Algorithm: MBGD Linear Regression Learner (with constant stepsize)

input: $D = ((\vec{x}_1, y_1), \dots, (\vec{x}_n, y_n))$, η, T, b

$\vec{w} \leftarrow$ random vector in $\mathbb{R}^{d+1}$

$M \leftarrow \text{floor}\left(\frac{n}{b}\right)$

for $t = 1, \dots, T$

randomly shuffle D

for $m = 1, \dots, M$

$$\nabla \hat{L}(\vec{w}) \leftarrow \frac{2}{b} \sum_{i=mb+1}^{(m+1)b} (\vec{x}_i^T \vec{w} - y_i)\, \vec{x}_i$$

$$\vec{w} \leftarrow \vec{w} - \eta \nabla \hat{L}(\vec{w})$$

return $\hat{f}(\vec{x}) = \vec{x}^T \vec{w}$

Setting $b = n \Rightarrow M = 1 \Rightarrow$ "gradient descent"

$b = 1 \Rightarrow M = n \Rightarrow$ "stochastic gradient descent (SGD)"

in this course

in general $\rightarrow$ b is small

computation:

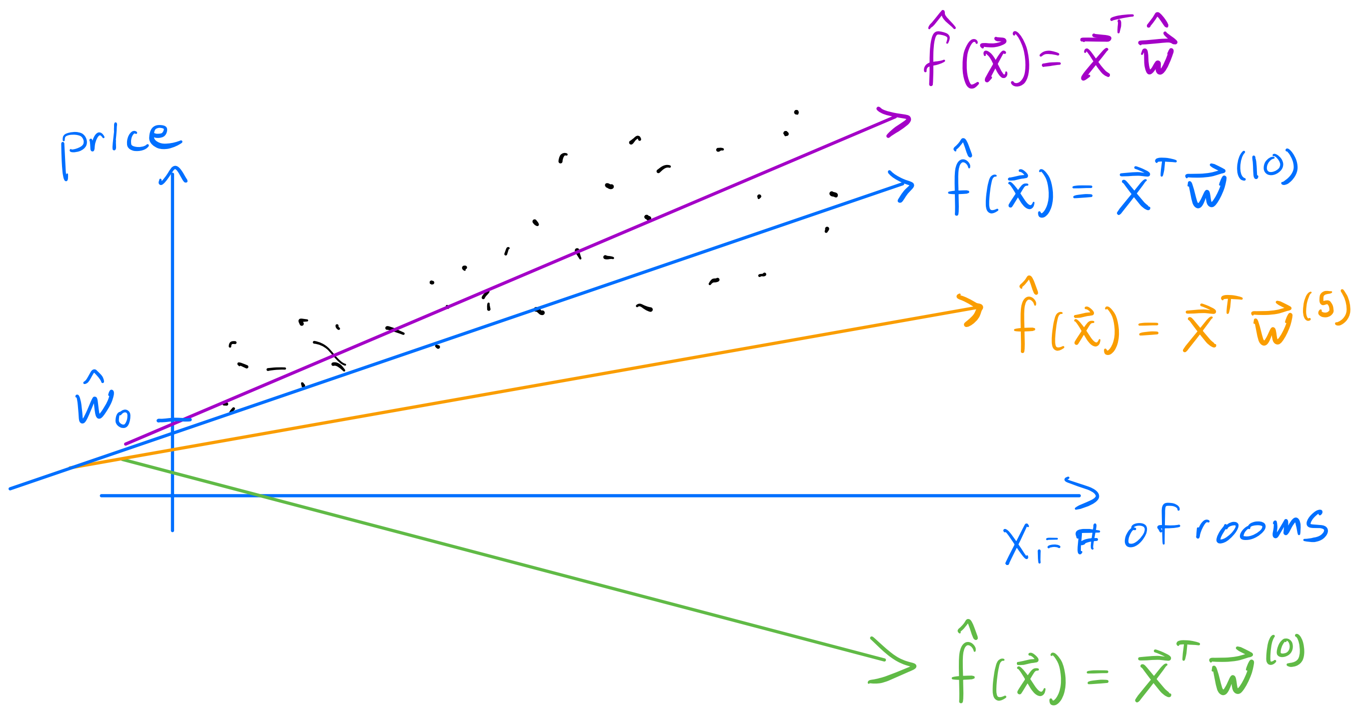
$$\text{BGD}: \quad O(\underbrace{d\,n}_{\nabla \hat{L}(\vec{w})}\,\overset{\#\text{epochs}}{T}) \qquad \text{MBGD}: \quad O(\underbrace{d\,\overset{\text{batch size}}{b}}_{\nabla \hat{L}(\vec{w})}\,\overset{\#\text{batches}}{M}\,\underset{\#\text{epochs}}{T})$$

For same T: computation identical for BGD & MGD

but

In practice, for the same T, MBGD usually finds better $\vec{w}$ (when $b < n$)

Ex:

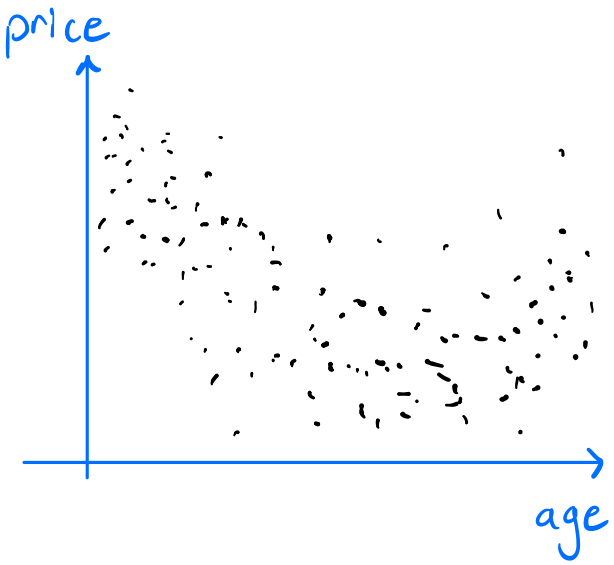


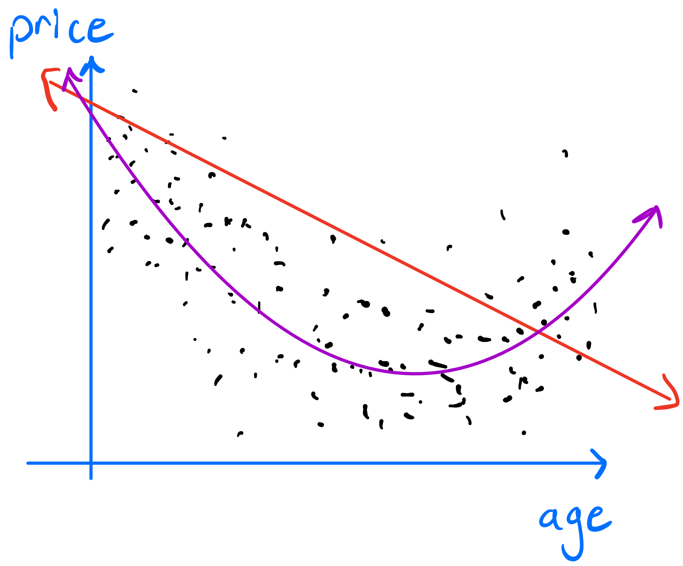
Should we always use a linear function class $\mathcal{F}$?

→ No, only if it fits your data

Polynomial Regression

$\underline{\underline{\text{Ex}}}$: Predicting car price based on age of the car





Let $\mathcal{F}$ contain polynomial functions instead of just linear functions

$\underline{\underline{\text{Ex}}}$: $d=1$, $\mathcal{X} = \mathbb{R}^{d+1} = \mathbb{R}^2$, $\mathcal{Y} = \mathbb{R}$

linear function: $\vec{x}^T w = (1, x_1)(w_0, w_1)^T = w_0 \overset{x_0=1}{\downarrow} + x_1 w_1$ $\qquad \vec{w} \in \mathbb{R}^2$

$$\mathcal{F}_1 = \{ f \mid f : \mathbb{R}^2 \to \mathbb{R} \text{ and } f(\vec{x}) = \vec{x}^T \vec{w} = w_0 + x_1 w_1 ,\ \vec{w} \in \mathbb{R}^2 \}$$

degree 2 polynomial: $w_0 \overset{x_0=1}{\downarrow} + x_1 w_1 + x_1^2 w_2$ $\qquad \vec{w} \in \mathbb{R}^3$

$= \phi_2(\vec{x})^T \vec{w}$ $\qquad$ linear in $\phi_2(\vec{x})$

"feature map"

where $\phi_2(\vec{x}) = (x_0 = 1, x_1, x_1^2)^T \in \mathbb{R}^3$ $\qquad$ transformed set of features

$$\mathcal{F}_2 = \{ f \mid f : \mathbb{R}^2 \to \mathbb{R} \text{ and } f(\vec{x}) = \phi_2(\vec{x})^T \vec{w} ,\ \vec{w} \in \mathbb{R}^3 \},\ \mathcal{F}_1 \subset \mathcal{F}_2$$

degree 3 polynomial: $w_0 + x_1 w_1 + x_1^2 w_2 + x_1^3 w_3 \qquad \vec{w} \in \mathbb{R}^4$

$= \phi_3(\vec{x})^T \vec{w}$ linear in $\phi_3(\vec{x})$

where $\phi_3(\vec{x}) = (1, x_1, x_1^2, x_1^3)^T \in \mathbb{R}^4$

$$\mathcal{F}_3 = \{ f \mid f : \mathbb{R}^2 \to \mathbb{R} \text{ and } f(\vec{x}) = \phi_3(\vec{x})^T \vec{w}, \vec{w} \in \mathbb{R}^4 \}$$

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \mathcal{F}_3$$

$\underline{\underline{Ex}}$: $d = 2$, $\mathcal{X} = \mathbb{R}^{d+1} = \mathbb{R}^3$, $\mathcal{Y} = \mathbb{R}$

linear function: $\vec{x}^T w = (\overset{x_0}{1}, x_1, x_2)(w_0, w_1, w_2)^T$

$= w_0 + x_1 w_1 + x_2 w_2 \qquad \vec{w} \in \mathbb{R}^3$

$$\mathcal{F}_1 = \{ f \mid f : \mathbb{R}^3 \to \mathbb{R} \text{ and } f(\vec{x}) = \vec{x}^T \vec{w}, \vec{w} \in \mathbb{R}^3 \}$$

degree 2 polynomial: $w_0 + x_1 w_1 + x_2 w_2 + x_1^2 w_3 + x_2^2 w_4 + x_1 x_2 w_5$

$= \phi_2(\vec{x})^T \vec{w}, \quad \vec{w} \in \mathbb{R}^6$ linear in $\phi_2(\vec{x})$

where $\phi_2(\vec{x}) = (x_0 = 1, x_1, x_2, x_1^2, x_2^2, x_1 x_2)^T \in \mathbb{R}^6$

$$\mathcal{F}_2 = \{ f \mid f : \mathbb{R}^3 \to \mathbb{R} \text{ and } f(\vec{x}) = \phi_2(\vec{x})^T \vec{w}, \vec{w} \in \mathbb{R}^6 \}$$

$$\mathcal{F}_1 \subset \mathcal{F}_2$$

In general $\phi_p : \mathcal{X} \to \mathcal{Z}$ is a degree p polynomial "feature map"

binomial coefficient $\binom{n}{k} = \frac{n!}{k!(n-k)!}$

For $\mathcal{X} = \mathbb{R}^{d+1}$, $\mathcal{Z} = \mathbb{R}^{\bar{p}}$ where $\bar{p} = \binom{(d+1)+p-1}{p} = \binom{d+p}{p}$

$\underline{\underline{\text{Ex}}}$: $d=2,\ p=2 \Rightarrow \bar{p} = \binom{2+2}{2} = \binom{4}{2} = \frac{4 \cdot 3}{2 \cdot 1} = 6$

$\underline{\underline{\text{Ex}}}$: $d=2,\ p=3 \Rightarrow \bar{p} = \binom{2+3}{2} = \binom{5}{2} = \frac{5 \cdot 4}{2} = 10$

The function class becomes:

$$\mathcal{F}_p = \{ f \mid f : \mathbb{R}^{d+1} \to \mathbb{R} \text{ and } f(\vec{x}) = \phi_p(\vec{x})^T \vec{w},\ \vec{w} \in \mathbb{R}^{\bar{p}} \}$$

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \dots \subset \mathcal{F}_p$$

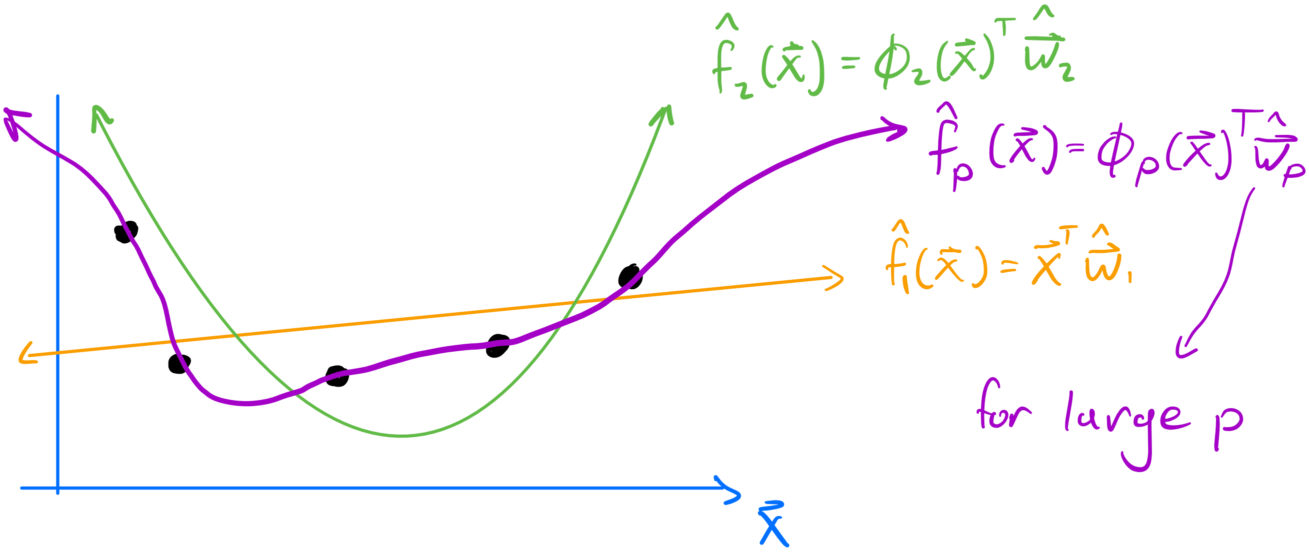
You can then solve for $\hat{\vec{w}}_p = \underset{\vec{w} \in \mathbb{R}^{\bar{p}}}{\arg\min}\ \hat{L}_p(\vec{w})$

where $\hat{L}_p(\vec{w}) = \frac{1}{n} \sum_{i=1}^{n} \ell(\phi_p(\vec{x}_i)^T \vec{w}, y_i)$

using the closed form solution

or gradient descent as before

Ex:



$$\hat{L}(\hat{f}_1) \geq \hat{L}(\hat{f}_2) \geq \cdots \geq \hat{L}(\hat{f}_p) \approx 0$$

Why not set p as large as possible?

what does $L(\hat{f}_p)$ look like?