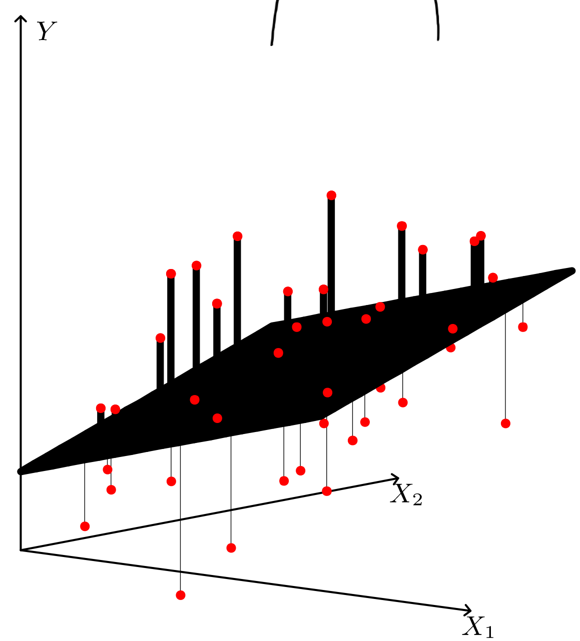
Did we escape the curse?!?

No, We Did Not.

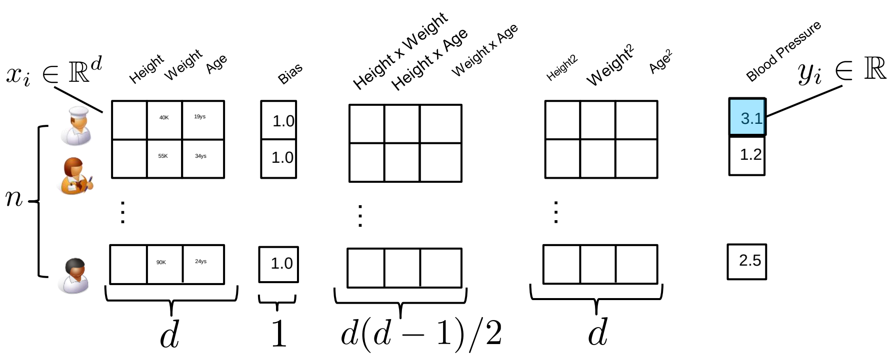


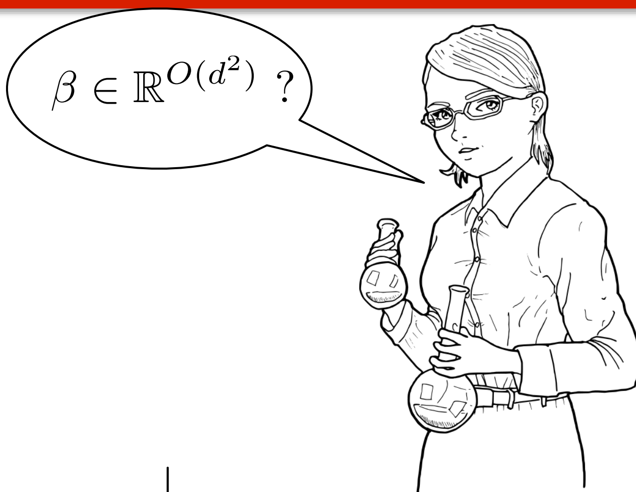
$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

$$\boxed{n \geq \frac{d\sigma^2}{\epsilon}}$$

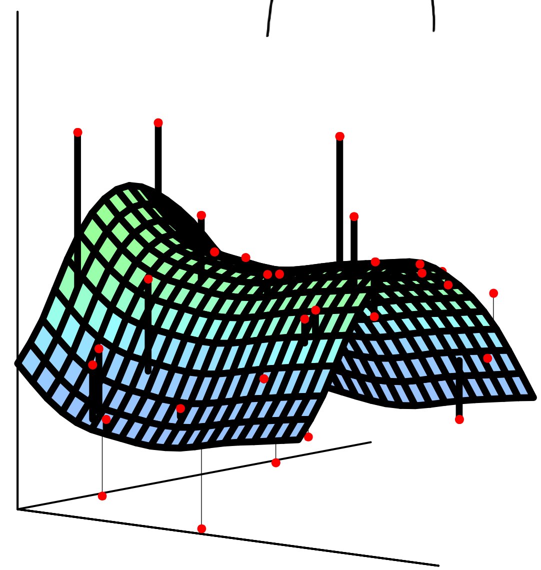


No, We Did Not.



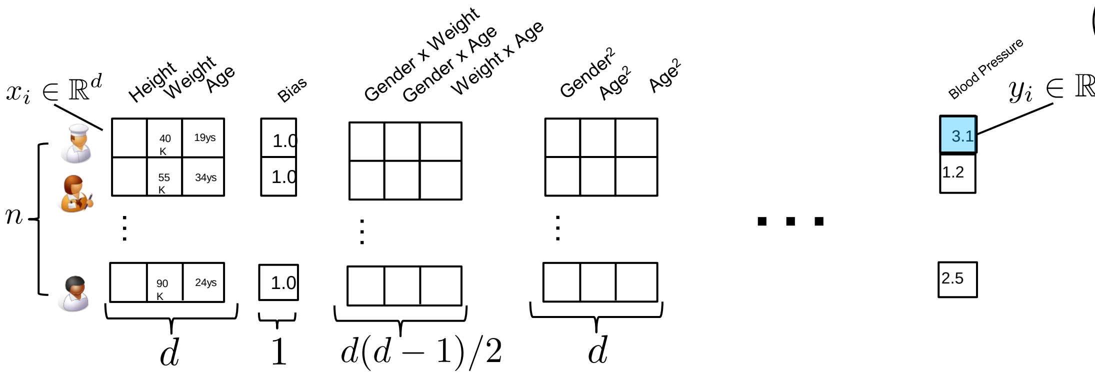


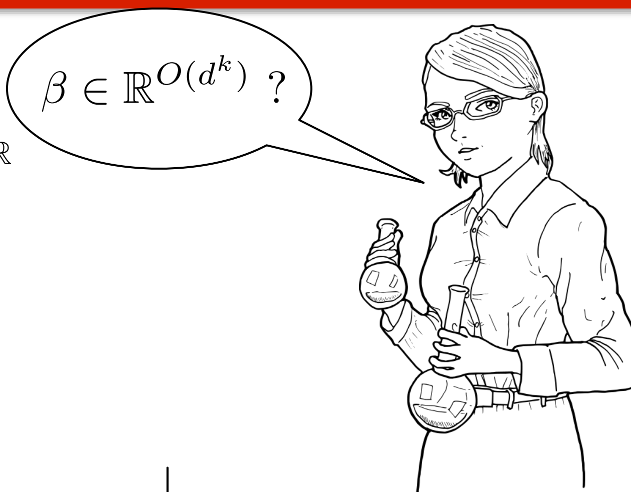
$$f(x) = \sum_{k=1}^{d} \beta_{kk} x_k^2 + \sum_{k=1}^{d} \sum_{k'>k}^{d} \beta_{kk'} x_k x_{k'} + \sum_{k=1}^{d} \beta_k x_k + \beta_0$$



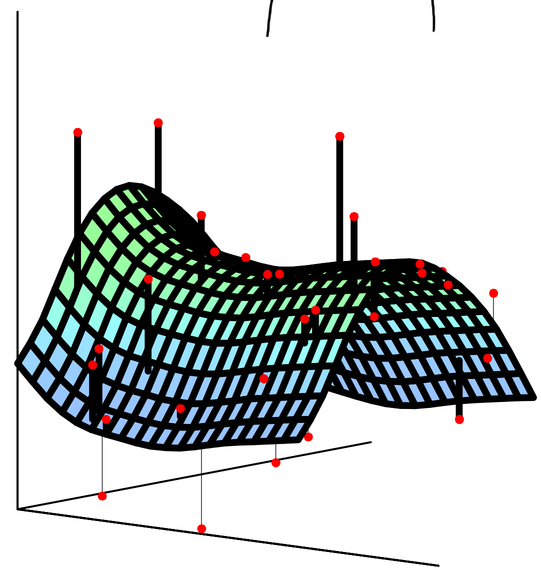
$$n \geq \frac{d^2 \sigma^2}{\epsilon}$$

No, We Did Not.



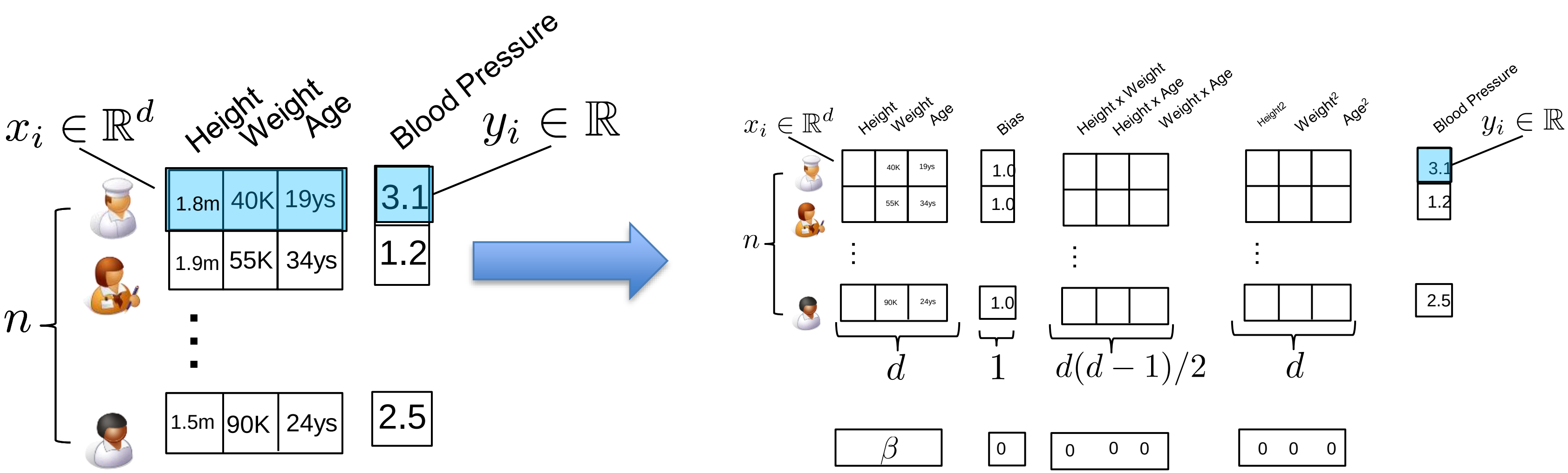


$$f(x) = \sum_{k_1,k_2,\ldots,k_n : \sum_{i=1}^n k_i \le k} \beta_{k_1,k_2,\ldots,k_n} \prod_{i=1}^n x_i^{k_i}.$$



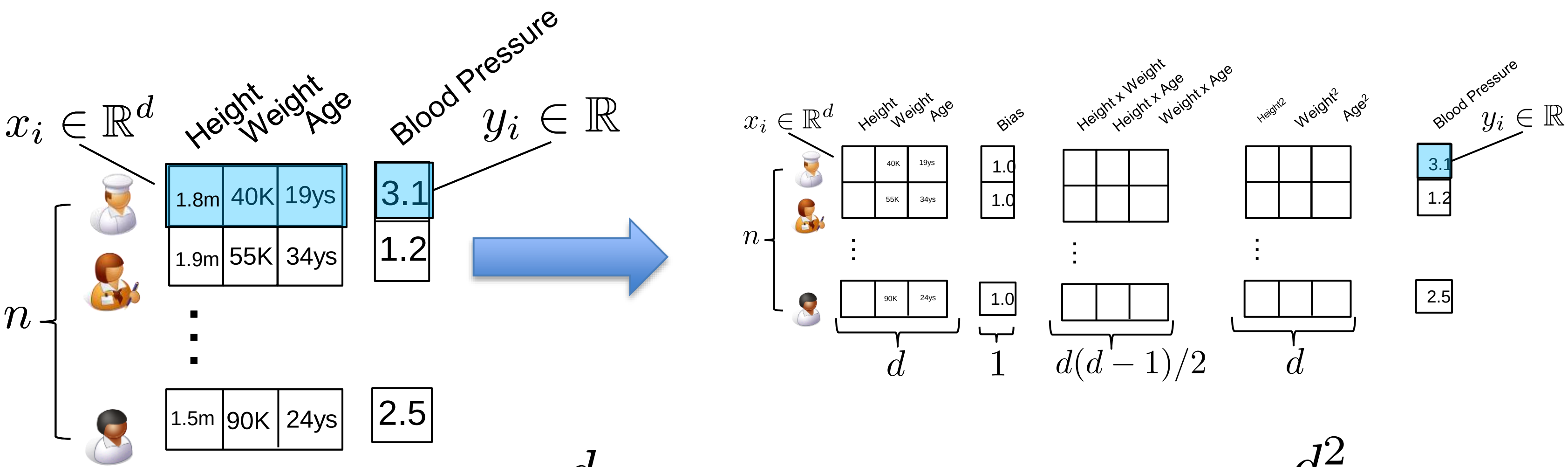
$$n \geq \frac{d^k \sigma^2}{\epsilon}$$

To Lift or Not to Lift?



- If f is **quadratic**, we want to lift
- If f is **linear** and then LSE should still learn a linear model

Lifting Can Lead to Redundant Features

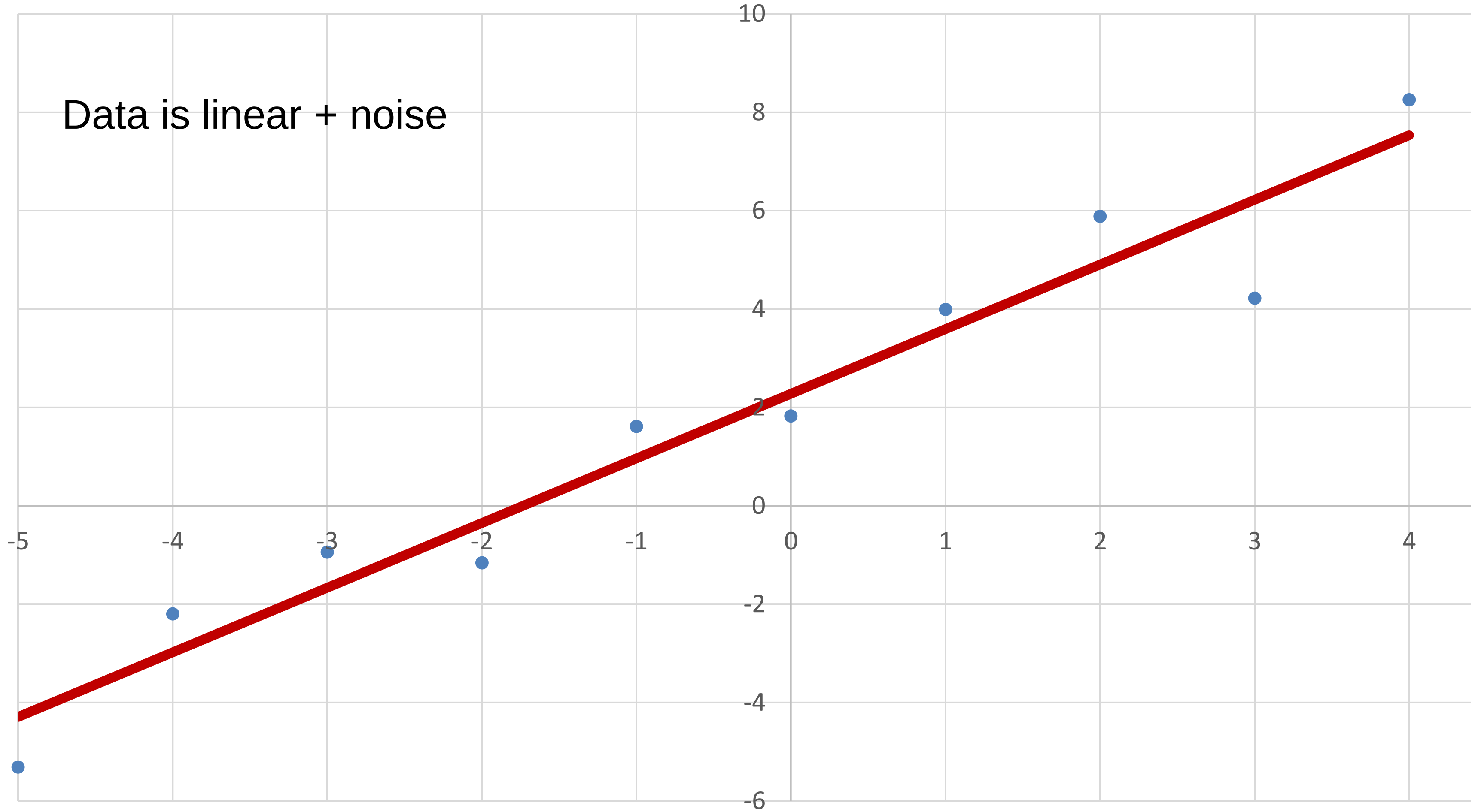


$$\mathbb{E}[\text{EPE}] \approx \sigma^2 + \sigma^2 \frac{d}{n}$$

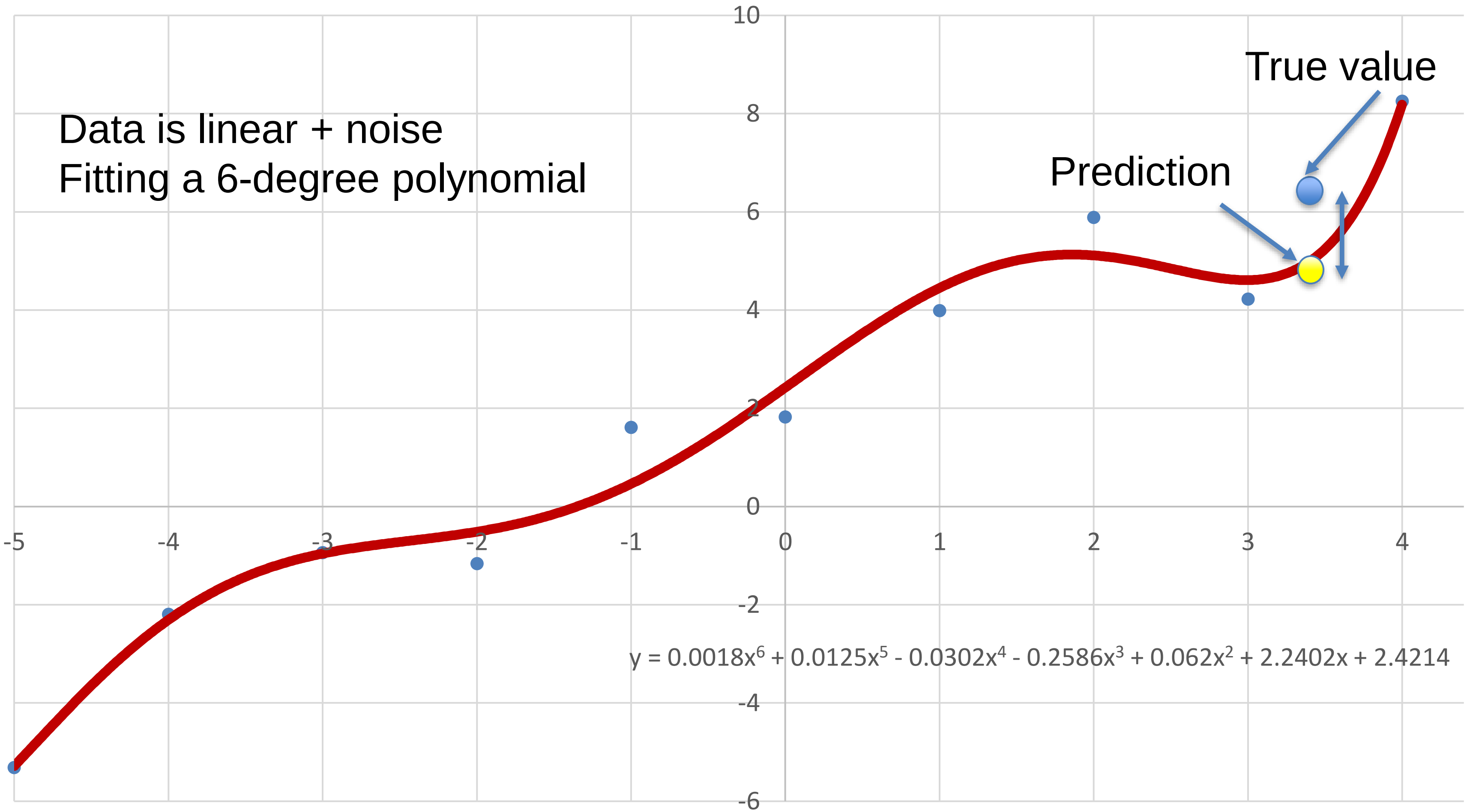
$$\mathbb{E}[\text{EPE}] \approx \sigma^2 + \sigma^2 \frac{d^2}{n}$$

- If f is linear, quadratic features are **redundant/irrelevant**
- Lifting will make EPE increase compared to not lifting!

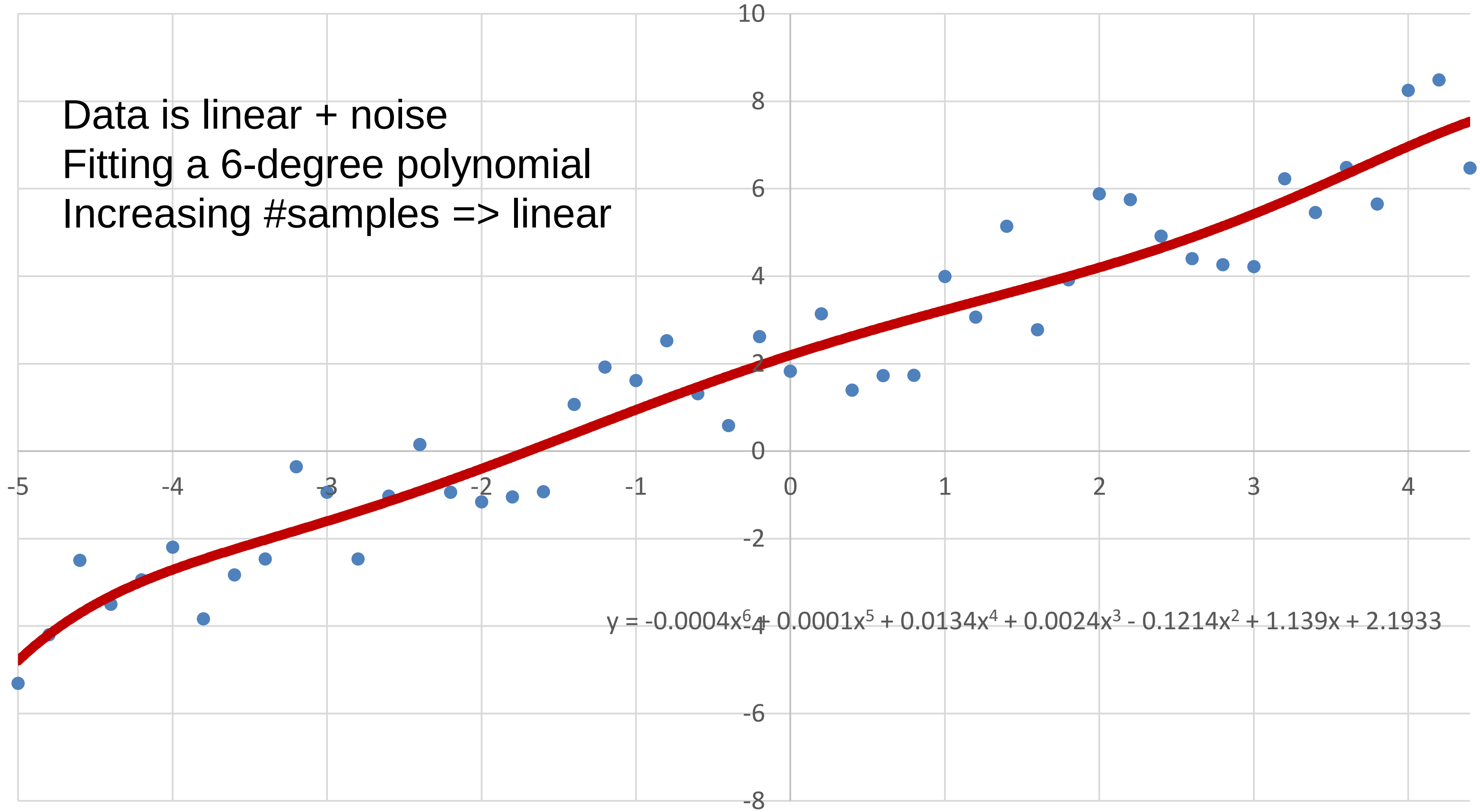
Over-fitting



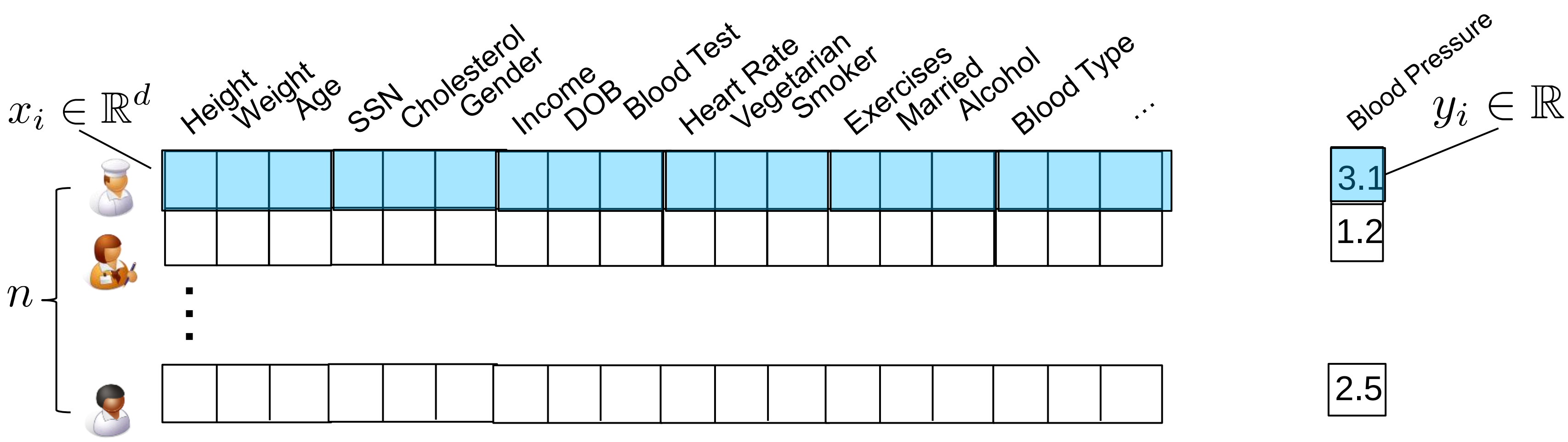
Over-fitting



Over-fitting



Redundant Features May Exist in Data!



$$\beta = \begin{bmatrix} 0 & 0.2 & 0 & 0 & 0 & 0 & -0.3 & 0 & 0 & 0 & 1.4 & 0 & 0 & 2.1 & 0 & 0 & 0 & -2.7 \end{bmatrix}$$

Only $d' \ll d$ features actually affect blood pressure!

Linear Regression needs:

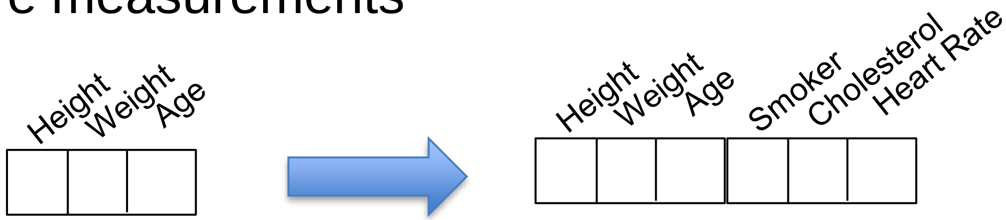
$$n = O(d) \gg O(d')$$

to learn β

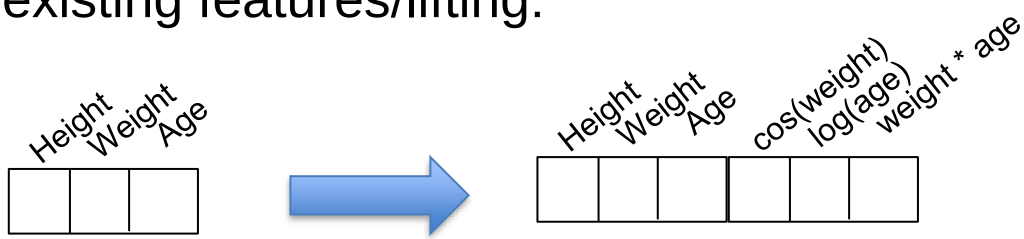
Summary

One can increase number of features by:

- ❑ Collecting more measurements



- ❑ Transforming existing features/lifting:



- ❑ If features are redundant, regression will set corresponding weight to zero, but…

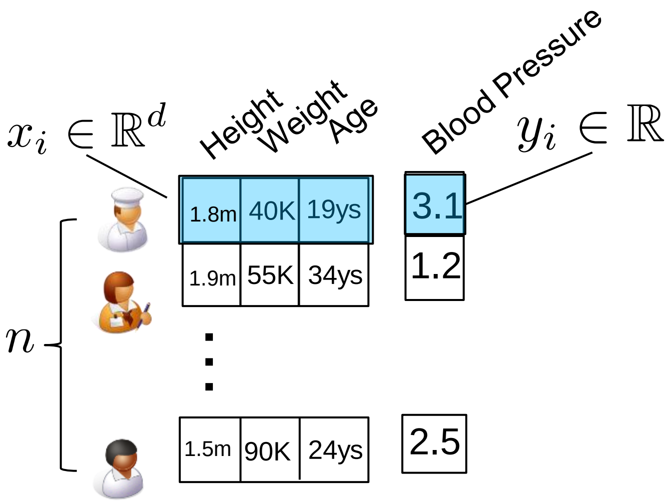
- ❑ ..it will require more samples to do this!!!

A New Challenge

- In practice, we are often just given a dataset

- We cannot increase n: we need to work with what we have

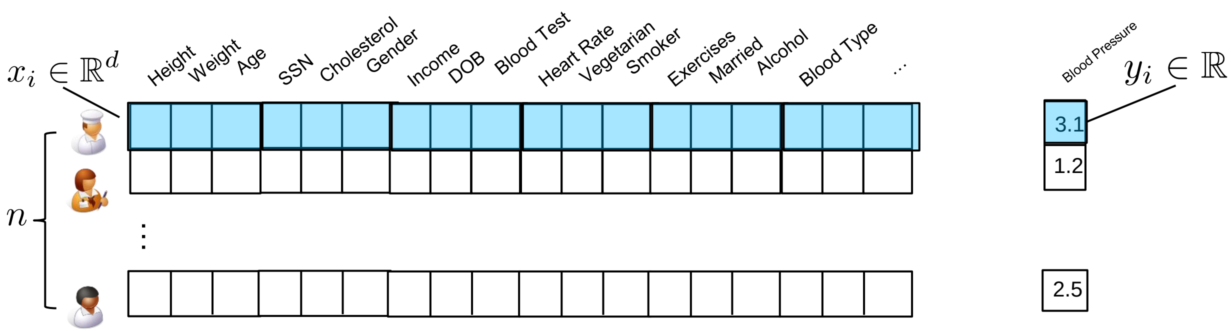
A New Challenge

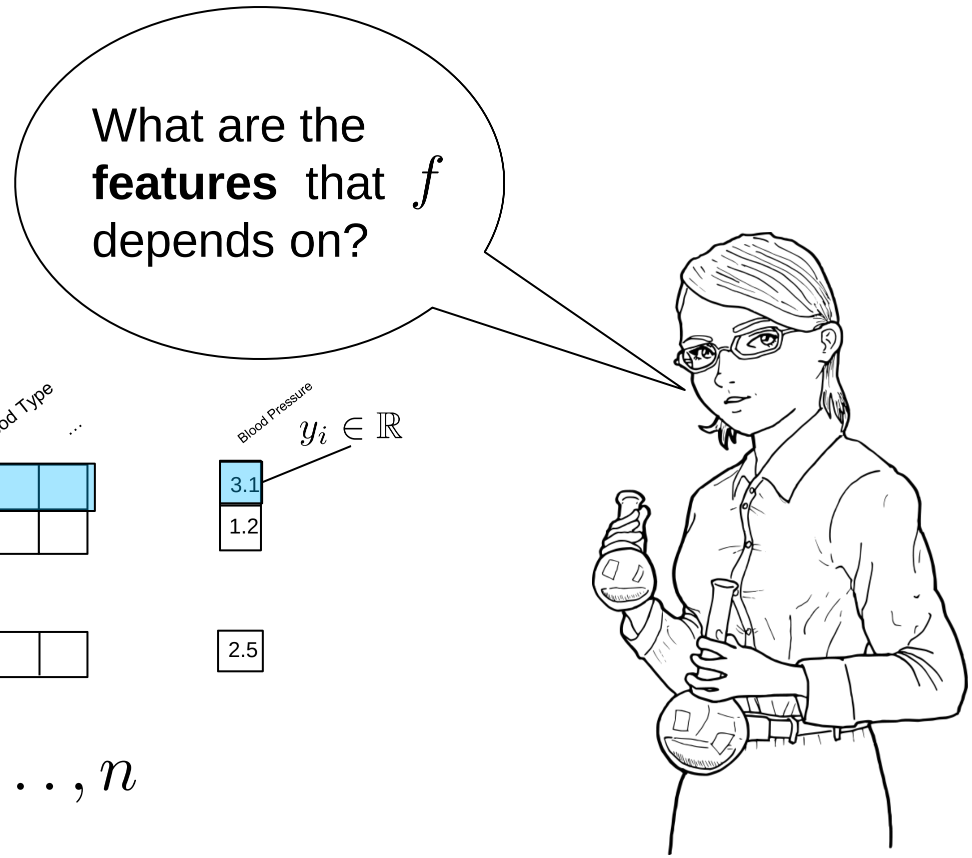


$$y_i \approx f(x_i), \quad i = 1, \dots, n$$

Northeastern

A New Challenge





$$y_i \approx f(x_i), \quad i = 1, \ldots, n$$

Feature Selection!!!

Northeastern

Q: How can we find out which features matter?

0	0.2	0	0	0	0	-0.3	0	0	0	1.4	0	0	2.1	0	0	0	-2.7	⋰

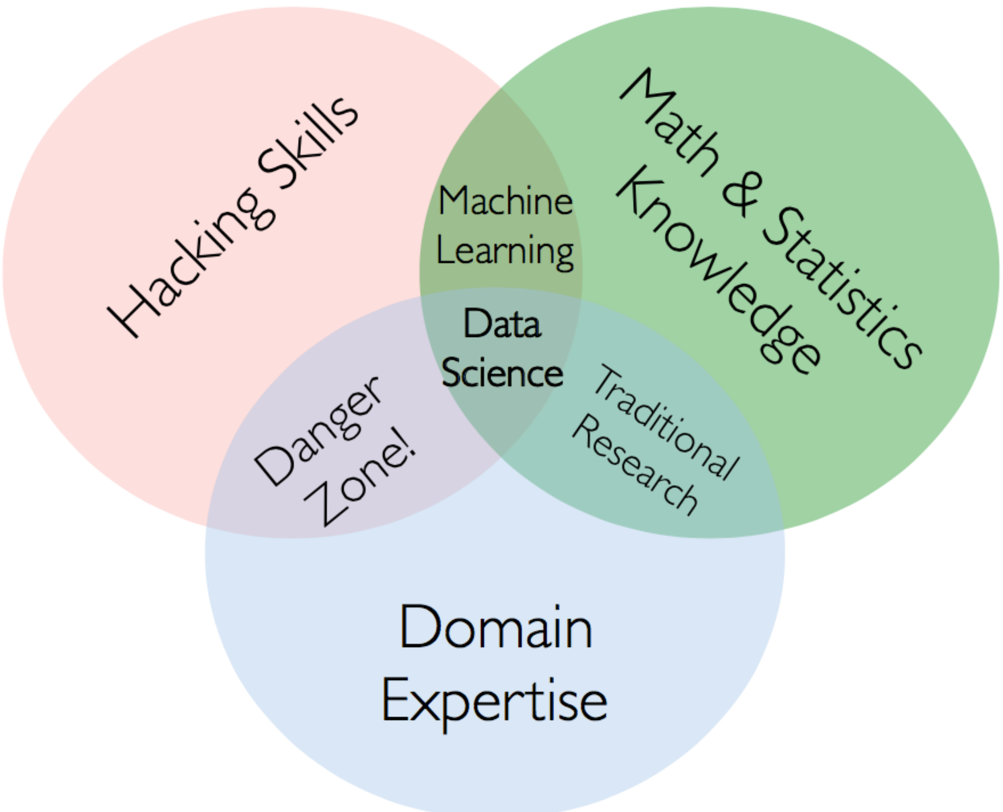
A Simple Solution: Just ask!!!

Experts

What causes high blood pressure?

Statistician/Data Analyst

Revisiting Venn Diagram



http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram

Ok, but can we do this from data alone?

- No experts
- Experts do not know either
- Discover features experts do not know
- …

Feature Selection

We actually need two things:

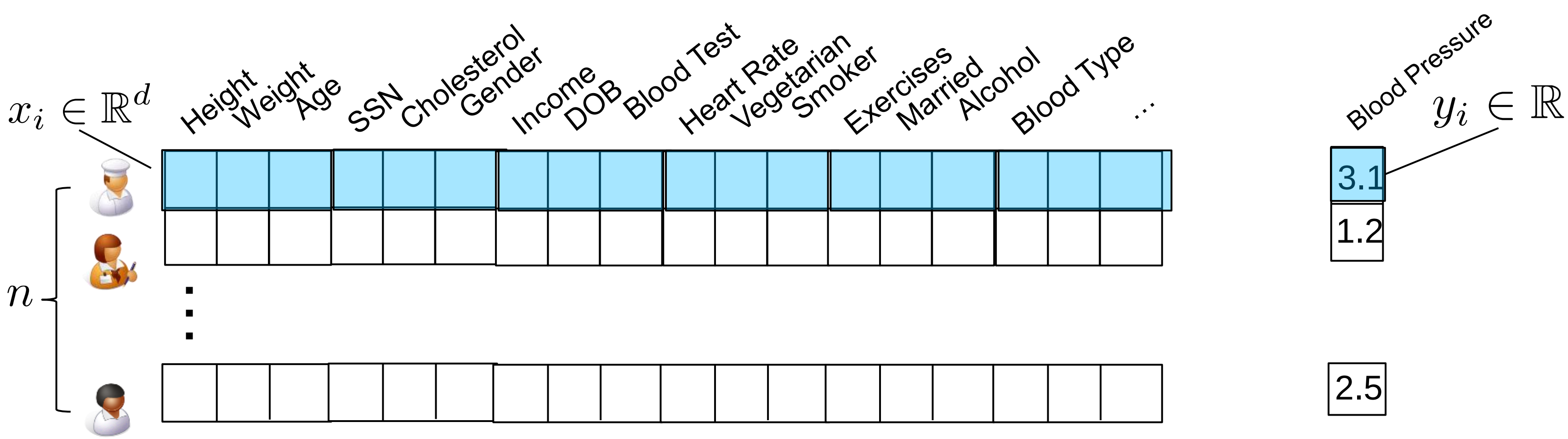
- A procedure for selecting features
- A way of measuring whether this selection is good

Feature Selection

We actually need two things:

- A procedure for selecting features
- **A way of measuring whether this selection is good**

How can I tell if I have a good set of features?

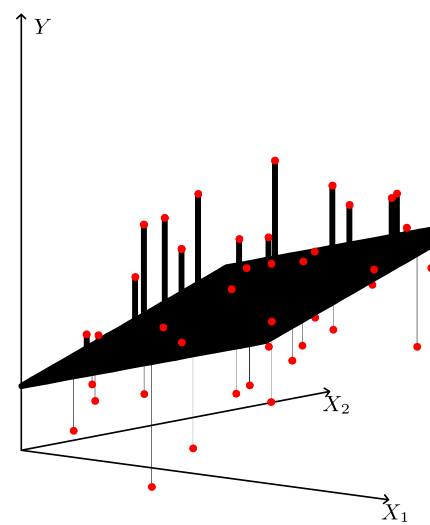


Least-Squares Estimator (LSE)

Residual Sum of Squares

$\mathrm{RSS}(\beta)$

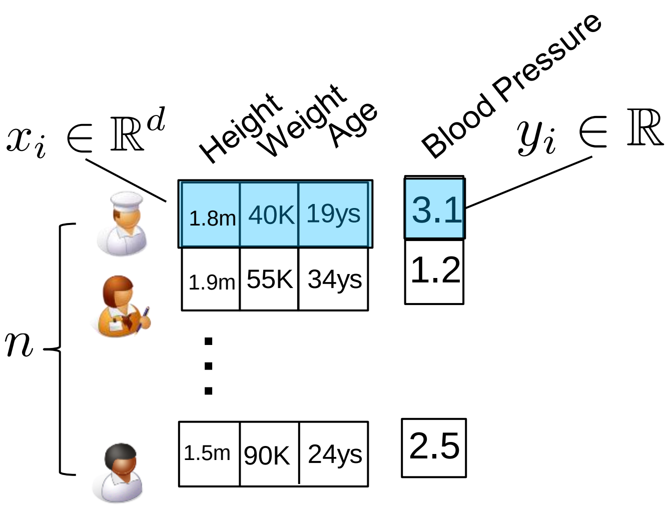
$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \langle \beta, x_i \rangle)^2$$
$$= \arg\min_{\beta \in \mathbb{R}^d} \|X\beta - y\|_2^2$$



Q: Can I use RSS to see if I have a good set of features?

Northeastern

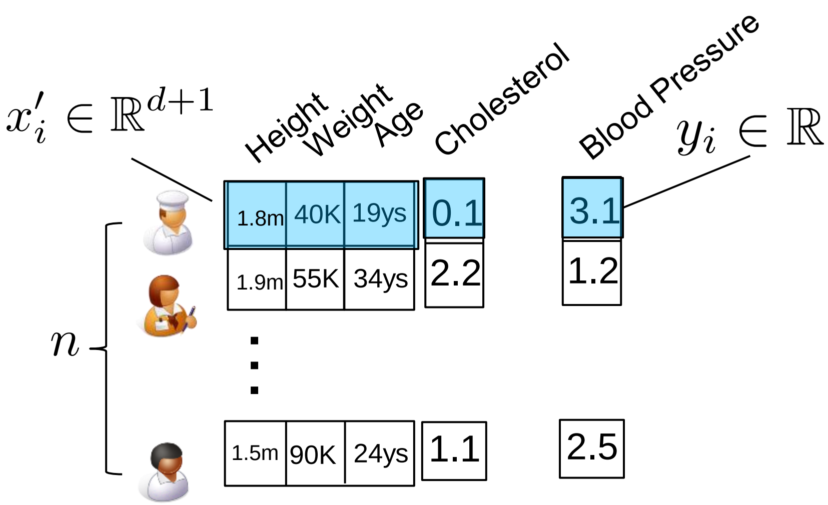
Residual Sum of Squares



$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \langle \beta, x_i \rangle)^2$$

$$\mathrm{RSS}(\hat{\beta}) = \sum_{i=1}^{n} (y_i - \hat{\beta}^\top x_i)$$

Residual Sum of Squares: Adding a New Feature



$$\hat{\beta}' = \arg\min_{\beta \in \mathbb{R}^{d+1}} \sum_{i=1}^{n} (y_i - \beta^\top x'_i)^2$$

$$\mathrm{RSS}(\hat{\beta}') = \sum_{i=1}^{n} (y_i - \hat{\beta}'^\top x'_i)$$

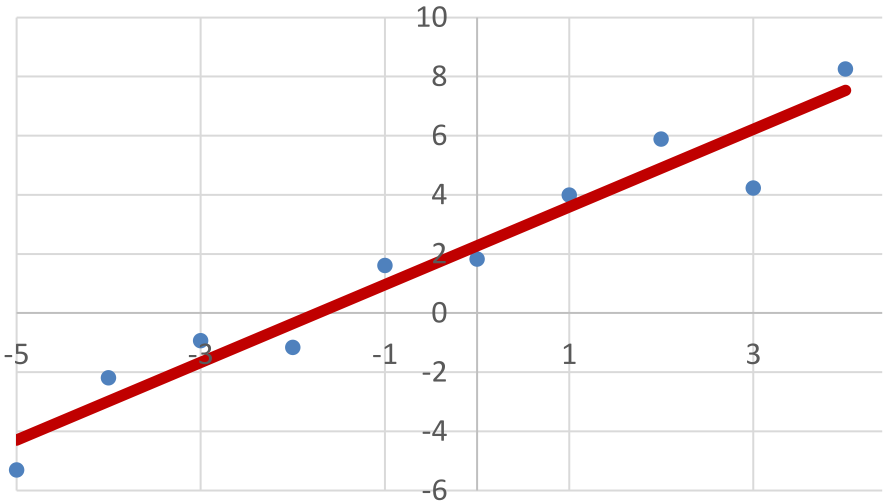
$$\overset{?}{\lessgtr} \mathrm{RSS}(\hat{\beta})$$

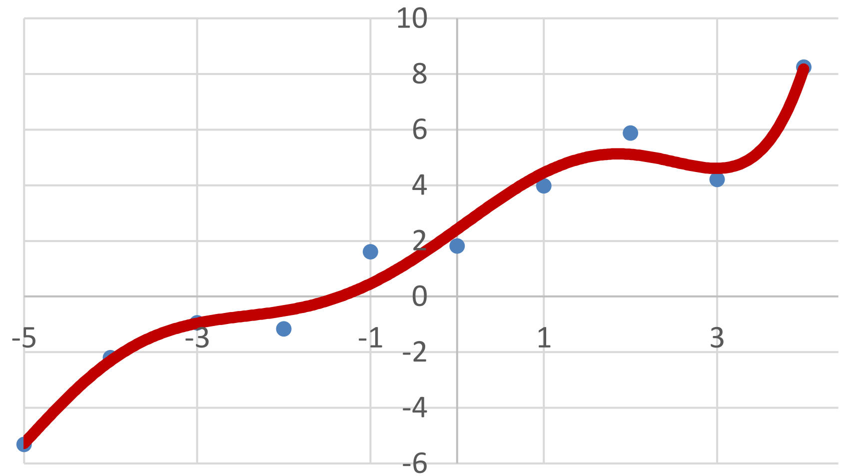
Adding Features Decreases RSS

Proof:

$$\begin{aligned}
\mathtt{RSS}(\hat\beta') &= \min_{\beta\in\mathbb{R}^{d+1}} \mathtt{RSS}(\beta) \\
&\leq \mathtt{RSS}\big((\hat\beta, 0.0)\big) \\
&= \sum_{i=1}^{n}\left(y_i - (\hat\beta, 0.0)^\top x_i'\right)^2 \\
&= \sum_{i=1}^{n}\left(y_i - \hat\beta^\top x_i\right)^2 \\
&= \mathtt{RSS}(\hat\beta)
\end{aligned}$$

Same Principle As Overfitting!





RSS(linear) > RSS(poly(6))

Illustration: Best-Subset Selection

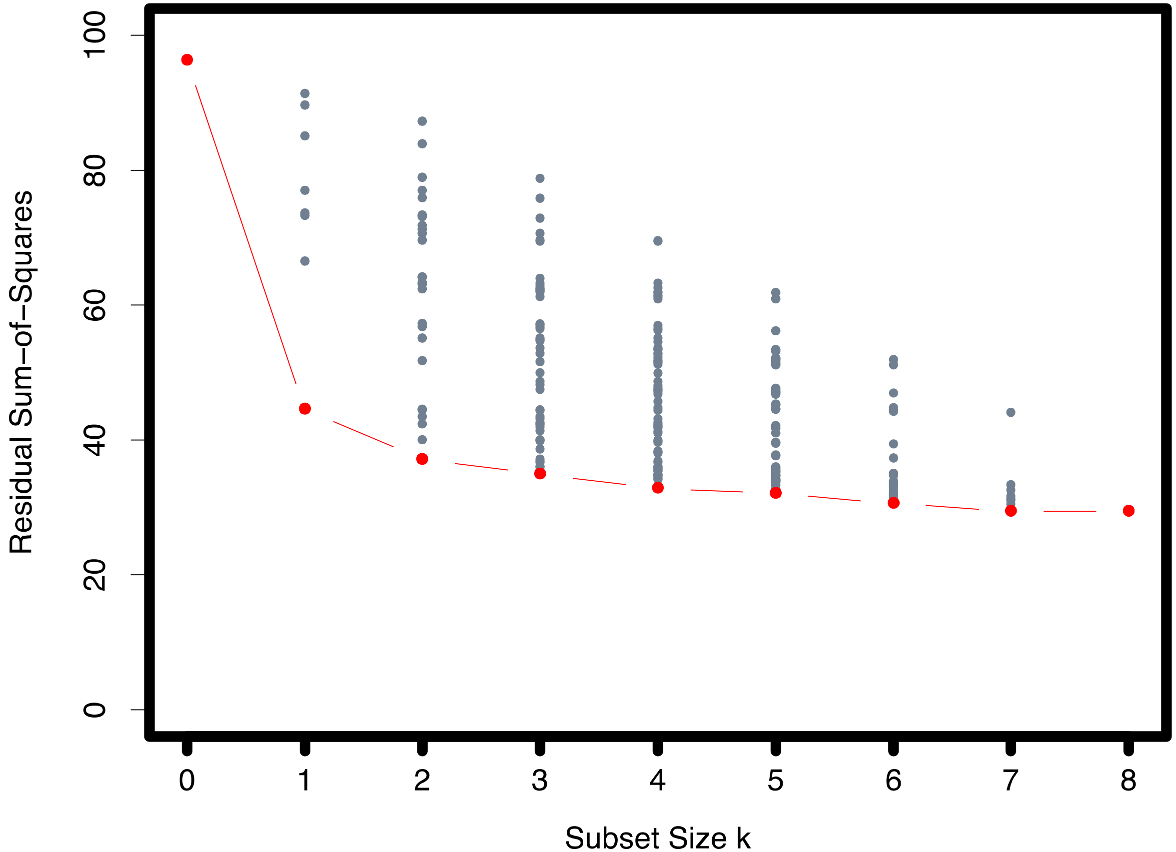
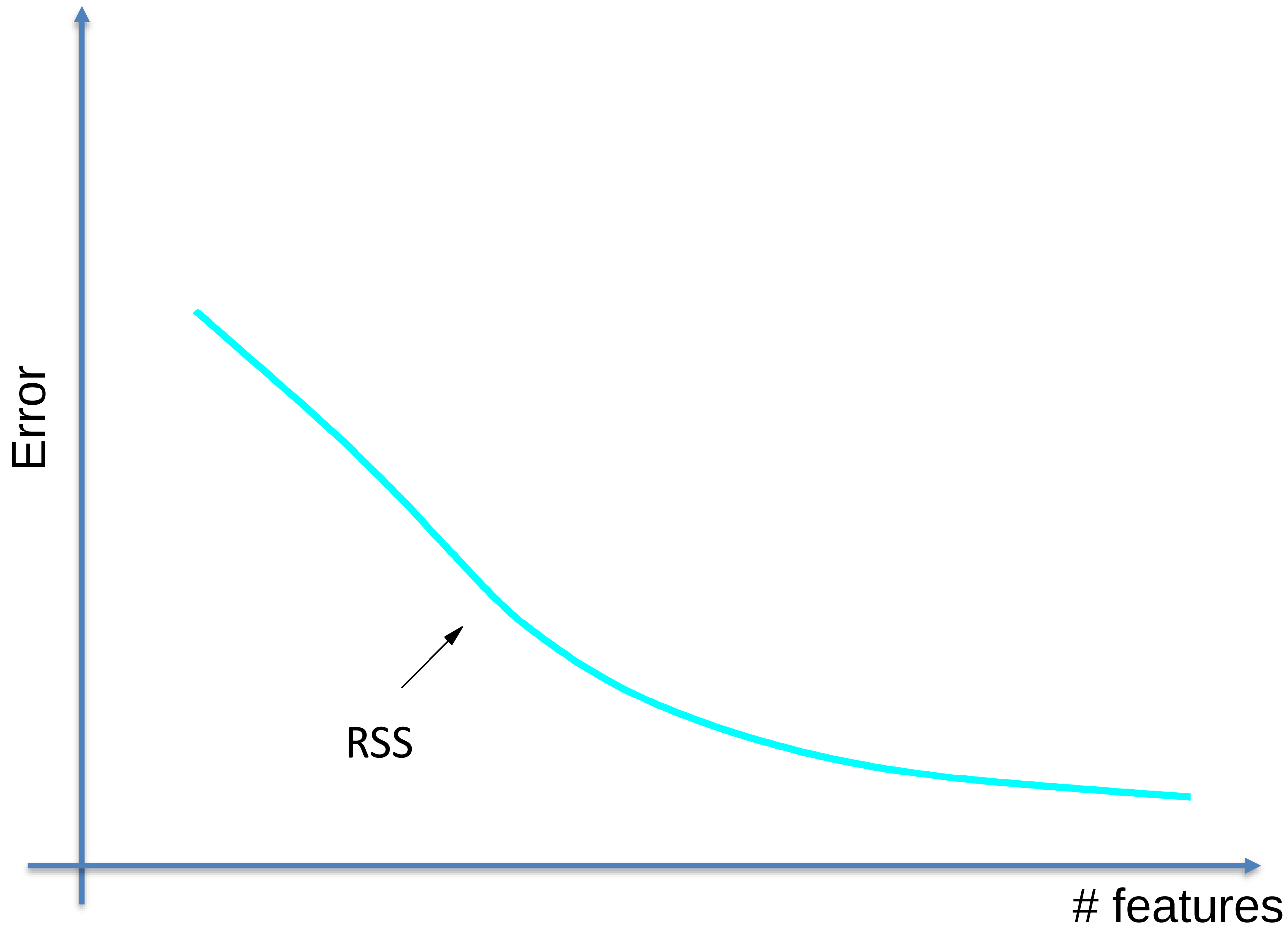


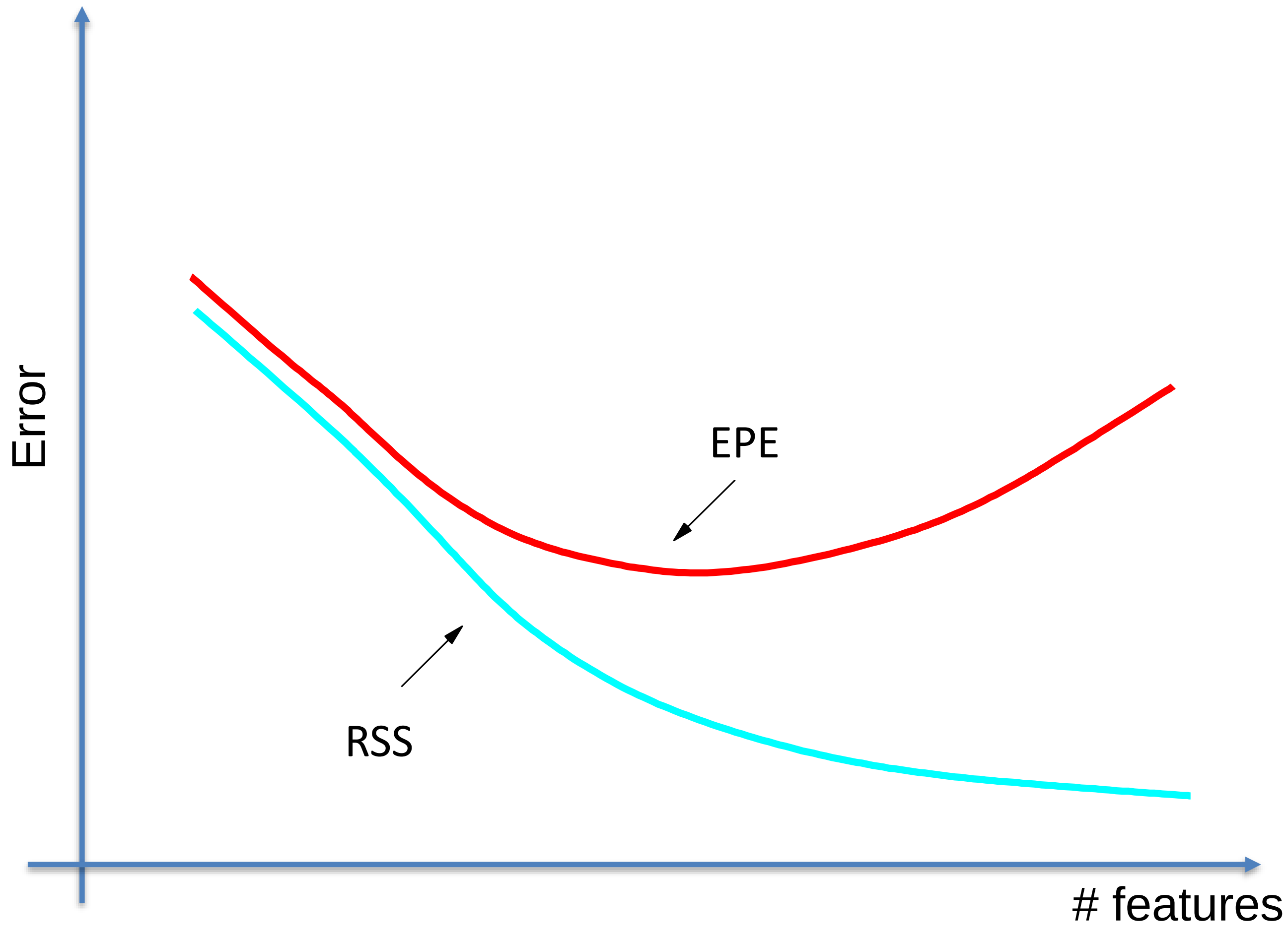
FIGURE 3.5. *All possible subset models for the prostate cancer example. At each subset size is shown the residual sum-of-squares for each model of that size.*

Solution: EPE

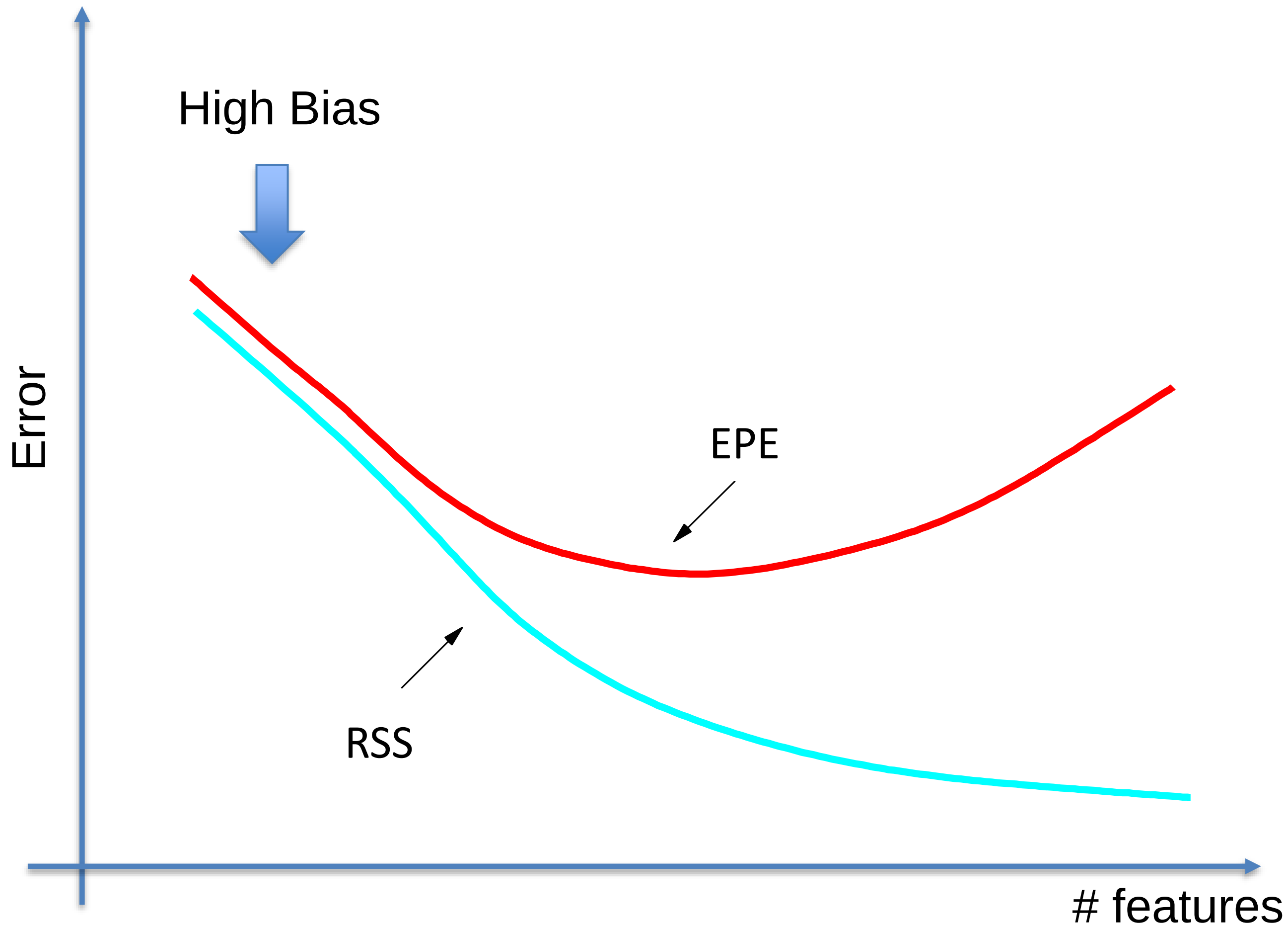


Northeastern

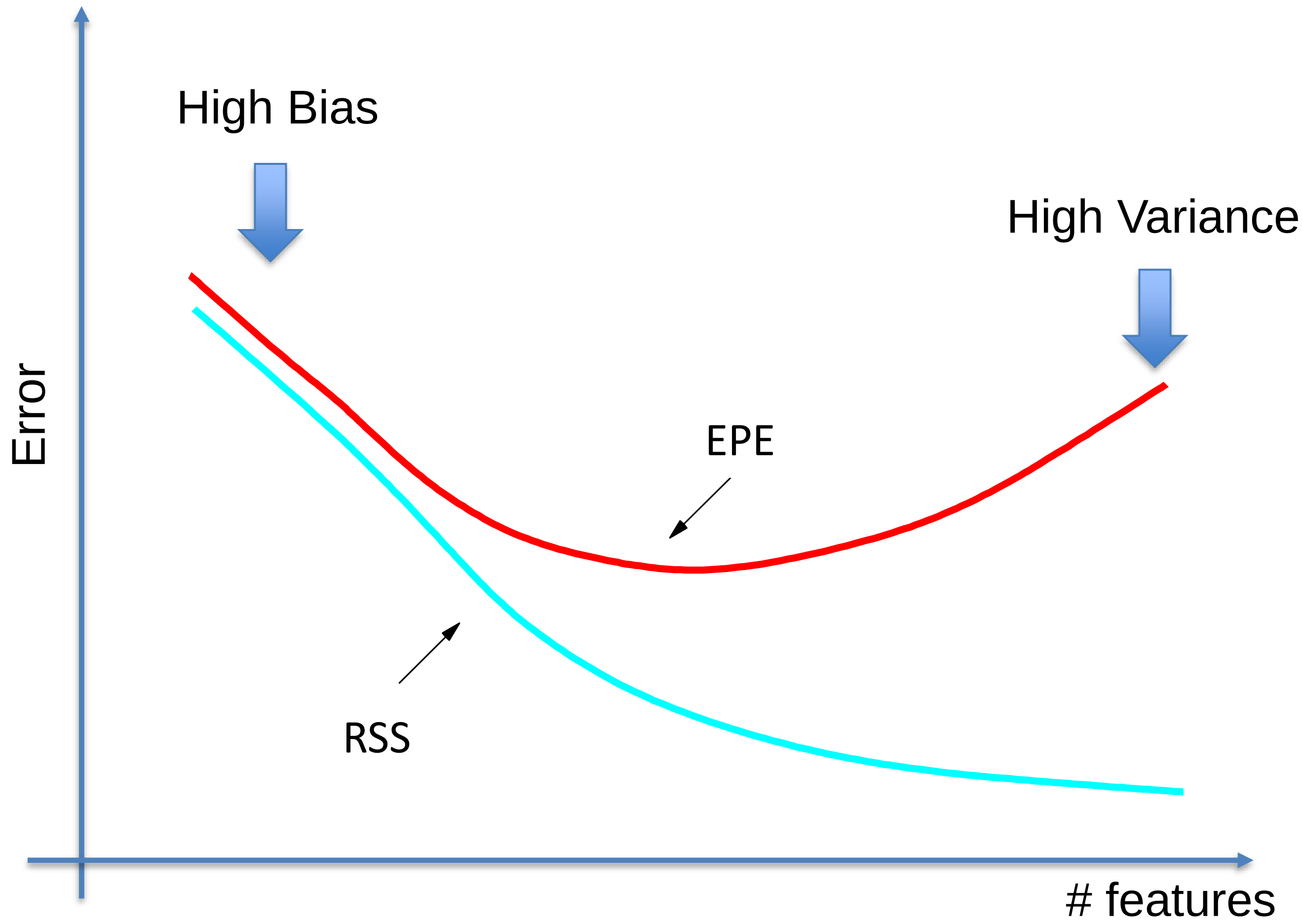
Solution: EPE



Solution: EPE

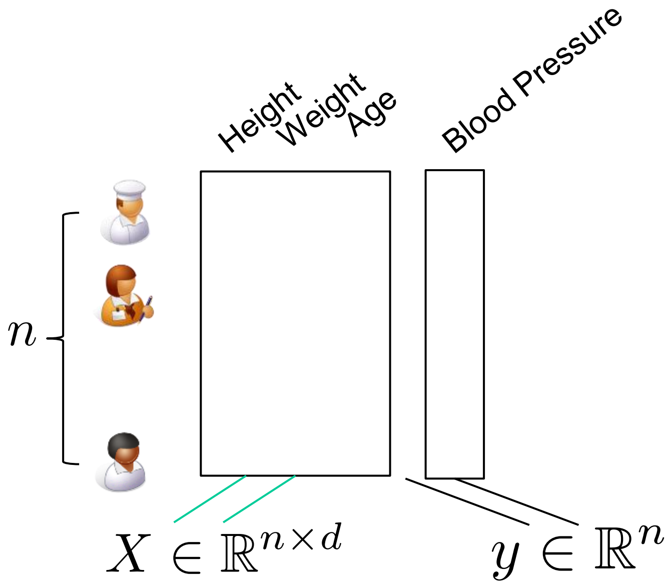


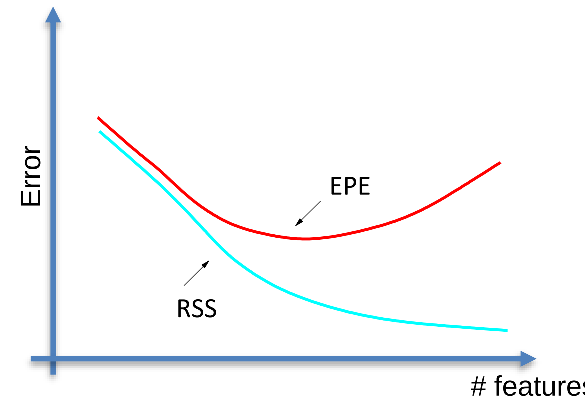
Solution: EPE



Northeastern

What Was EPE Again?

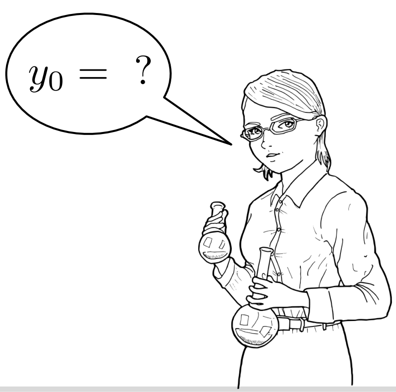




$x_0 =$

1.8m	90K	24ys

$$y_0 = \beta^\top x_0 + \varepsilon_0$$



Estimate: $\hat{y}_0 = \hat{\beta}^\top x_0$

EPE:
$$\begin{aligned}\mathbb{E}[(y_0 - \hat{y}_0)^2] &= \mathbb{E}[(y_0 - \beta^\top x_0)^2] + \mathbb{E}[(\beta^\top x_0 - \hat{\beta}^\top x_0)^2] \\ &= \sigma^2 + {x_0}^\top \mathbb{E}[(\beta - \hat{\beta})(\beta - \hat{\beta})^\top] x_0 \\ &= \sigma^2 + {x_0}^\top \mathrm{Cov}(\hat{\beta}) x_0\end{aligned}$$

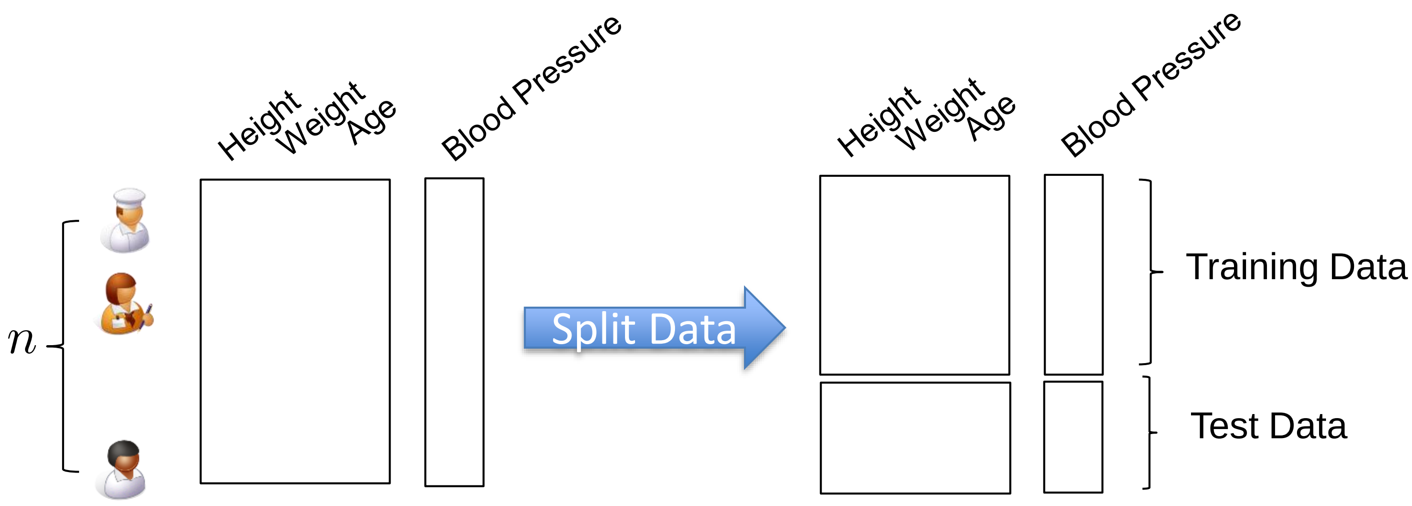
If $x_i, x \in \mathbb{R}^d$ are sampled from the same distribution, then:

$$\mathbb{E}\,[\text{EPE}] \approx \sigma^2 + \sigma^2 \frac{d}{n}$$

Problem: We don't know the distribution!

Solution: use data!!!

Estimating EPE



- **Train** $\hat{\beta}$ by minimizing:

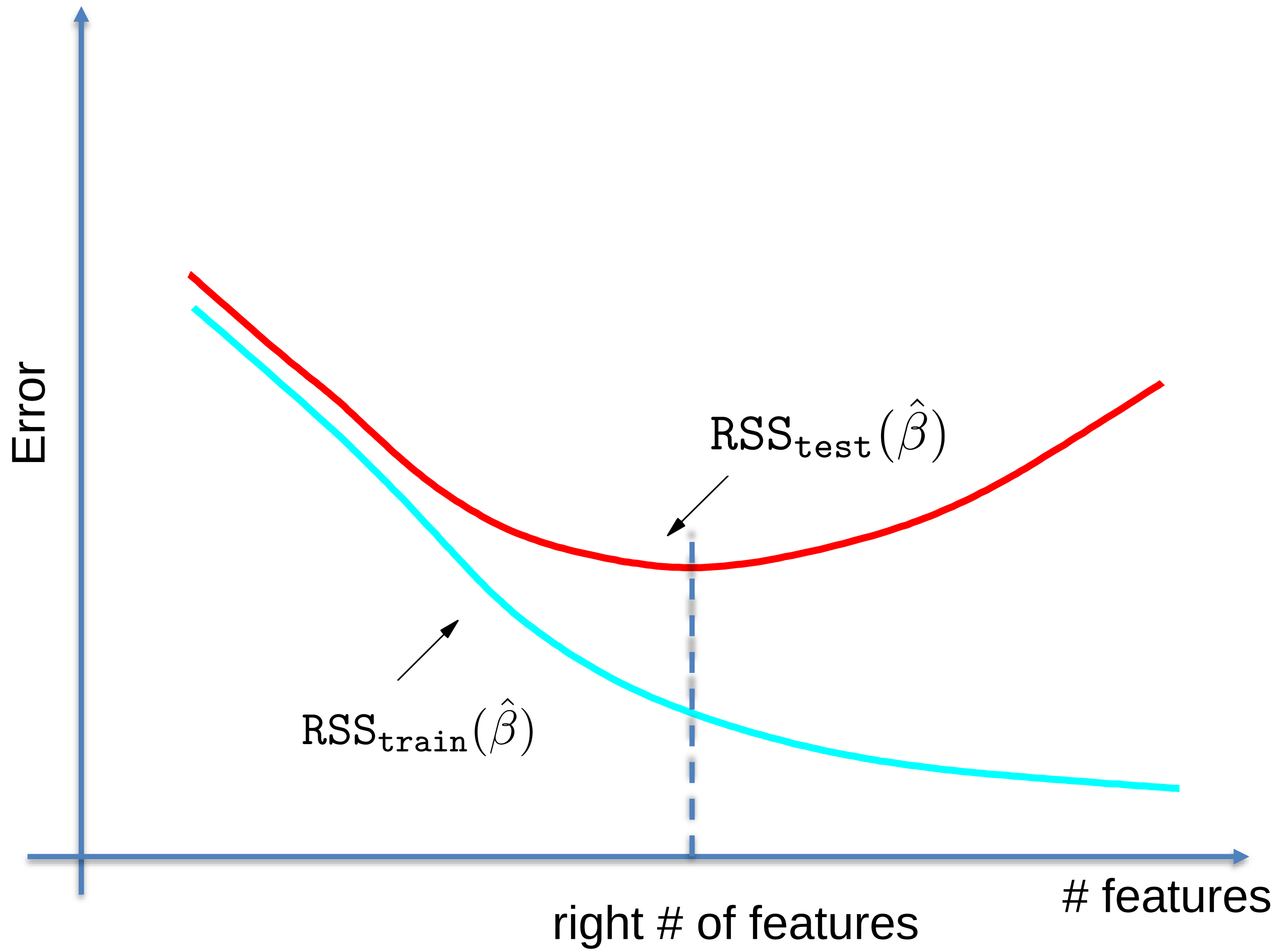
$$\mathrm{RSS}_{\mathtt{train}}(\beta) = \sum_{i \in \mathtt{train}} (y_i - \beta^\top x_i)^2$$

- **Test** $\hat{\beta}$ by evaluating:

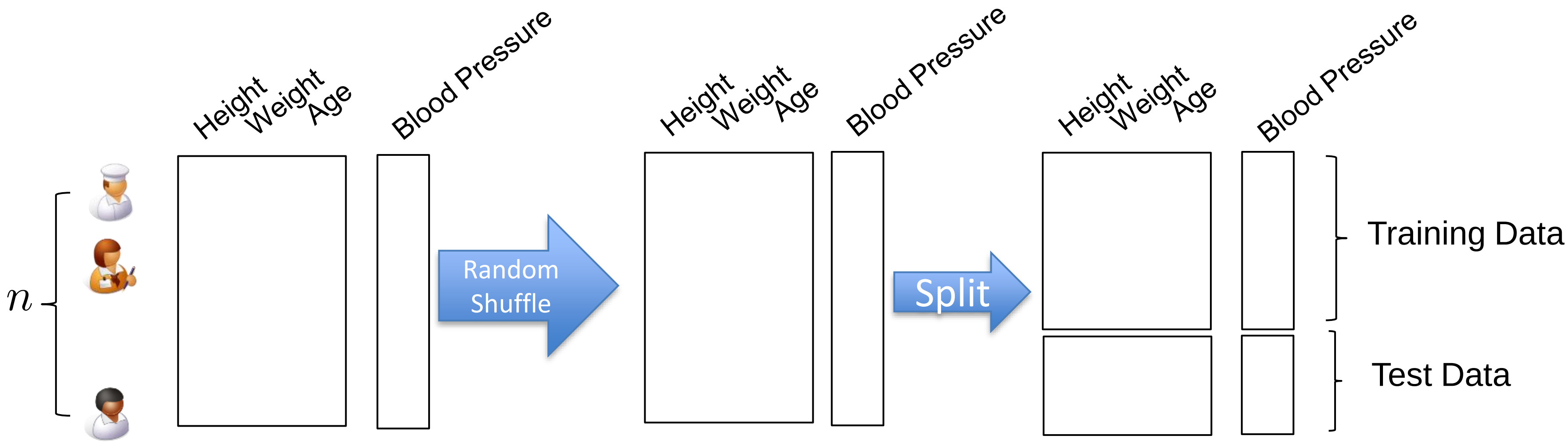
$$\mathrm{RSS}_{\mathtt{test}}(\hat{\beta}) = \sum_{i \in \mathtt{test}} (y_i - \hat{\beta}^\top x_i)^2$$

"Proxy" for EPE!!

Feature Selection Revisited



Improvement #1



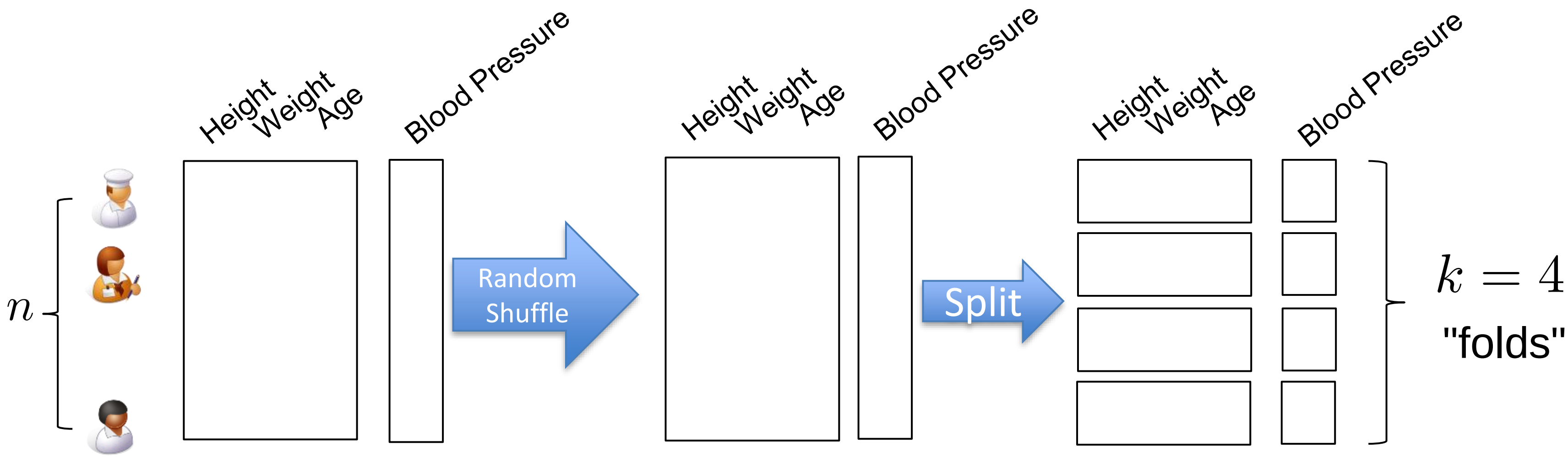
- **Train** $\hat{\beta}$ by minimizing:

$$\mathrm{RSS}_{\mathtt{train}}(\beta) = \sum_{i \in \mathtt{train}} (y_i - \beta^\top x_i)^2$$

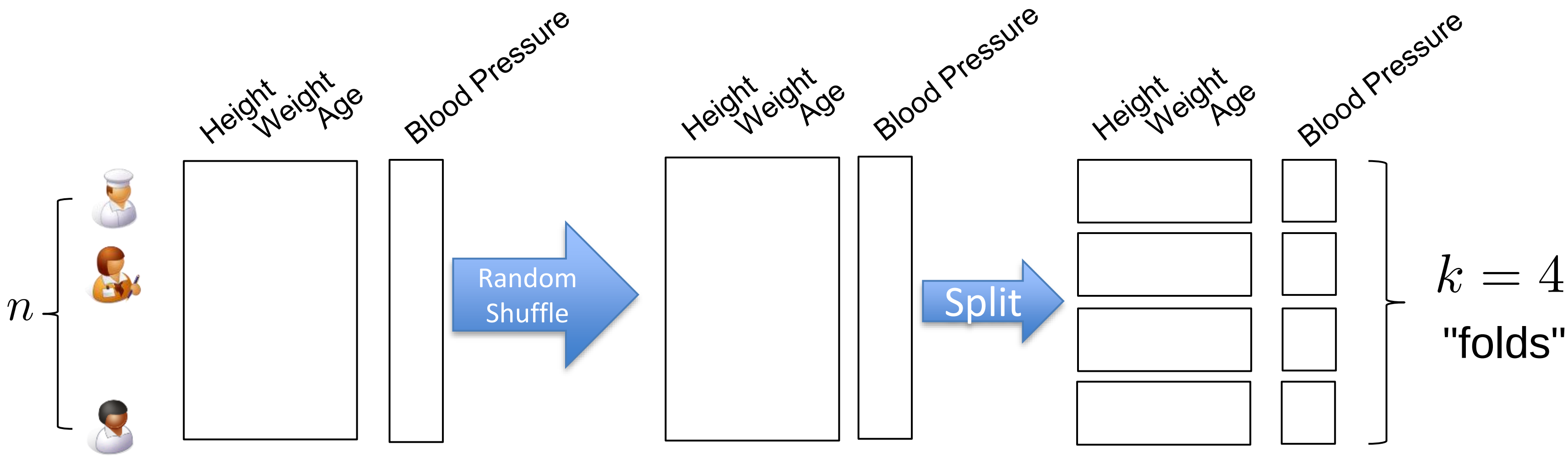
- **Test** $\hat{\beta}$ by evaluating:

$$\mathrm{RSS}_{\mathtt{test}}(\hat{\beta}) = \sum_{i \in \mathtt{test}} (y_i - \hat{\beta}^\top x_i)^2$$

Improvement #2: k-fold Cross Validation

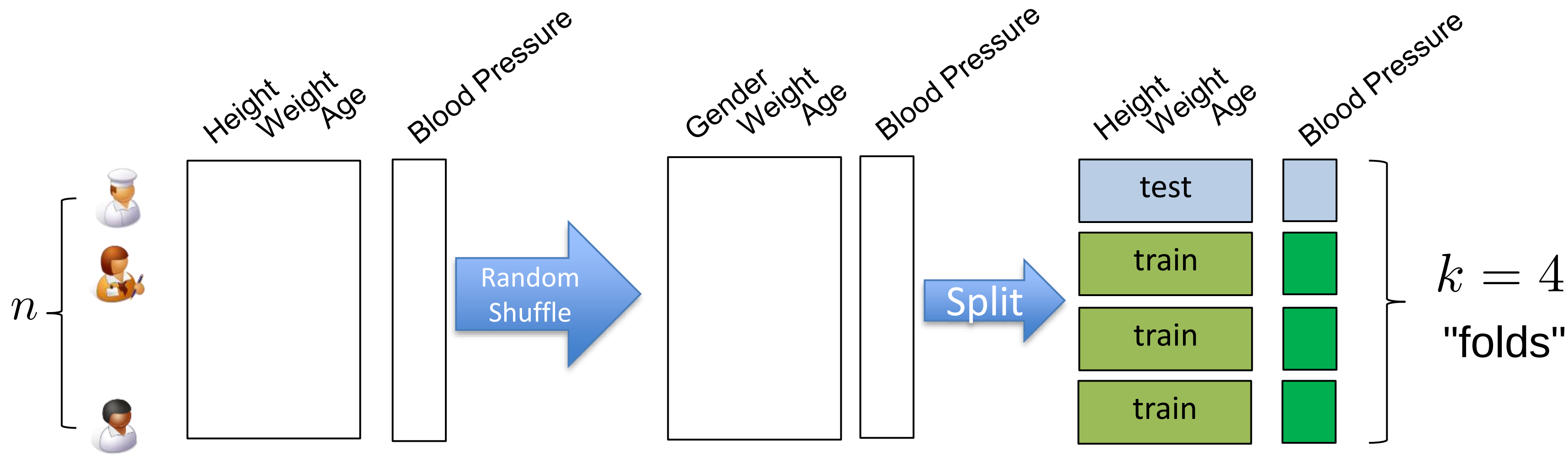


Improvement #2: k-fold Cross Validation



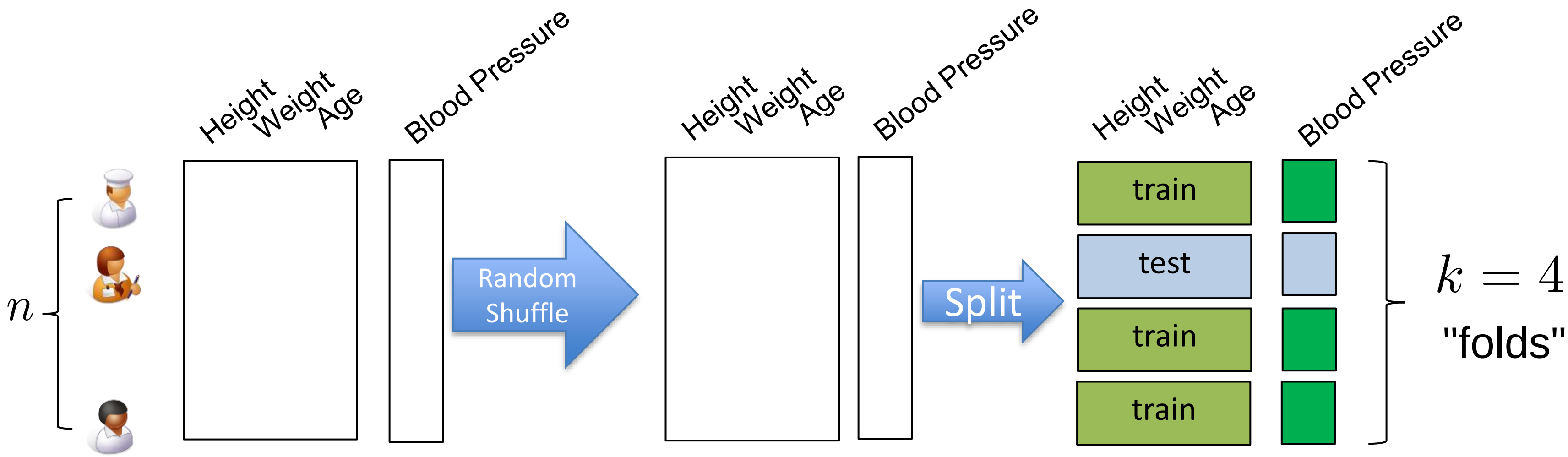
- For each fold $\ell = 1, \ldots, k$:
  - Set $\texttt{test}_\ell$ to include all data in fold ℓ.
  - Put remaining folds in $\texttt{train}_\ell$

Improvement #2: k-fold Cross Validation



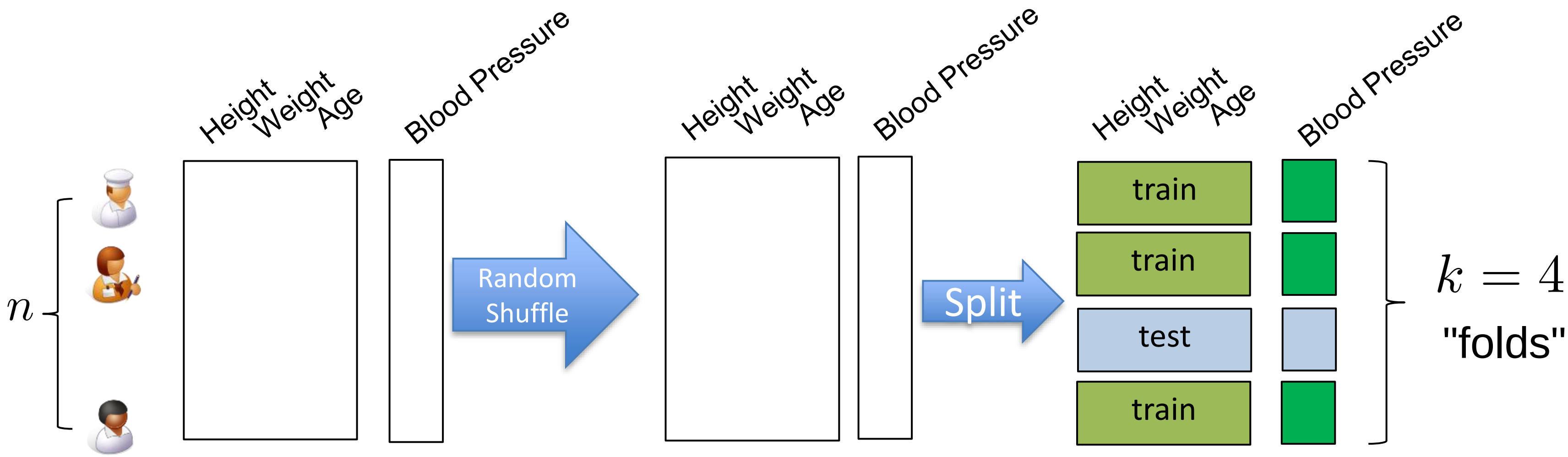
- For each fold $\ell = 1, \ldots, k$:
  - Set $\texttt{test}_\ell$ to include all data in fold ℓ.
  - Put remaining folds in $\texttt{train}_\ell$
  - **Train** $\hat{\beta}$ by minimizing: $\text{RSS}_{\texttt{train}_\ell}(\beta) = \sum_{i \in \texttt{train}_\ell} (y_i - \beta^\top x_i)^2$
  - **Test** $\hat{\beta}$ by evaluating: $\text{RSS}_{\texttt{test}_\ell}(\hat{\beta}) = \sum_{i \in \texttt{test}_\ell} (y_i - \hat{\beta}^\top x_i)^2$

Improvement #2: k-fold Cross Validation



- For each fold $\ell = 1, \ldots, k$:
  - Set $\texttt{test}_\ell$ to include all data in fold ℓ.
  - Put remaining folds in $\texttt{train}_\ell$
  - **Train** $\hat{\beta}$ by minimizing: $\texttt{RSS}_{\texttt{train}_\ell}(\beta) = \sum_{i \in \texttt{train}_\ell} (y_i - \beta^\top x_i)^2$
  - **Test** $\hat{\beta}$ by evaluating: $\texttt{RSS}_{\texttt{test}_\ell}(\hat{\beta}) = \sum_{i \in \texttt{test}_\ell} (y_i - \hat{\beta}^\top x_i)^2$

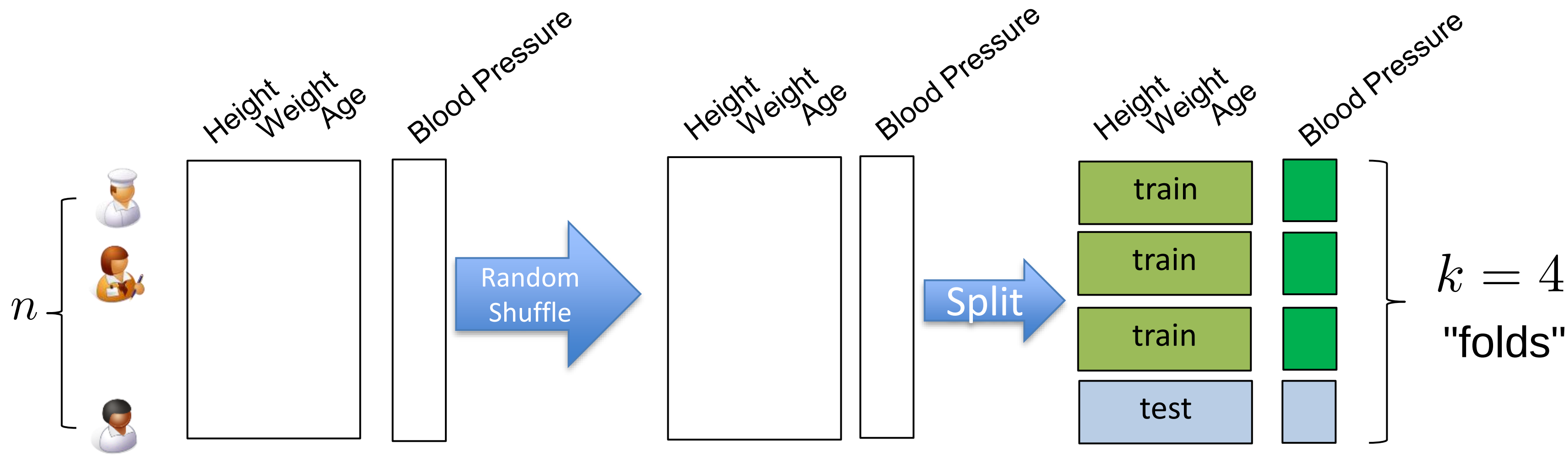
Improvement #2: k-fold Cross Validation



- For each fold $\ell = 1, \ldots, k$:
  - Set $\texttt{test}_\ell$ to include all data in fold ℓ.
  - Put remaining folds in $\texttt{train}_\ell$
  - **Train** $\hat{\beta}$ by minimizing: $\texttt{RSS}_{\texttt{train}_\ell}(\beta) = \sum_{i \in \texttt{train}_\ell} (y_i - \beta^\top x_i)^2$
  - **Test** $\hat{\beta}$ by evaluating: $\texttt{RSS}_{\texttt{test}_\ell}(\hat{\beta}) = \sum_{i \in \texttt{test}_\ell} (y_i - \hat{\beta}^\top x_i)^2$

Northeastern

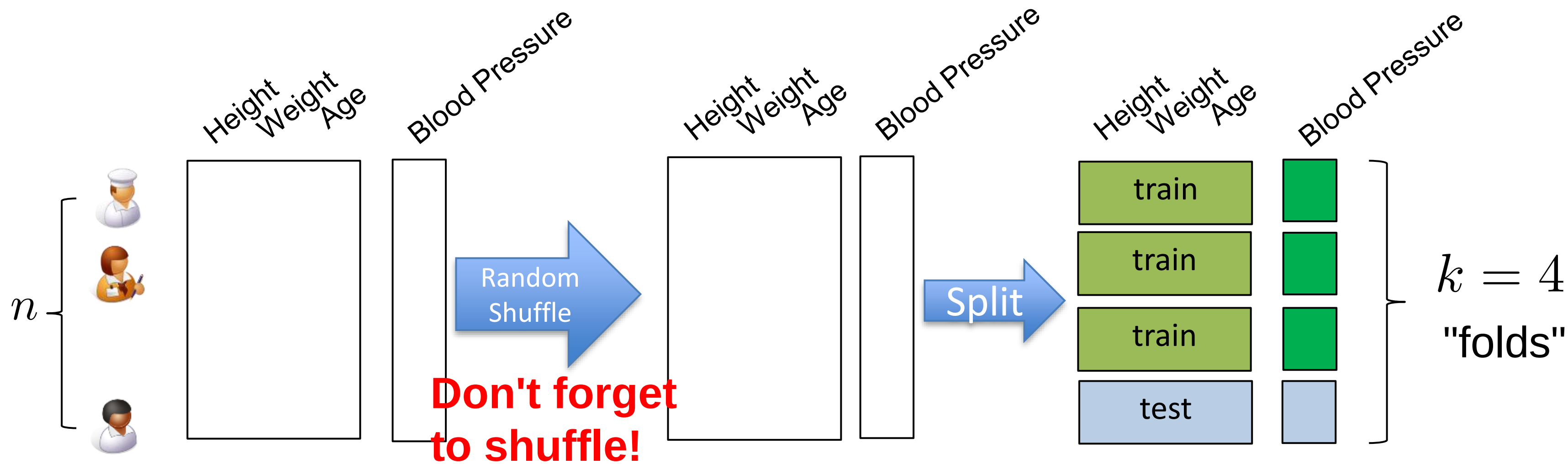
Improvement #2: k-fold Cross Validation



- For each fold $\ell = 1, \ldots, k$:
  - Set $\texttt{test}_\ell$ to include all data in fold ℓ.
  - Put remaining folds in $\texttt{train}_\ell$
  - **Train** $\hat{\beta}$ by minimizing: $\mathrm{RSS}_{\texttt{train}_\ell}(\beta) = \sum_{i \in \texttt{train}_\ell} (y_i - \beta^\top x_i)^2$
  - **Test** $\hat{\beta}$ by evaluating: $\mathrm{RSS}_{\texttt{test}_\ell}(\hat{\beta}) = \sum_{i \in \texttt{test}_\ell} (y_i - \hat{\beta}^\top x_i)^2$
- Quality of solution: $\overline{\mathrm{RSS}} = \frac{1}{k} \sum_{\ell=1}^{k} \mathrm{RSS}_{\texttt{test}_\ell}$

"Proxy" for EPE!!

k-fold Cross Validation

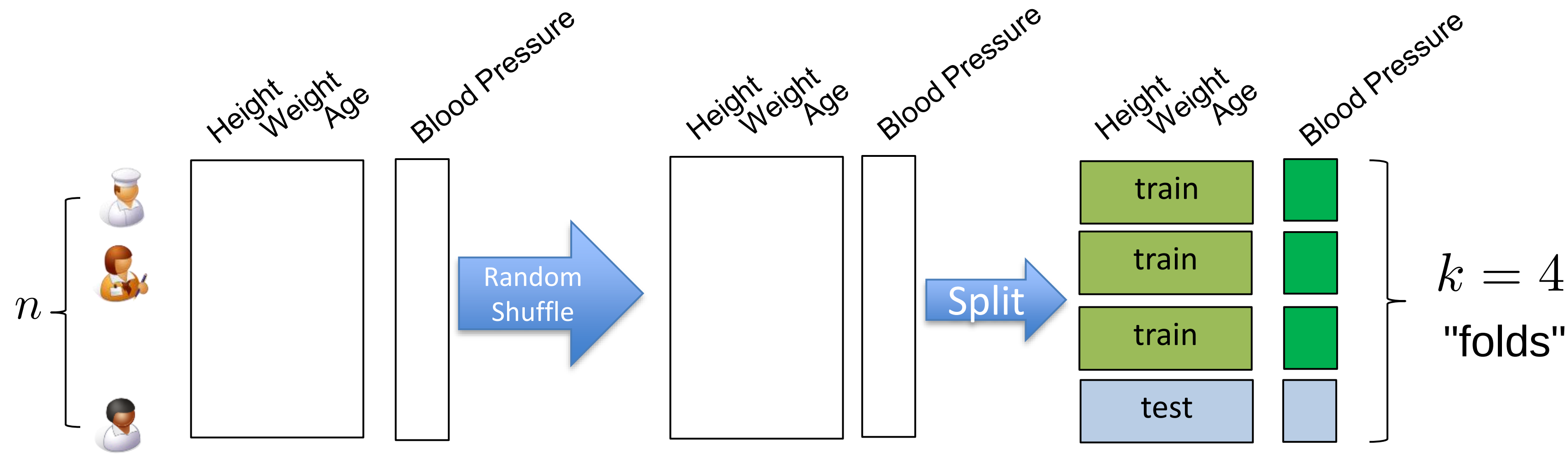


Cross-validation error:

$$\overline{\mathrm{RSS}} = \frac{1}{k}\sum_{\ell=1}^{k} \mathrm{RSS}_{\mathtt{test}_\ell}$$

- ❑ **Less sensitive** to how split happens than train/test
- ❑ Can be applied to **other metrics** (accuracy, precision, recall, AUC)…
- ❑ Can be applied to **pick other parameters** of estimation procedure:
  - ❑ Feature selection
  - ❑ Number of iterations
  - ❑ …
- ❑ Can be used to compute **standard deviation, confidence intervals, etc.**

k-fold Cross Validation



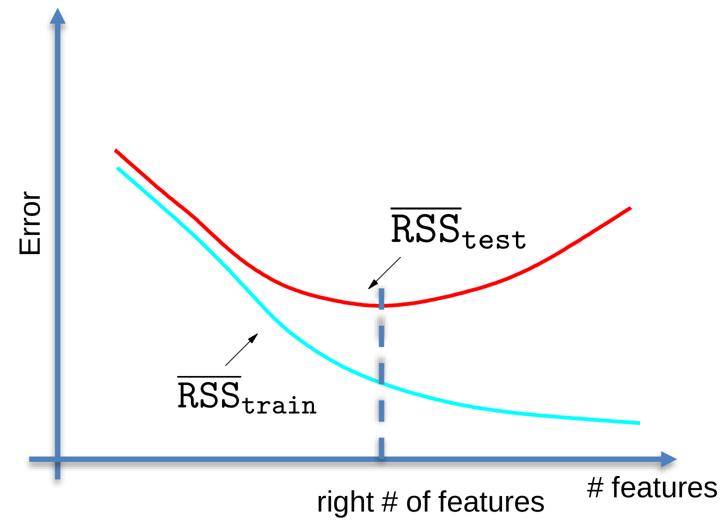
Cross-validation error:

$$\overline{\mathrm{RSS}} = \frac{1}{k} \sum_{\ell=1}^{k} \mathrm{RSS}_{\mathtt{test}_\ell}$$

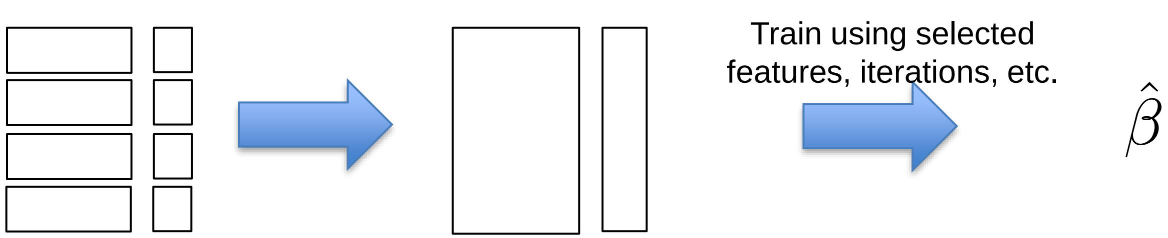
THIS IS AN EXTREMELY IMPORTANT TOPIC!!! IF YOU ONLY REMEMBER A SINGLE THING FROM ENTIRE CLASS, PLEASE REMEMBER TO CROSS-VALIDATE!!!

Finding the Right Features

- Use k-fold CV to find right problem parameters:
  - Features
  - Iterations
  - Regularization parameters (coming up)…



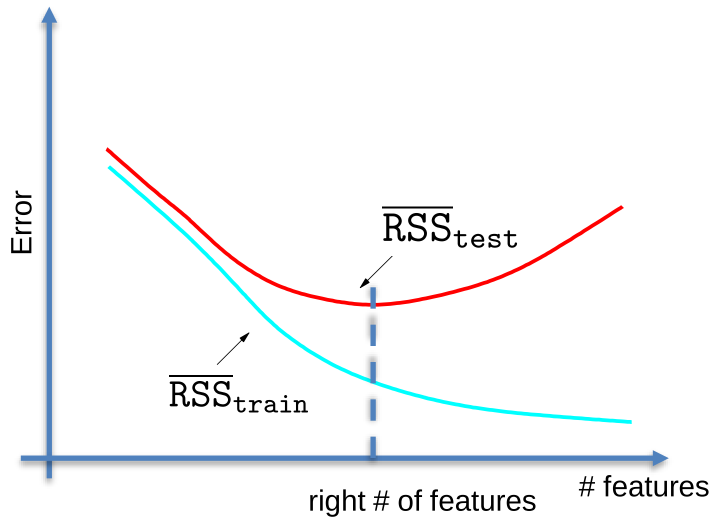
- One model $\hat{\beta}_\ell$ per fold $\ell = 1, \ldots, k$.
  - Fix these parameters and then retrain model **over entire dataset**



Feature Selection

We actually need two things:

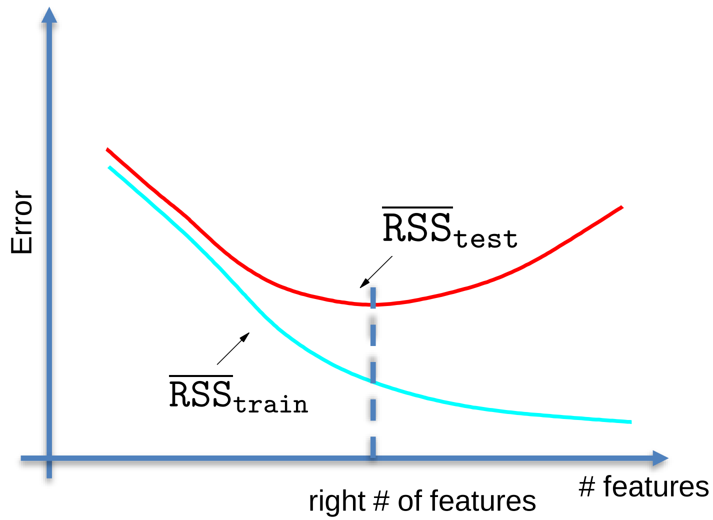
- ❑ A procedure for selecting features
- ❑ A way of measuring whether this selection is good



Feature Selection

We actually need two things:

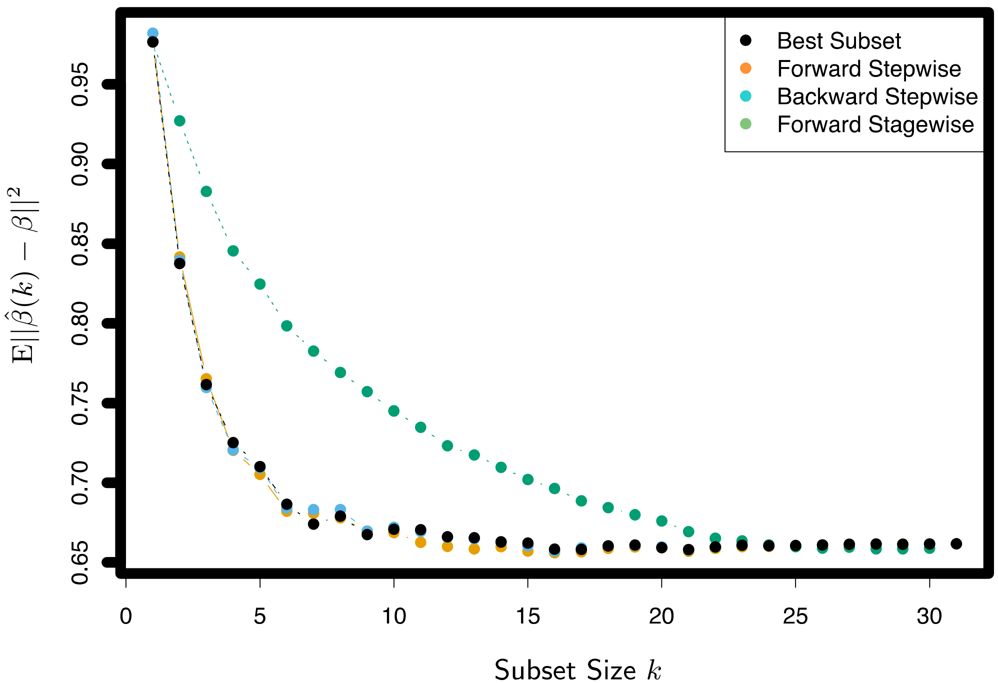
- ❑ **A procedure for selecting features**
- ❑ A way of measuring whether this selection is good



A Few Combinatorial Approaches

Best Subset Selection:

- ❑ Try all subsets of features
- ❑ Too expensive



❑ Greedy approaches:

- ❑ Forward step-wise
- ❑ Backward step-wise
- ❑ Forward stage-wise

❑More efficient

❑Not robust: solutions can change drastically with small changes in data

Northeastern

Shrinkage/Regularization Methods

$x_i \in \mathbb{R}^d$ $\qquad$ $y_i \in \mathbb{R}$

n

Height	Weight	Age	Blood Pressure
1.8m	40K	19ys	3.1
1.9m	55K	34ys	1.2
⋮			
1.5m	90K	24ys	2.5

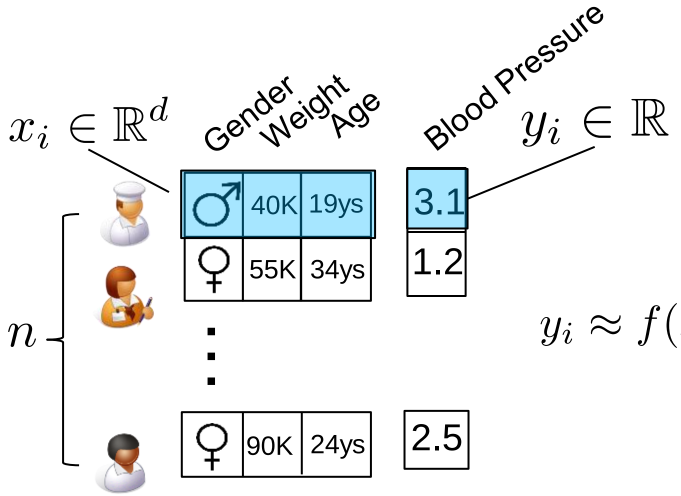
$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \underbrace{\sum_{i=1}^{n} (y_i - \beta^\top x_i)^2}_{\mathrm{RSS}(\beta)} + c(\beta)$$

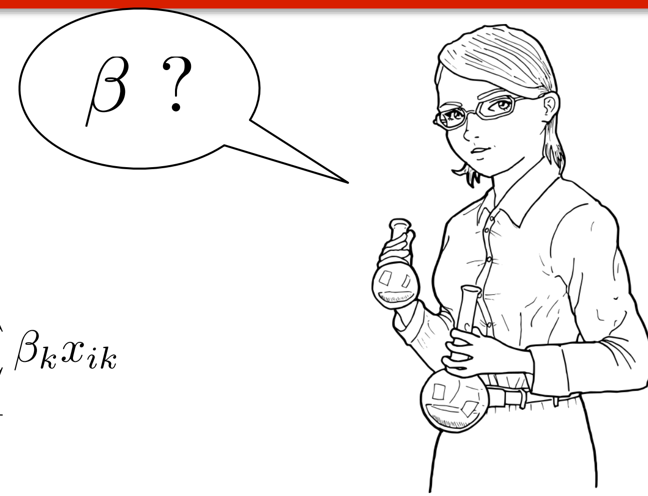
Penalty if β is "complicated"

Occam's razor, KISS (Keep it Simple, Stratis): among two solutions that produce **the same RSS**, we prefer the one that has **smaller complexity**

Shrinkage/Regularization Methods



$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$



$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_0 \text{ , for some } \lambda > 0$$

$\|\beta\|_0 =$ # of non-zero elements of β (i.e., size of β's **support**)

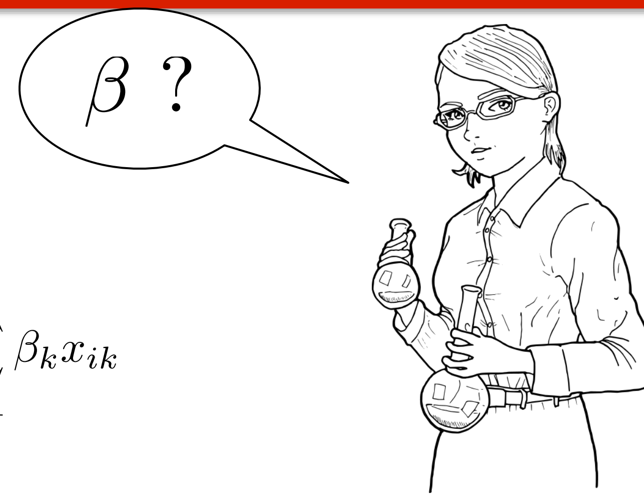
Occam's razor: Between two β with the same RSS, we prefer the one that is **sparser**, i.e., has fewer features.

Shrinkage/Regularization Methods

$x_i \in \mathbb{R}^d$ $\qquad$ $y_i \in \mathbb{R}$

	Height	Weight	Age	Blood Pressure
n	1.8m	40K	19ys	3.1
	1.9m	55K	34ys	1.2
	⋮			
	1.5m	90K	24ys	2.5

$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$



$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_0 \text{ , for some } \lambda > 0$$

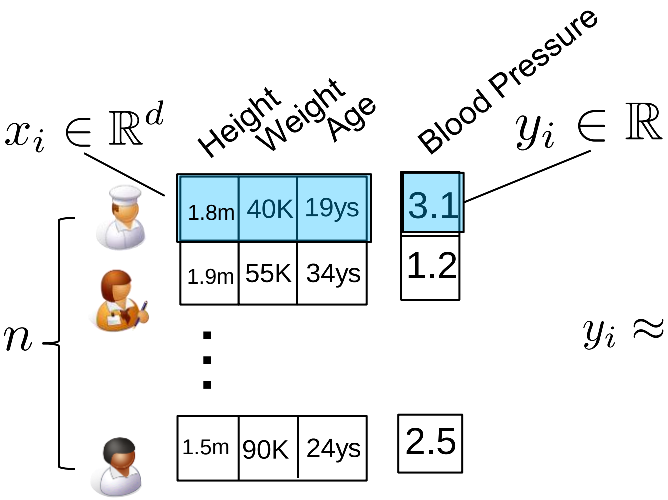
$\|\beta\|_0 =$ # of non-zero elements of β (i.e., size of β's **support**)

$\lambda \gg 0$: optimal solution contains only zeros

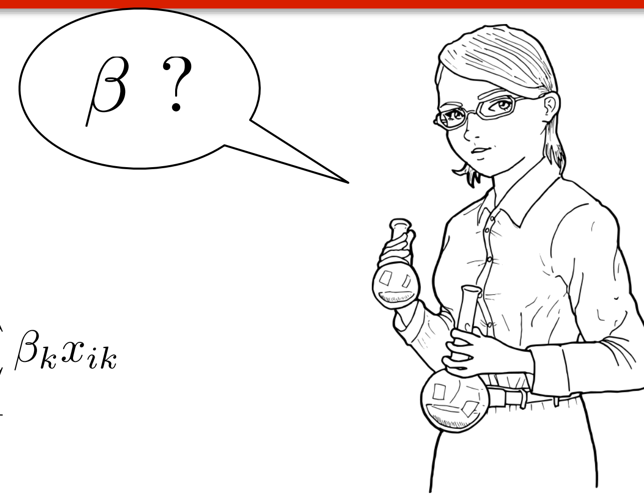
$\lambda = 0$: linear regression

Varying λ can be used for feature selection!

Shrinkage/Regularization Methods



$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$



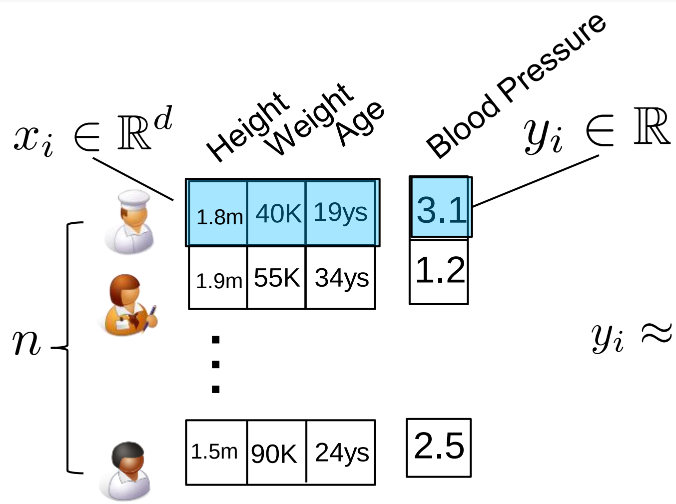
$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_0 \text{ , for some } \lambda > 0$$

Problem: Alas, this is **not a convex objective!**

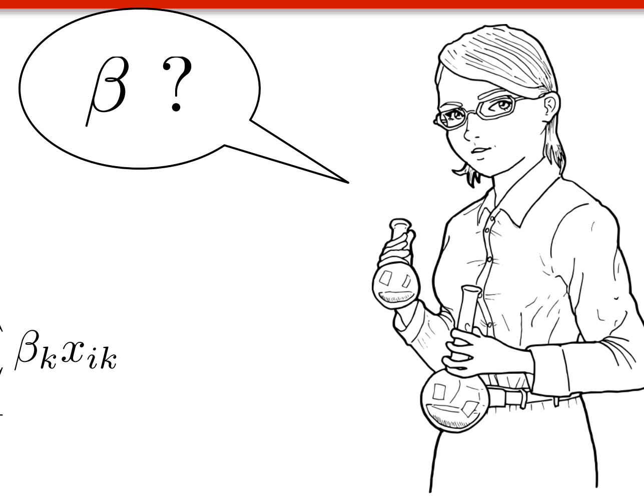
Solution: We replace it with **convex relaxations**

Northeastern

Ridge Regression



$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

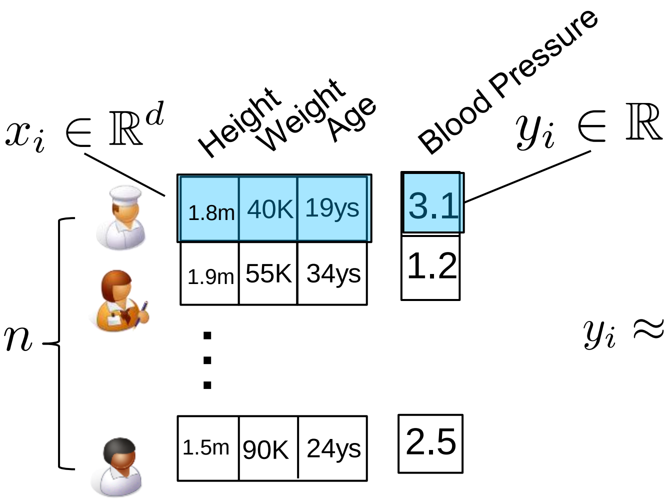


$$\hat{\beta}^{\texttt{ridge}} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_2^2, \text{ for some } \lambda > 0$$

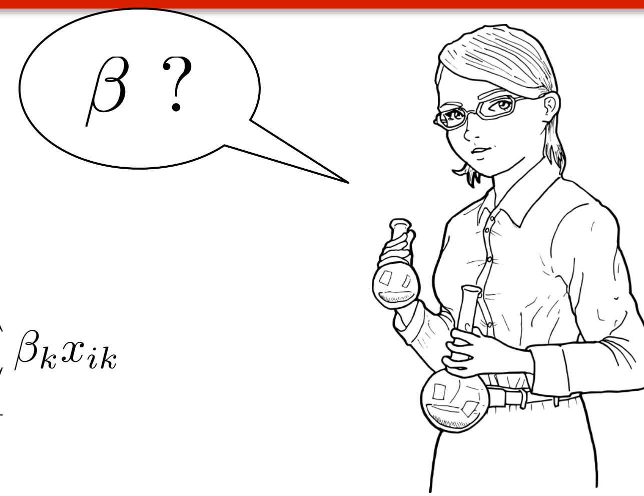
where $\|\beta\|_2^2 = \beta^\top \beta = \sum_{k=1}^{d} \beta_k^2$

Strongly Convex!!!!

Lasso Regression



$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

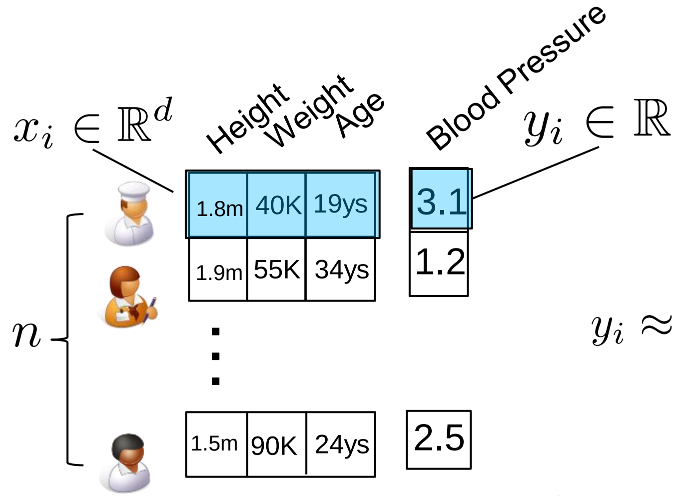


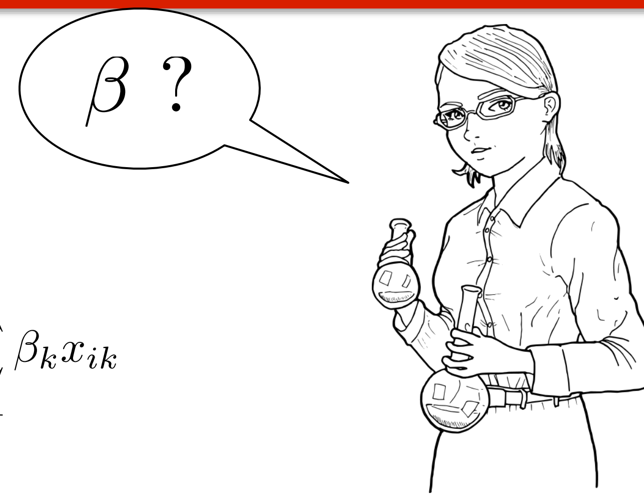
$$\hat{\beta}^{\texttt{lasso}} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_1, \text{ for some } \lambda > 0$$

where $\|\beta\|_1 = \sum_{k=1}^{d} |\beta_k|$

Convex!
(not differentiable)

Ridge Regression





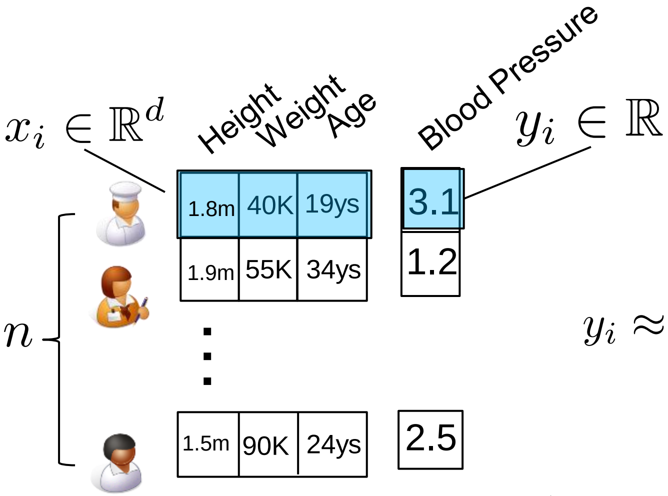
$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

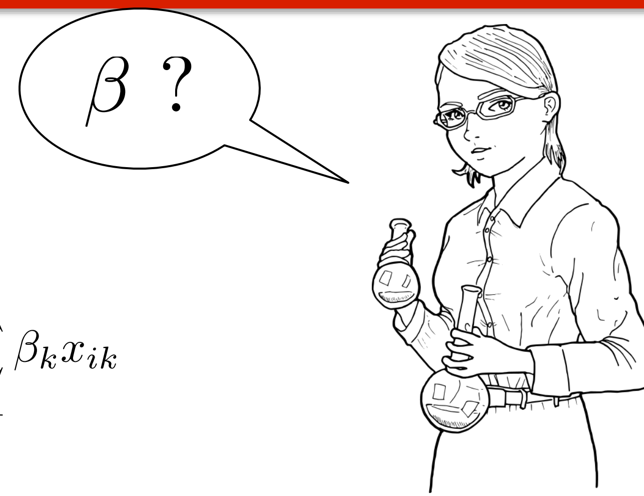
$$\hat{\beta}^{\texttt{ridge}} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \underbrace{\lambda \|\beta\|_2^2}$$, for some $\lambda > 0$

l2-penalty,
ridge penalty,
regularization term,

Northeastern

Ridge Regression



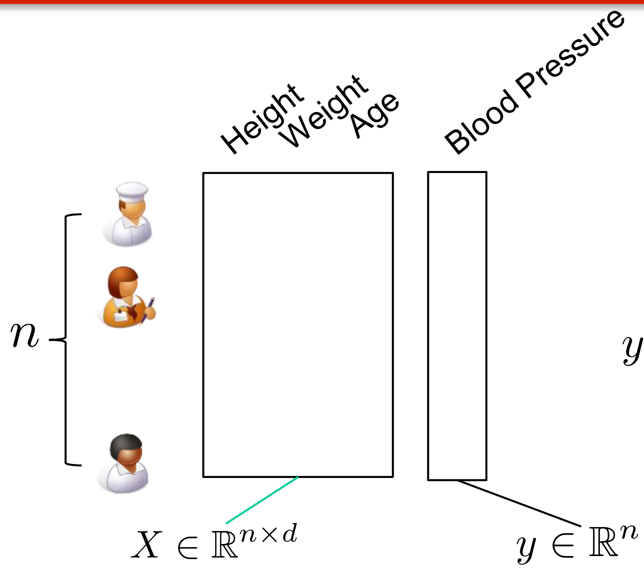


$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

$$\hat{\beta}^{\texttt{ridge}} = \underset{\beta \in \mathbb{R}^d}{\arg\min} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_2^2$$, for some $\lambda > 0$

regularization parameter

Ridge Regression



$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

$$F(\beta) = \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_2^2 \text{ , for some } \lambda > 0$$

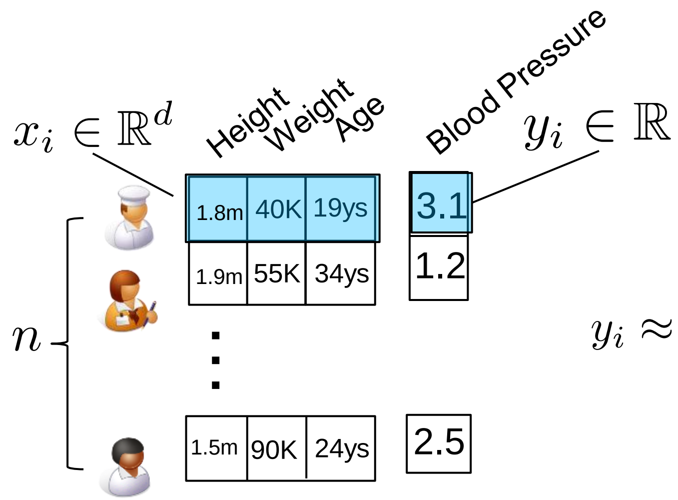
$$= \|X\beta - y\|_2^2 + \lambda \|\beta\|_2^2$$

$$\nabla F(\beta) = 2X^\top X\beta - 2X^\top y + 2\lambda\beta$$

$$\nabla^2 F(\beta) = 2(X^\top X + \lambda I) \succ 0$$

$$\nabla F(\hat{\beta}) = 0 \Leftrightarrow \hat{\beta} = \underbrace{(X^\top X + \lambda I)}_{\text{Invertible for } \lambda > 0!}{}^{-1} X^\top y$$

Ridge Regression: Intuition #1



$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

For every $\lambda \geq 0$, there exists a $t \geq 0$ such that the two problems produce **the same solution**

Minimize: $\sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_2^2$

subject to: $\beta \in \mathbb{R}^d$

Minimize: $\sum_{i=1}^{n} (y_i - \beta^\top x_i)^2$

subject to: $\|\beta\|_2^2 \leq t$

Ridge Regression: Intuition #1

For every $\lambda \geq 0$, there exists a $t \geq 0$ such that the two problems produce **the same solution**

PROBLEM 1

Minimize: $$\sum_{i=1}^{n}(y_i - \beta^\top x_i)^2 + \lambda\|\beta\|_2^2$$

subject to: $$\beta \in \mathbb{R}^d$$

PROBLEM 2

Minimize: $$\sum_{i=1}^{n}(y_i - \beta^\top x_i)^2$$

subject to: $$\|\beta\|_2^2 \leq t$$

Proof: Given a $\lambda \geq 0$, let β^* be an optimal solution to PROBLEM 1.

Let $t = \|\beta^*\|_2^2$. Then, β^* is an optimal solution to PROBLEM 2 for this $t \geq 0$.

Suppose not. Then, there exists a $\beta' \neq \beta^*$ such that $\|\beta'\|_2^2 \leq t$ and

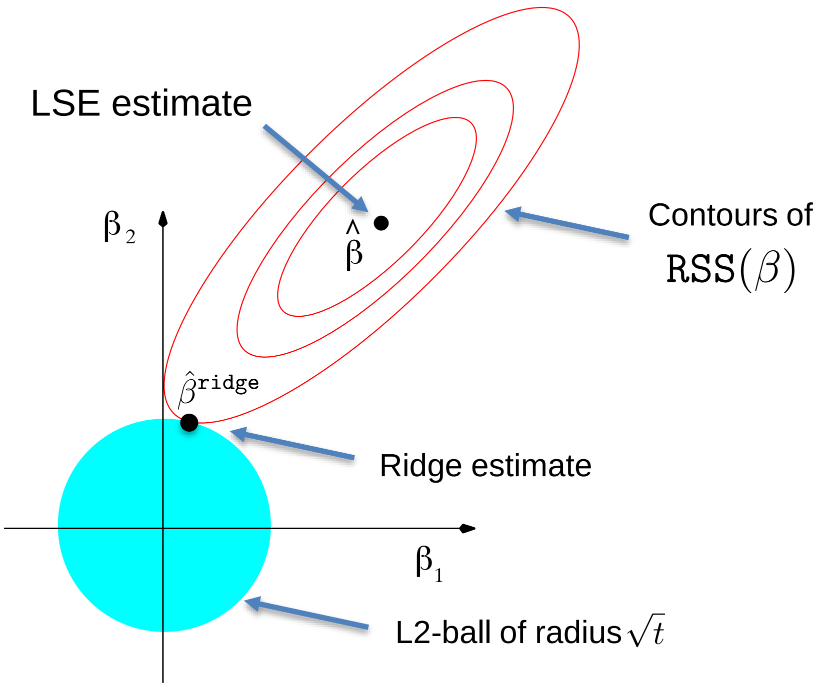
$$\sum_{i=1}^{n}(y_i - {\beta'}^\top x_i)^2 < \sum_{i=1}^{n}(y_i - {\beta^*}^\top x_i)^2$$

But then

$$\sum_{i=1}^{n}(y_i - {\beta'}^\top x_i)^2 + \lambda\|\beta'\|_2^2 \leq \sum_{i=1}^{n}(y_i - {\beta'}^\top x_i)^2 + \lambda t < \sum_{i=1}^{n}(y_i - {\beta^*}^\top x_i)^2 + \lambda t = \sum_{i=1}^{n}(y_i - {\beta^*}^\top x_i)^2 + \lambda\|\beta^*\|_2^2$$

a contradiction, as β^* is an optimal solution to PROBLEM 1 $\square$

Ridge Regression: Intuition #1



PROBLEM 1

Minimize: $$\sum_{i=1}^{n}(y_i - \beta^\top x_i)^2 + \lambda\|\beta\|_2^2$$

subject to: $\beta \in \mathbb{R}^d$

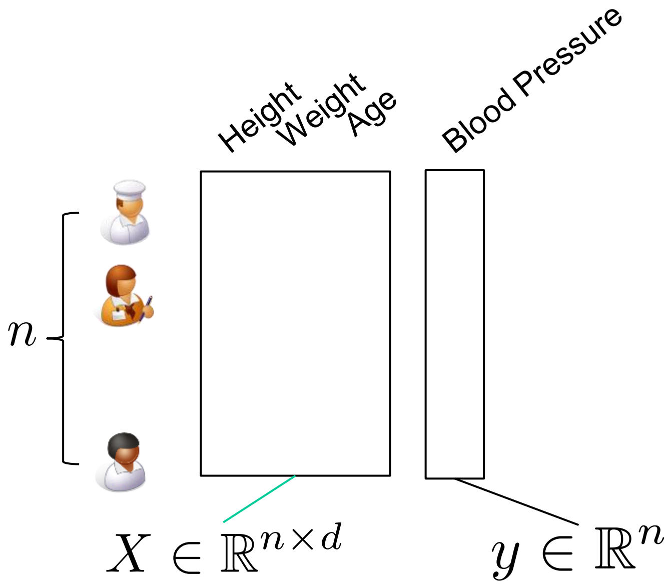
PROBLEM 2

Minimize: $$\sum_{i=1}^{n}(y_i - \beta^\top x_i)^2$$

subject to: $\|\beta\|_2^2 \leq t$

$$\lambda \Uparrow \quad \Rightarrow \quad t \Downarrow$$

Ridge Regression: Intuition #2



$$y_i = \beta^\top x_i + \varepsilon_i, \quad i = 1, \ldots, n$$

ε_i i.i.d., $\mathbb{E}[\varepsilon_i] = 0, \mathbb{E}[\varepsilon_i^2] = \sigma^2 < \infty$

- Suppose, in addition, that

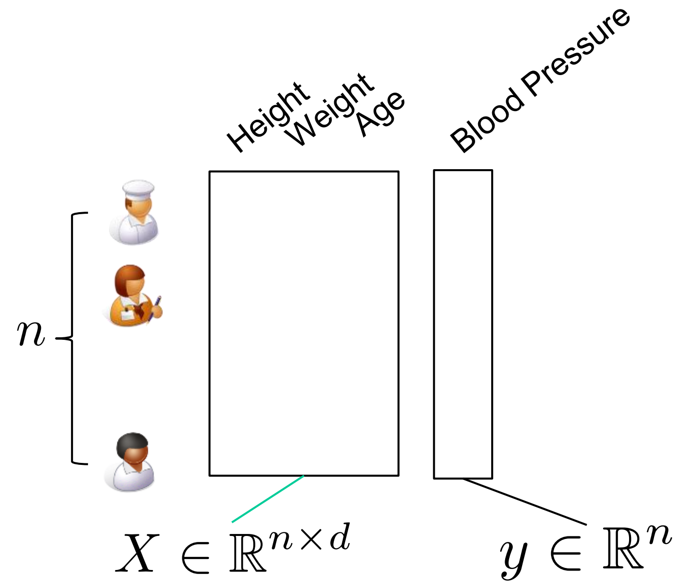
$$\varepsilon_i \sim N(0, \sigma^2)$$

Then, LSE is a **Maximum Likelihood Estimator**:

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \langle \beta, x_i \rangle)^2 = \arg\min_{\beta \in \mathbb{R}^d} \; -\log\left(P\left(y|\beta, X\right)\right)$$

$$= \arg\max_{\beta \in \mathbb{R}^d} \; P\left(y|\beta, X\right)$$

Ridge Regression: Intuition #2



$$y_i = \beta^\top x_i + \varepsilon_i, \quad i = 1, \ldots, n$$

ε_i i.i.d., $\mathbb{E}[\varepsilon_i] = 0, \mathbb{E}[\varepsilon_i^2] = \sigma^2 < \infty$

- Suppose, in addition, that

$$\varepsilon_i \sim N(0, \sigma^2)$$

and

$$\beta \sim N(0, \frac{\sigma^2}{\lambda} I)$$

Bayes prior

Then, Ridge Regression is a **Maximum A-Posteriori (MAP) Estimation**:

The bigger the λ the higher our "prior" belief on small $\|\beta\|_2$

$$\begin{aligned}\hat{\beta}^{\texttt{ridge}} = \arg\min_{\beta\in\mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda\|\beta\|_2^2 &= \arg\min_{\beta\in\mathbb{R}^d} - \log(P(y|\beta)) - \log P(\beta) \\ &= \arg\min_{\beta\in\mathbb{R}^d} - \log(P(y, \beta)) \\ &= \arg\max_{\beta\in\mathbb{R}^d} P(y, \beta) = \arg\max_{\beta\in\mathbb{R}^d} P(\beta|y)P(y) = \arg\max_{\beta\in\mathbb{R}^d} P(\beta|y)\end{aligned}$$

Towards Intuition #3

Gauss

Markov

Best **L**inear **U**nbiased **E**stimator

LSE has the "smallest" covariance among all linear unbiased estimators

What About Ridge Regression?

$$\hat{\beta}^{\texttt{ridge}} = \underset{\beta \in \mathbb{R}^d}{\arg\min} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_2^2$$
$$= \underbrace{(\lambda I + X^\top X)^{-1} X^\top}_{\text{This is a linear estimator!}} y$$

- ❑ Q: Does it have a smaller covariance than LSE?
- ❑ Yes! Because it is **biased!**

Bias of Ridge Regression

$$\hat{\beta}^{\texttt{ridge}} = (\lambda I + X^\top X)^{-1} X^\top y$$

$$\begin{aligned}
\mathbb{E}[\hat{\beta}^{\texttt{ridge}}] &= \mathbb{E}[(\lambda I + X^\top X)^{-1} X^\top y] = (\lambda I + X^\top X)^{-1} X^\top \mathbb{E}[y] \\
&= (\lambda I + X^\top X)^{-1} X^\top X \beta \\
&= (\lambda I + X^\top X)^{-1} X^\top X \beta + \lambda(\lambda I + X^\top X)^{-1}\beta - \lambda(\lambda I + X^\top X)^{-1}\beta \\
&= (\lambda I + X^\top X)^{-1}(X^\top X + \lambda I)\beta - \lambda(\lambda I + X^\top X)^{-1}\beta \\
&= \beta - \lambda(\lambda I + X^\top X)^{-1}\beta
\end{aligned}$$

So the **bias** or the ridge estimator is:

$$\mathtt{b} = \mathbb{E}[\hat{\beta}^{\texttt{ridge}}] - \beta = -\lambda(\lambda I + X^\top X)^{-1}\beta \neq 0$$

Covariance of Ridge Regression

$$\hat{\beta}^{\texttt{ridge}} = (\lambda I + X^\top X)^{-1} X^\top y$$

$$\mathbb{E}[\hat{\beta}^{\texttt{ridge}}] = \beta - \lambda(\lambda I + X^\top X)^{-1}\beta \neq \beta$$

$$\begin{aligned}
\texttt{Cov}\left(\hat{\beta}^{\texttt{ridge}}\right) &= \mathbb{E}\left[\left(\hat{\beta}^{\texttt{ridge}} - \mathbb{E}[\hat{\beta}^{\texttt{ridge}}]\right)\left(\hat{\beta}^{\texttt{ridge}} - \mathbb{E}[\hat{\beta}^{\texttt{ridge}}]\right)^\top\right] \\
&= \ldots \\
&= \sigma^2(\lambda I + X^\top X)^{-1} X^\top X (\lambda I + X^\top X)^{-1}
\end{aligned}$$

So What?

$$\hat{\beta}^{\texttt{ridge}} = (\lambda I + X^\top X)^{-1} X^\top y$$

$$\mathbb{E}[\hat{\beta}^{\texttt{ridge}}] = \beta - \lambda(\lambda I + X^\top X)^{-1}\beta \neq \beta$$

$$\texttt{Cov}(\hat{\beta}^{\texttt{ridge}}) = \sigma^2(\lambda I + X^\top X)^{-1} X^\top X (\lambda I + X^\top X)^{-1}$$

Recall that $X^\top X \succeq 0$

Let $0 \leq \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_d$ be its eigenvalues, and $e_i, i = 1, \ldots, d$ the corresponding eigenvectors.

Then:

$$X^\top X = \sum_{i=1}^{d} \lambda_i e_i e_i^\top \qquad \lambda I + X^\top X = \sum_{i=1}^{d} (\lambda + \lambda_i) e_i e_i^\top \succ X^\top X$$

Hence: $(\lambda I + X^\top X)^{-1} = \sum_{i=1}^{d} \frac{1}{\lambda + \lambda_i} e_i e_i^\top$

So What?

$$X^\top X = \sum_{i=1}^{d} \lambda_i e_i e_i^\top \qquad (\lambda I + X^\top X)^{-1} = \sum_{i=1}^{d} \frac{1}{\lambda + \lambda_i} e_i e_i^\top$$

$$\begin{aligned}
\texttt{Cov}(\hat{\beta}^{\texttt{ridge}}) &= \sigma^2 (\lambda I + X^\top X)^{-1} X^\top X (\lambda I + X^\top X)^{-1} \\
&= \sigma^2 \sum_{i=1}^{d} \frac{1}{\lambda + \lambda_i} e_i e_i^\top \cdot \sum_{i=1}^{d} \lambda_i e_i e_i^\top \cdot \sum_{i=1}^{d} \frac{1}{\lambda + \lambda_i} e_i e_i^\top \\
&= \sigma^2 \sum_{i=1}^{d} \frac{\lambda_i}{(\lambda + \lambda_i)^2} e_i e_i^\top \\
&= \sigma^2 \sum_{i=1}^{d} \frac{1}{\lambda_i + 2\lambda + \frac{\lambda^2}{\lambda_i}} e_i e_i^\top \\
&\prec \sigma^2 \sum_{i=1}^{d} \frac{1}{\lambda_i} e_i e_i^\top = (X^\top X)^{-1} = \texttt{Cov}(\hat{\beta}^{\texttt{LSE}})
\end{aligned}$$

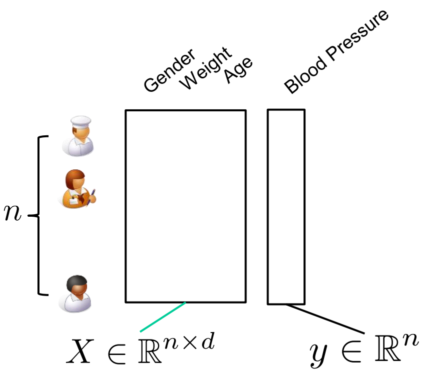
Opposite of What G-M predicts!

$$\mathtt{Cov}(\hat{\beta}^{\mathtt{ridge}}) \prec \mathtt{Cov}(\hat{\beta}^{\mathrm{LSE}})$$

No contradiction with G-M, as ridge estimator **is biased.**

$$\mathtt{Cov}(\hat{\beta}^{\mathtt{ridge}}) = \sigma^2 \sum_{i=1}^{d} \frac{\lambda_i}{(\lambda+\lambda_i)^2} e_i e_i^\top$$

$$\mathtt{b} = -\lambda(\lambda I + X^\top X)^{-1}\beta$$



Predicted Value: $\hat{y}_0 = \langle \hat{\beta}^{\mathtt{ridge}}, x_0 \rangle$

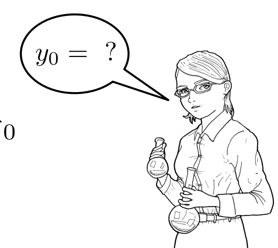
Expected Prediction Error:

$$\mathbb{E}[(y_0-\hat{y}_0)^2] = \mathbb{E}[(y_0-\beta^\top x_0)^2] + \mathbb{E}\left[\left({\beta^{\mathtt{ridge}}}^\top x_0 - \mathbb{E}[\beta^{\mathtt{ridge}}]^\top x_0\right)^2\right] + \left(\mathbb{E}[\beta^{\mathtt{ridge}}]^\top x_0 - \beta^\top x_0\right)^2$$

$$= \underbrace{\sigma^2}_{\text{inherent noise}} + \underbrace{x_0^\top \mathtt{Cov}(\hat{\beta}^{\mathtt{ridge}}) x_0}_{\text{variance}} + \underbrace{(\mathtt{b}^\top x_0)^2}_{\text{bias}}$$

$x_0 =$ [♀ | 90K | 24ys]

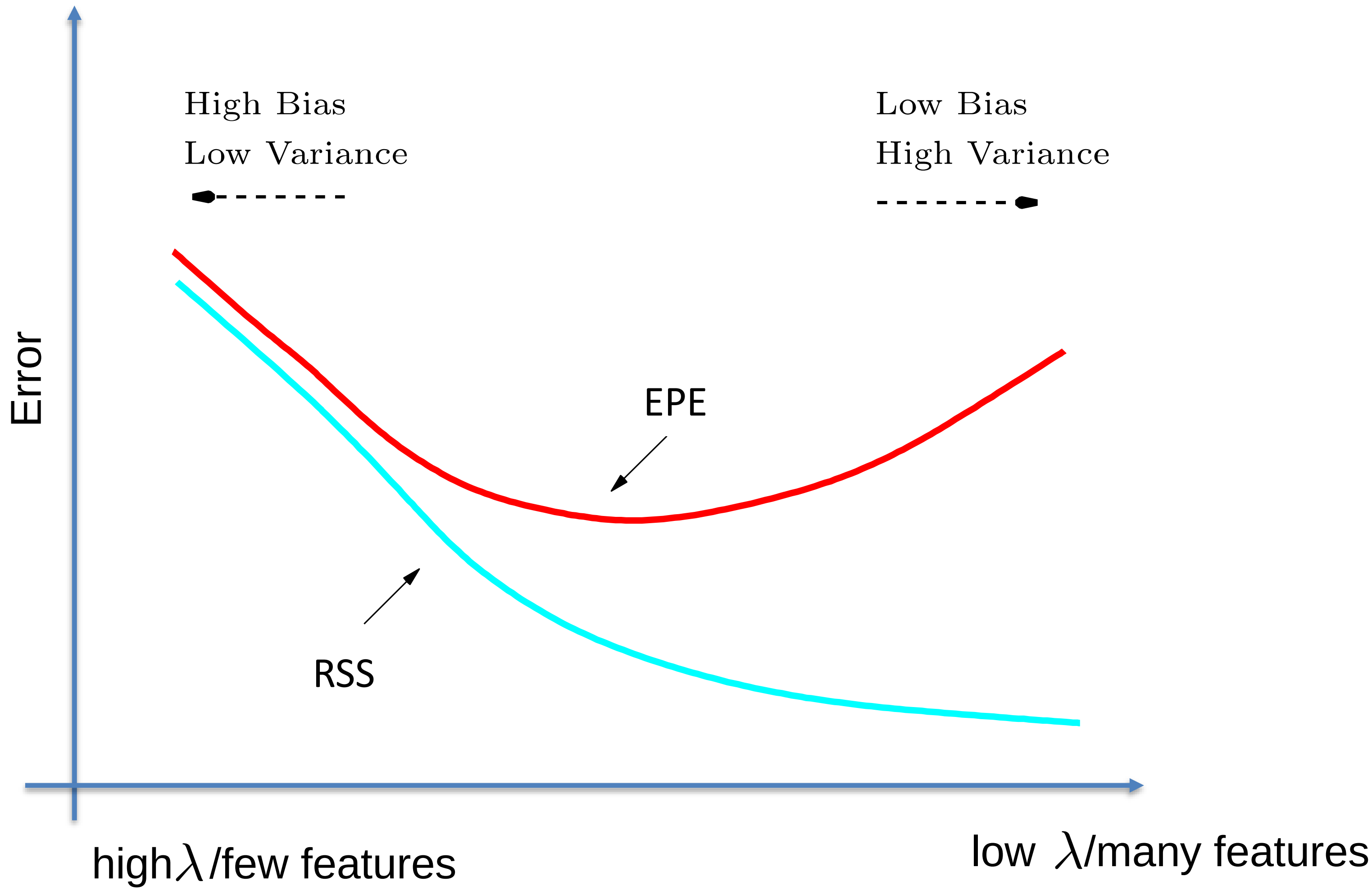
$y_0 = \beta^\top x_0 + \varepsilon_0$



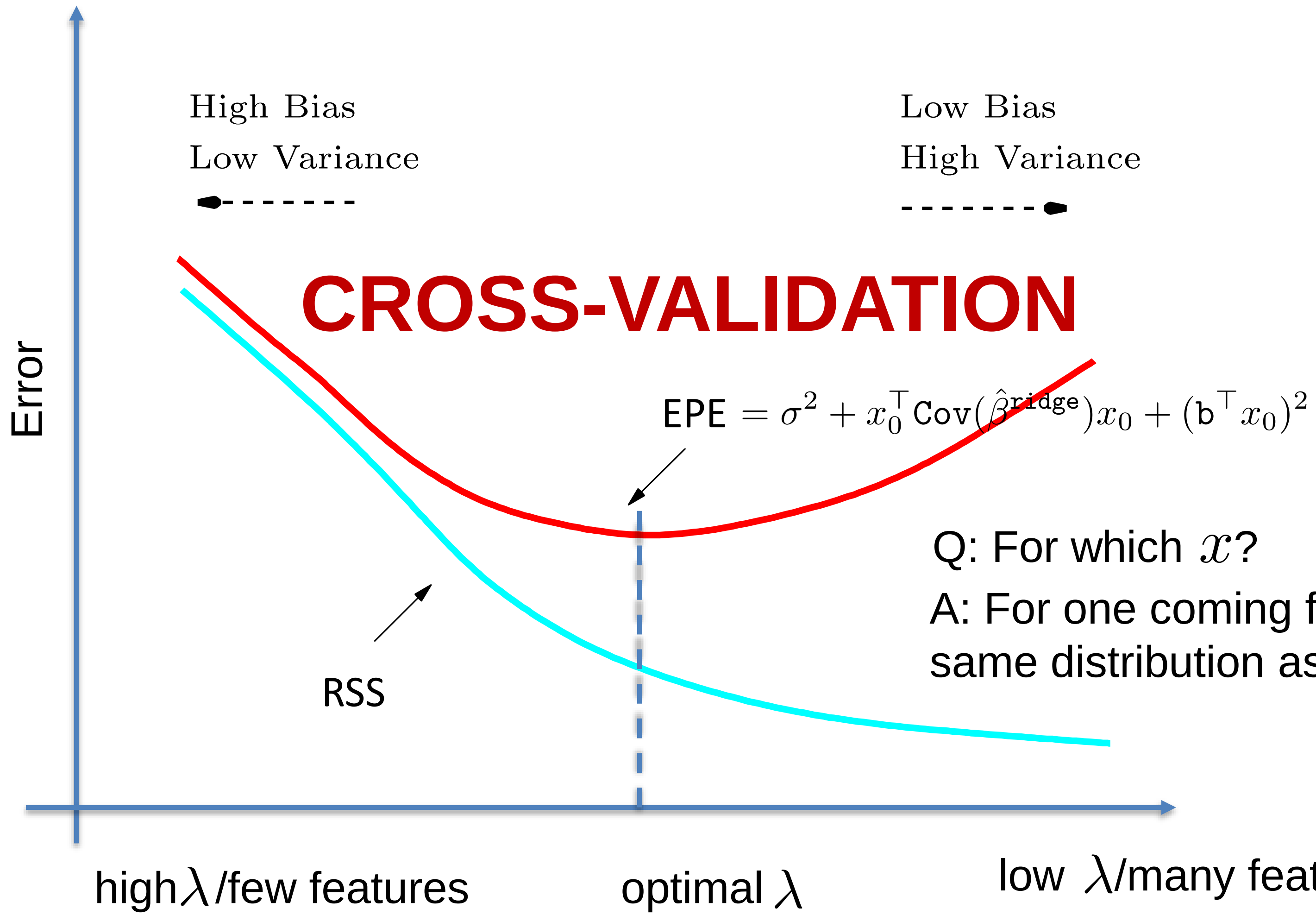
λ establishes a bias variance trade-off

Northeastern

Ridge Regression: Intuition #3



Ridge Regression: Intuition #3

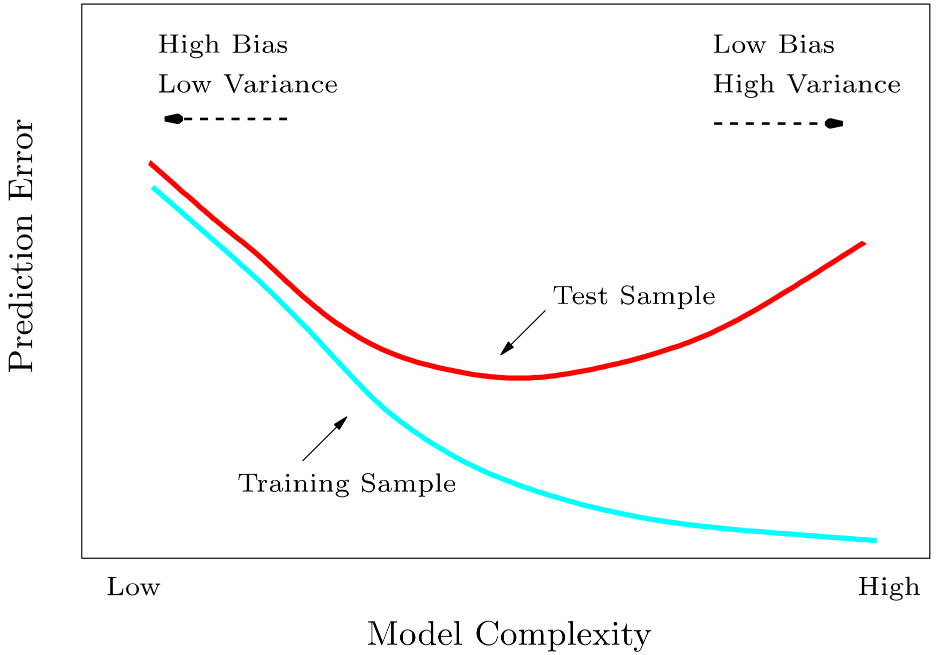


Q: For which x?

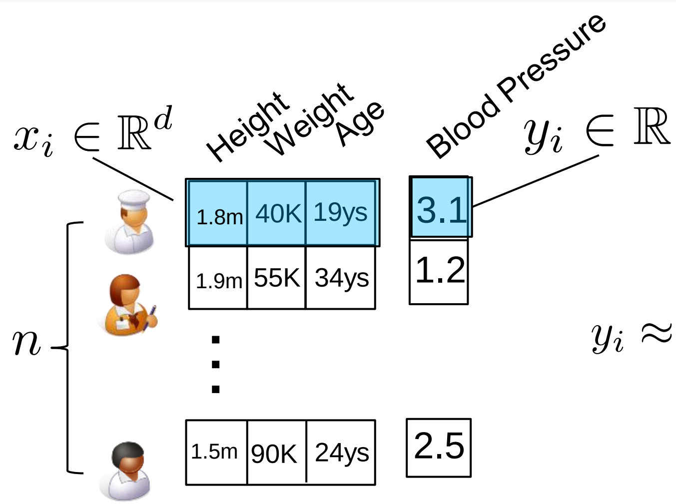
A: For one coming from same distribution as data

Intuition 3 Has a Universal, General Interpretation

- Cross-Validation minimizes EPE, assuming new data comes from **same distribution** as existing data
- When varying **model complexity,** we are establishing a tradeoff between **variance** and **bias**



Lasso Regression



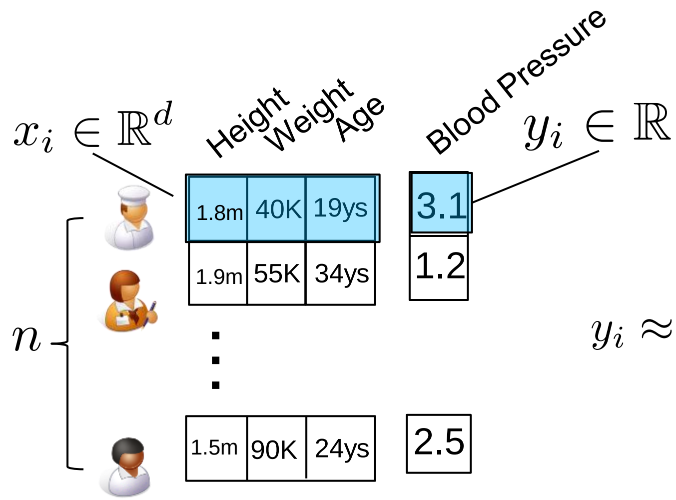
$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

$$\hat{\beta}^{\texttt{lasso}} = \arg\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_1, \text{ for some } \lambda > 0$$

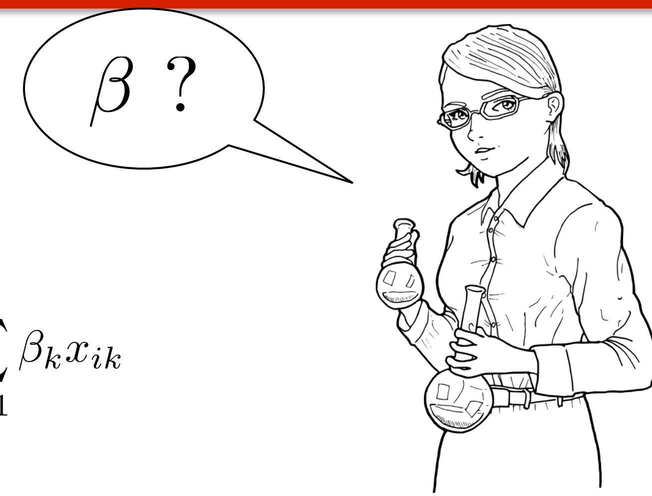
where $\|\beta\|_1 = \sum_{k=1}^{d} |\beta_k|$

Convex!
(not differentiable)

Lasso Regression



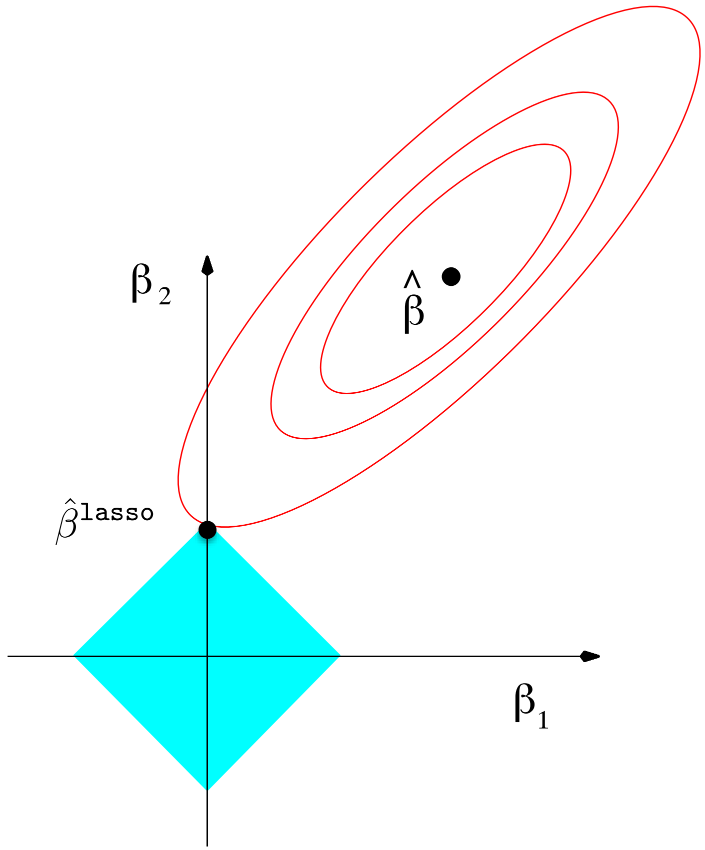
$$y_i \approx f(x_i) = \beta^\top x_i = \sum_{k=1}^{d} \beta_k x_{ik}$$

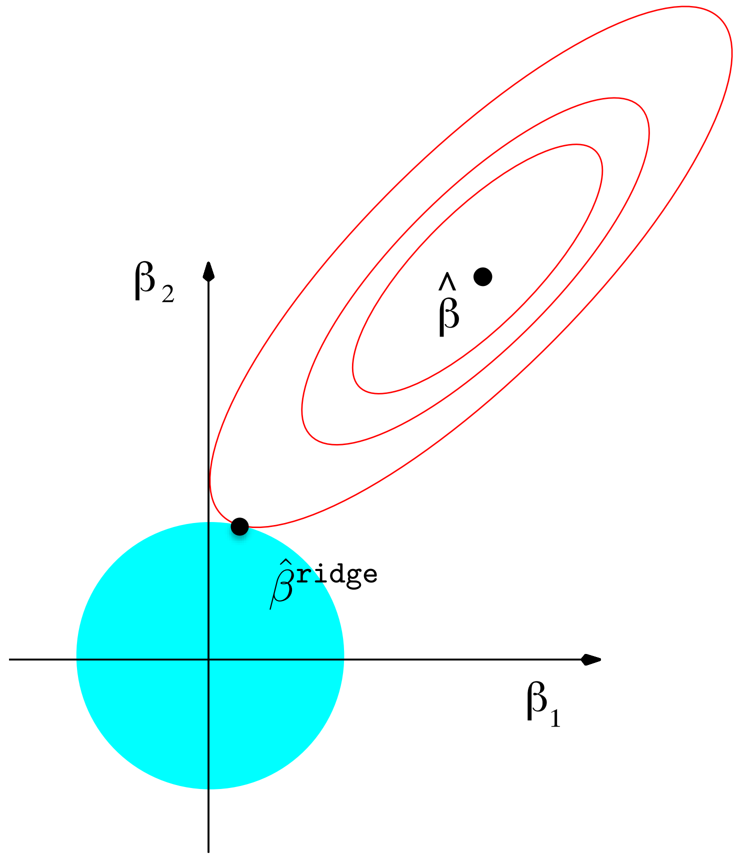


For every $\lambda \geq 0$, there exists a $t \geq 0$ such that the two problems produce **the same solution**

Minimize: $\sum_{i=1}^{n} (y_i - \beta^\top x_i)^2 + \lambda \|\beta\|_1$	Minimize: $\sum_{i=1}^{n} (y_i - \beta^\top x_i)^2$
subject to: $\beta \in \mathbb{R}^d$	subject to: $\|\beta\|_1 \leq t$

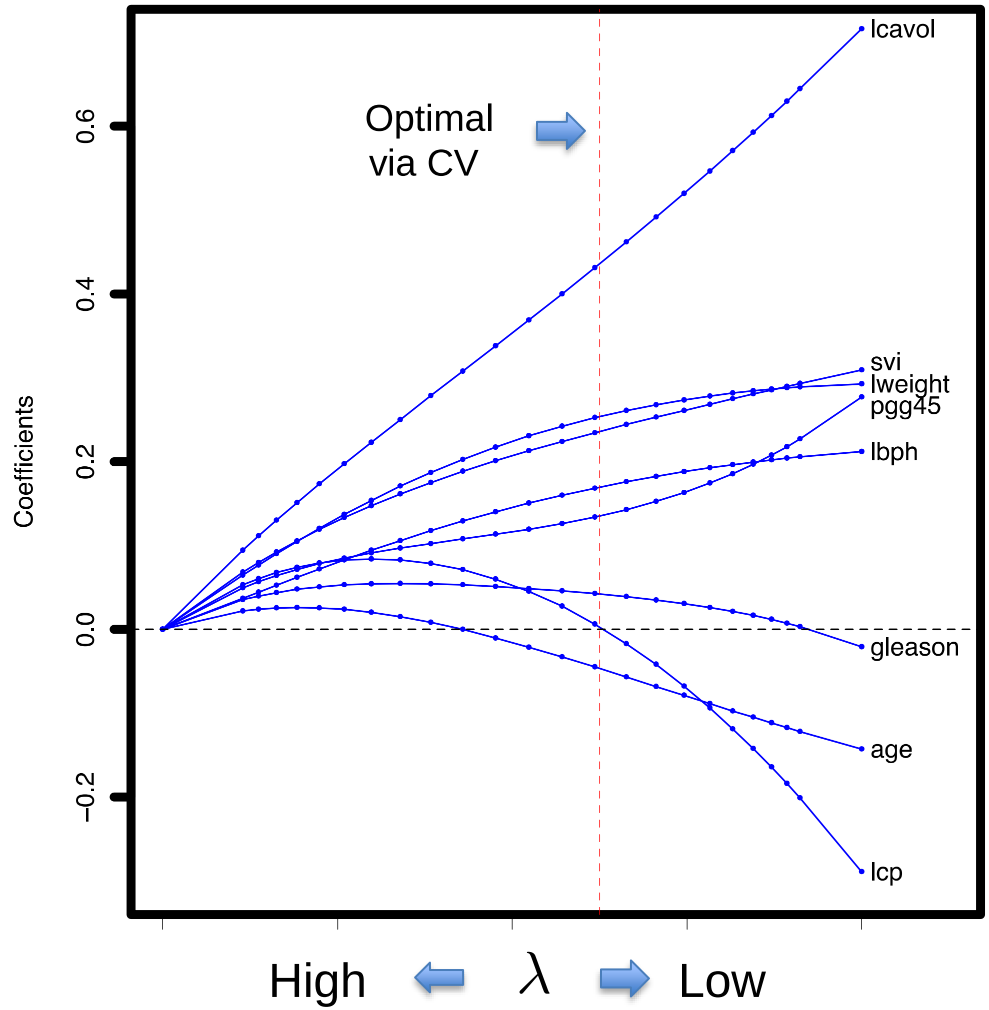
Lasso Regression vs. Ridge Regression





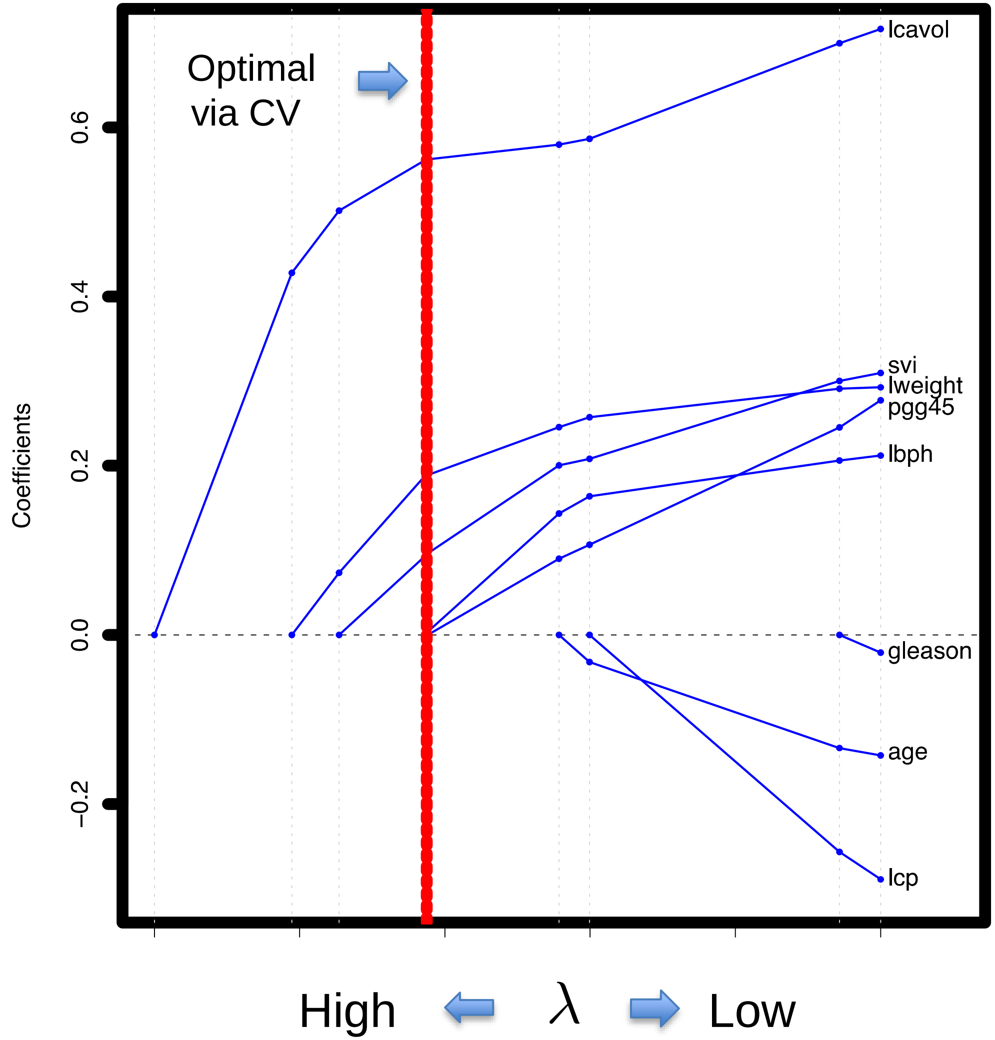
Lasso is more prone to sparse solutions

Varying λ in Ridge Regression



Northeastern

Varying λ in Lasso Regression



Lasso Regression via Constrained Optimization

Minimize: $$\sum_{i=1}^{n}(y_i - \beta^\top x_i)^2 + \lambda \sum_{k=1}^{d} |\beta_k|$$

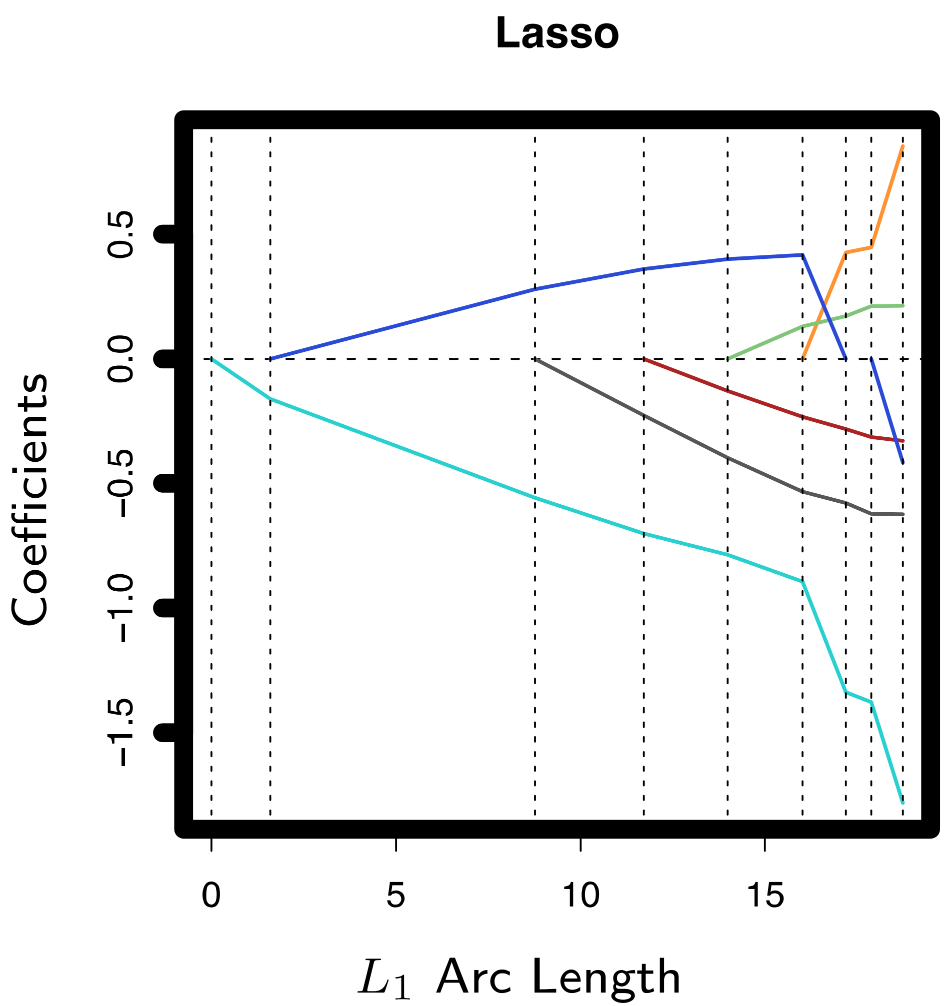
subject to: $$\beta \in \mathbb{R}^d$$

Lasso Regression via Constrained Optimization

Minimize:
$$\sum_{i=1}^{n}(y_i - \beta^\top x_i)^2 + \lambda \sum_{k=1}^{d} t_k$$

subject to:
$$\beta_k \leq t_k,$$
$$\beta_k \geq -t_k, \qquad \forall k = 1, \ldots, d$$

Least Angle Regression



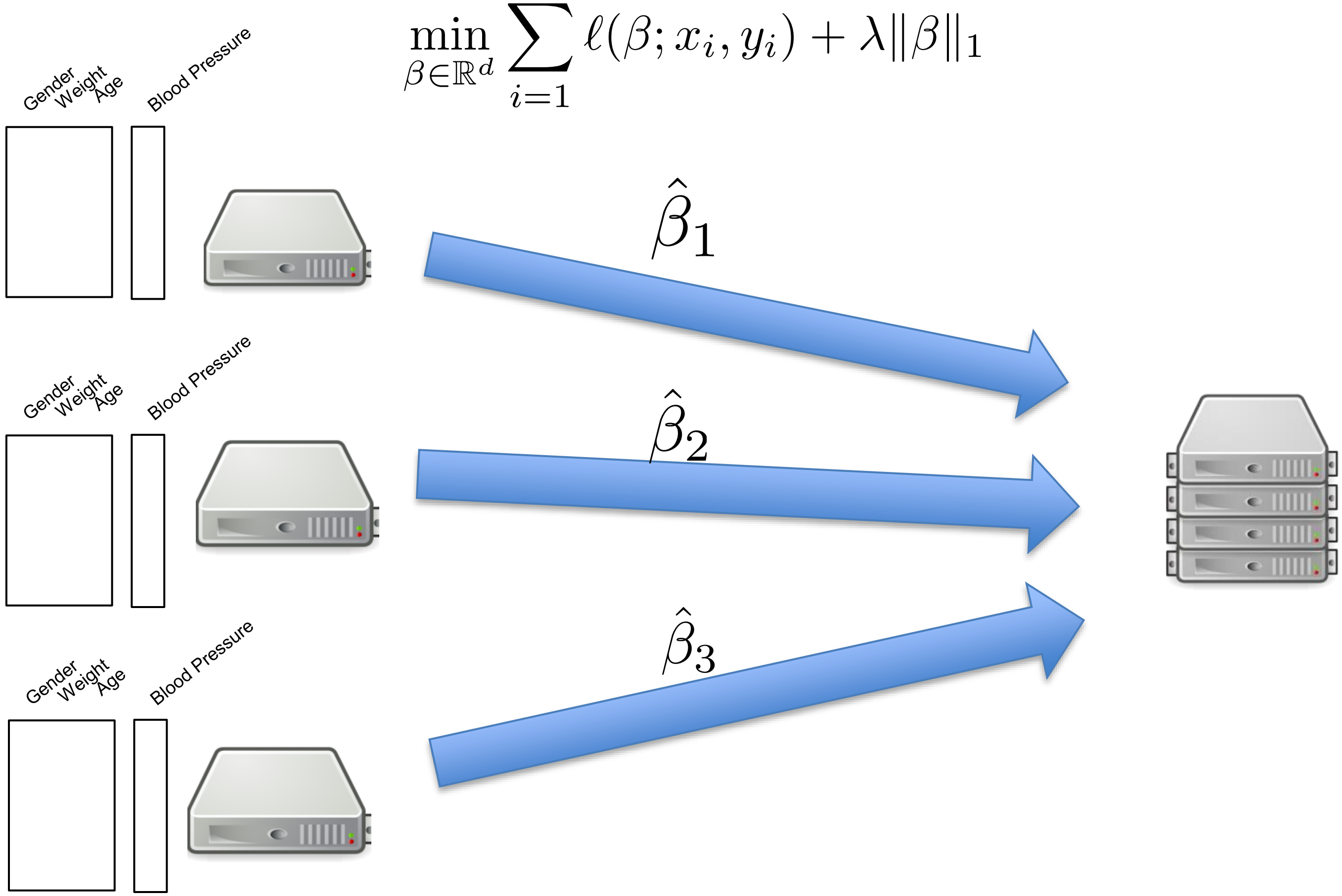
- ❑ Computes **entire path** under λ
  - ❑ needed anyway for CV
- ❑ Same complexity as standard linear regression
- ❑ Not that easy to parallelize

Northeastern

Alternating Directions Method of Multipliers

Distributed optimization and statistical learning via the alternating direction method of multipliers S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, 2011

$$\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} \ell(\beta; x_i, y_i) + \lambda \|\beta\|_1$$

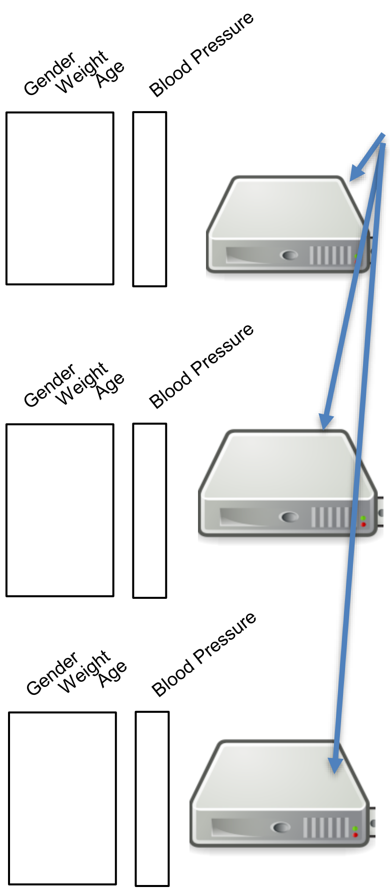


Alternating Directions Method of Multipliers

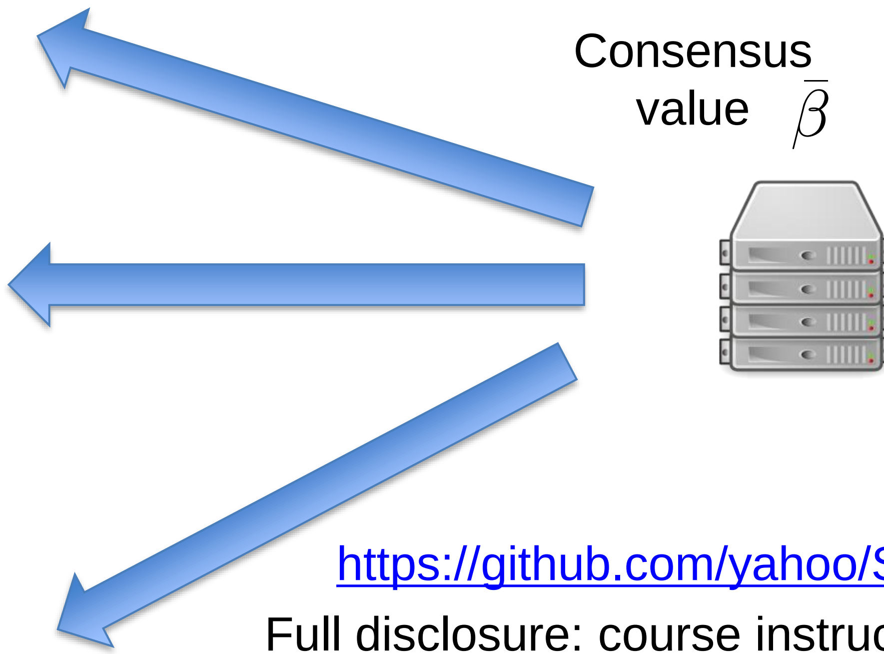
[Distributed optimization and statistical learning via the alternating direction method of multipliers](#) S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, 2011

$$\min_{\beta \in \mathbb{R}^d} \sum_{i=1}^{n} \ell(\beta; x_i, y_i) + \lambda \|\beta\|_1$$

Solve problem again, forcing agreement with $\bar{\beta}$



Consensus value $\bar{\beta}$



https://github.com/yahoo/SparkADMM

Full disclosure: course instructor involved

A Note on Biases and Regularization

"bias"

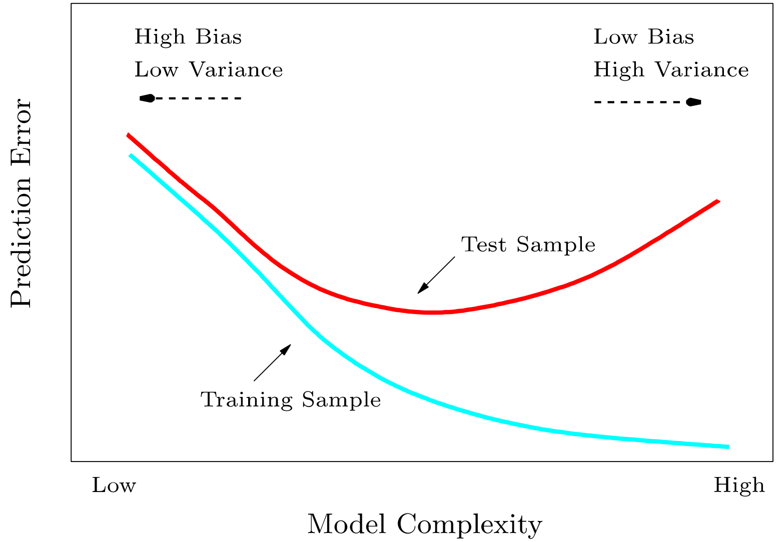
$$\min_{\beta \in \mathbb{R}^d, \beta_0 \in \mathbb{R}} \quad \sum_{i=1}^{n} \|y_i - \beta^\top x_i - \beta_0\|_2^2 + \lambda \|\beta\|_1$$

- Bias is typically **not** regularized
- Intuition: solution should be invariant to shifting the origin of either features or response

Conclusions

- ❑ Learning f ➡ learning right **features**

- ❑ Fitting is not enough! **Cross Validation**



- ❑ Model complexity ⬌ Bias-Variance Tradeoff

Northeastern

Conclusions: Relaxing $||\cdot||_0$