Lecture 2: MLE for Gaussian and linear regression

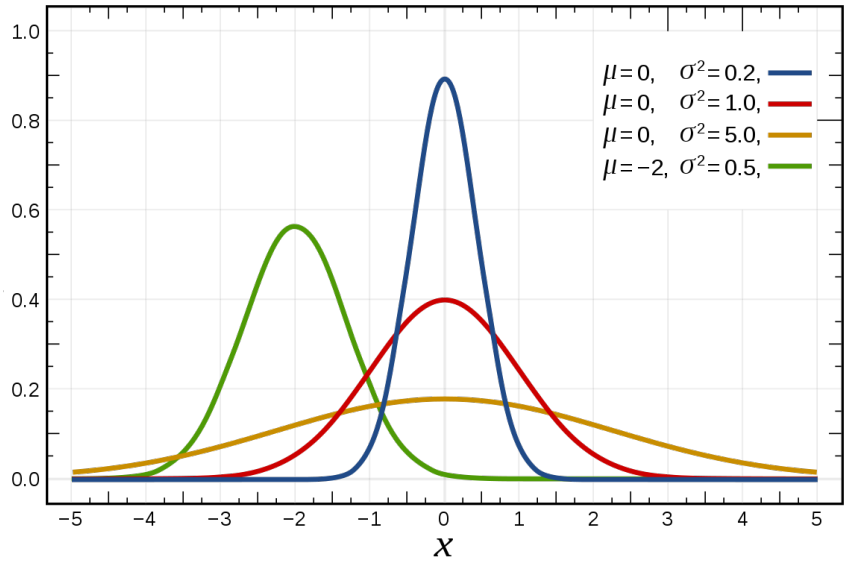
Recap: Maximum Likelihood Estimation

- **Observe** $X_1, X_2, \ldots, X_n$ drawn i.i.d. from $P(X_i; \theta)$ for some true $\theta = \theta^*$
- **Likelihood function**: $L_n(\theta) = \prod_{i=1}^{n} P(X_i; \theta)$
- **Log-likelihood function**: $\ell_n(\theta) = \log L_n(\theta) = \sum_{i=1}^{n} \log P(X_i; \theta)$
- **Maximum Likelihood Estimator (MLE)**: $\widehat{\theta}_{\text{MLE}} = \arg\max_{\theta} \ell_n(\theta)$

What about continuous variables?

- *Client*: What if I am measuring a **continuous variable**?
- ***You*: Let me tell you about Gaussians…**
  - A Gaussian random variable is written as $X \sim \mathcal{N}(\mu, \sigma^2)$ with mean $\mu \triangleq \mathbb{E}[X]$ and variance $\sigma^2 \triangleq \mathbb{E}\left[(X - \mathbb{E}[X])^2\right]$
  - The p.d.f. (Probability Density Function) of X is

$$P(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



Some properties of Gaussians

- affine transformation
  (multiplying by scalar and adding a constant)
  - $X \sim \mathcal{N}(\mu, \sigma^2)$
  - $Y = aX + b \implies Y \sim \mathcal{N}(a\mu + b, a^2\sigma^2)$

- Sum of Gaussians
  - $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$
  - $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$
  - $Z = X + Y \implies Z \sim \mathcal{N}(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2)$

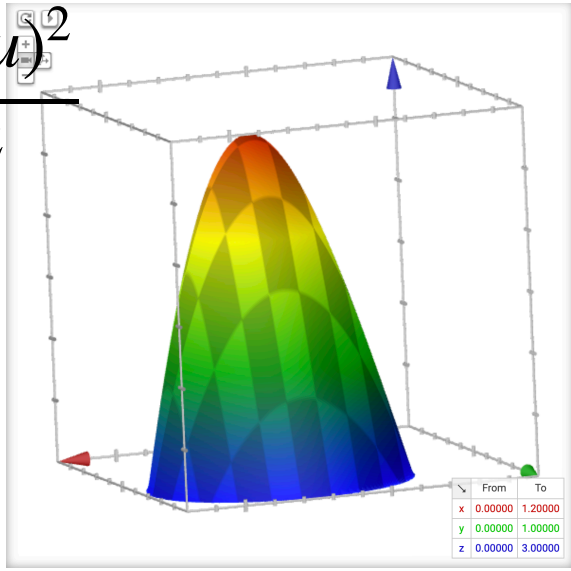
MLE for Gaussian

- **Hypothesis**: i.i.d. samples $\mathcal{D} = \{x_1, x_2, \ldots, x_n\}$ from $\mathcal{N}(\mu, \sigma^2)$

$$\begin{aligned} P(\mathcal{D}; \mu, \sigma^2) &= P(x_1, \ldots, x_n; \mu, \sigma^2) \\ &= P(x_1; \mu, \sigma^2) \times P(x_2; \mu, \sigma^2) \times \cdots P(x_n; \mu, \sigma^2) \\ &= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i - \mu)^2}{2\sigma^2}} \end{aligned}$$

- **Log-likelihood** of data:

$$\log P(\mathcal{D}; \mu, \sigma^2) = -n \log(\sigma\sqrt{2\pi}) - \sum_{i=1}^{n} \frac{(x_i - \mu)^2}{2\sigma^2}$$

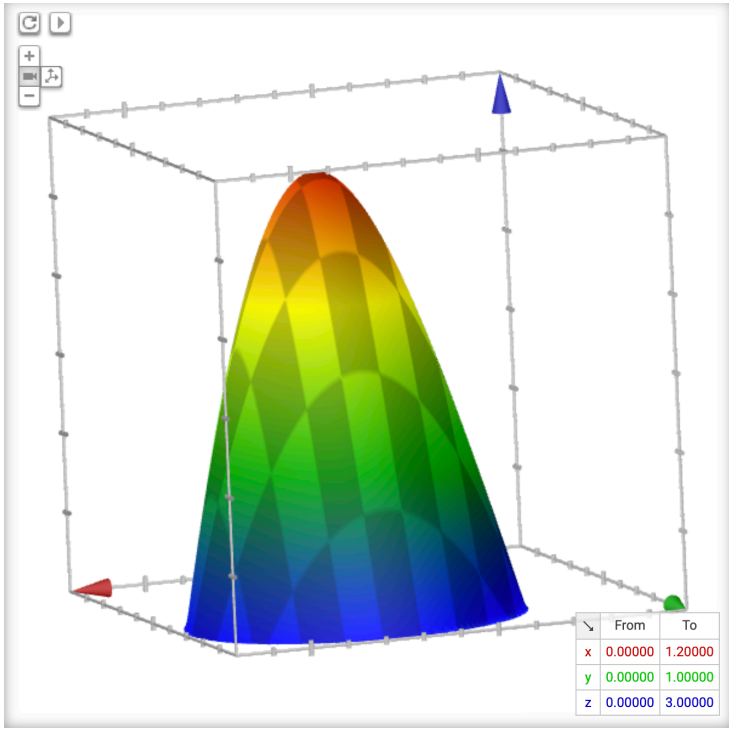


- What is $\widehat{\theta}_{\text{MLE}}$ for $\theta = (\mu, \sigma^2)$?

Your second learning algorithm: MLE for mean of a Gaussian

- What's MLE for mean?

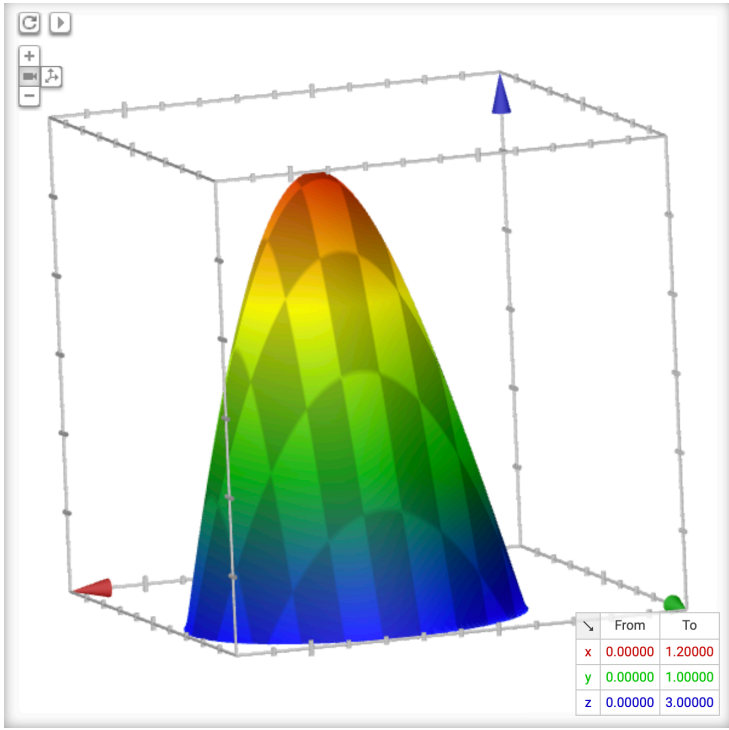
$$\frac{d}{d\mu}\log P(\mathcal{D};\mu,\sigma^2) = \frac{d}{d\mu}\left[-n\log(\sigma\sqrt{2\pi}) - \sum_{i=1}^{n}\frac{(x_i-\mu)^2}{2\sigma^2}\right]$$



MLE for variance

- Again, set derivative to zero:

$$\frac{d}{d\sigma}\log P(\mathcal{D};\mu,\sigma^2) = \frac{d}{d\sigma}\left[-n\log(\sigma\sqrt{2\pi}) - \sum_{i=1}^{n}\frac{(x_i-\mu)^2}{2\sigma^2}\right]$$



What can we say about the MLE?

- MLE:
  - $\widehat{\mu}_{\text{MLE}} = \frac{1}{n}\sum_{i=1}^{n} x_i$
  - $\widehat{\sigma}^2_{\text{MLE}} = \frac{1}{n}\sum_{i=1}^{n} (x_i - \widehat{\mu}_{\text{MLE}})^2$

- MLE for the mean of a Gaussian is **unbiased**
- MLE for the variance of a Gaussian is **biased**
  - $\mathbb{E}[\widehat{\sigma}^2_{\text{MLE}}] \neq \sigma^2$

- Unbiased variance estimator:
  - $\widehat{\sigma}^2_{\text{unbiased}} = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \widehat{\mu}_{\text{MLE}})^2$

Maximum Likelihood Estimation

- **Observe** $X_1, X_2, \ldots, X_n$ drawn i.i.d. from $P(X_i; \theta)$ for some true $\theta = \theta^*$
- **Likelihood function**: $L_n(\theta) = \prod_{i=1}^{n} P(X_i; \theta)$
- **Log-likelihood function**: $\ell_n(\theta) = \log L_n(\theta) = \sum_{i=1}^{n} \log P(X_i; \theta)$
- **Maximum Likelihood Estimator (MLE)**: $\widehat{\theta}_{\text{MLE}} = \arg\max_{\theta} \ell_n(\theta)$

Properties (under benign regularity conditions—smoothness, identifiability, etc.):

- Asymptotically consistent and normal: $\frac{\widehat{\theta}_{MLE} - \theta_*}{\widehat{se}} \sim \mathcal{N}(0, 1)$
- Asymptotic Optimality, minimum variance (see Cramer-Rao lower bound)

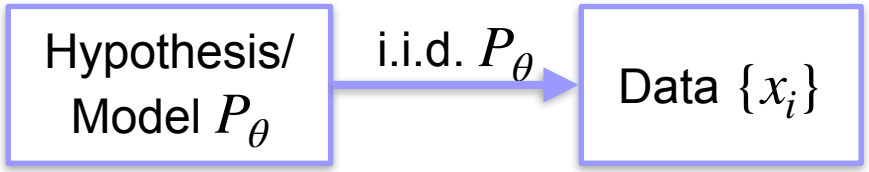
Recap

- Learning is…
  - Collect some data
    - E.g., coin flips

Data $\{x_i\}$

Recap

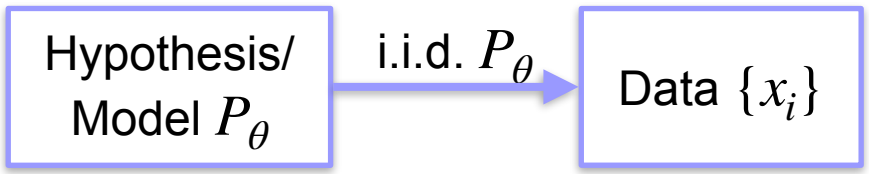
- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial



Recap

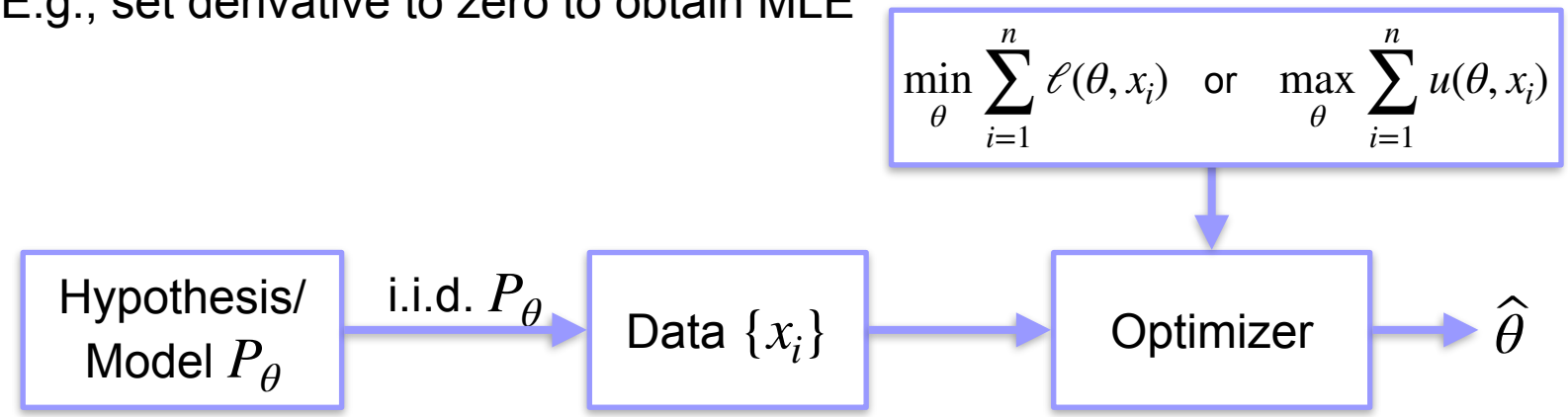
- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial
  - Choose a loss function
    - E.g., data likelihood

$$\min_{\theta} \sum_{i=1}^{n} \ell(\theta, x_i) \quad \text{or} \quad \max_{\theta} \sum_{i=1}^{n} u(\theta, x_i)$$



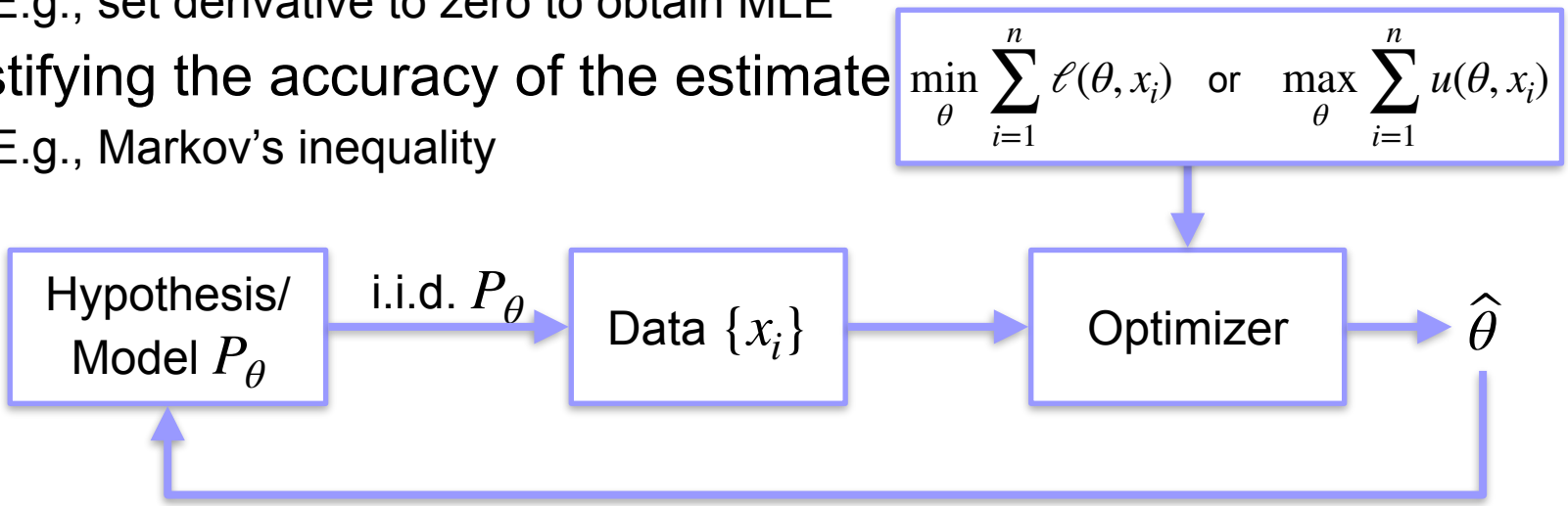
Recap

- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial
  - Choose a loss function
    - E.g., data likelihood
  - Choose an optimization procedure
    - E.g., set derivative to zero to obtain MLE



Recap

- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial
  - Choose a loss function
    - E.g., data likelihood
  - Choose an optimization procedure
    - E.g., set derivative to zero to obtain MLE
  - Justifying the accuracy of the estimate
    - E.g., Markov's inequality



Linear Regression

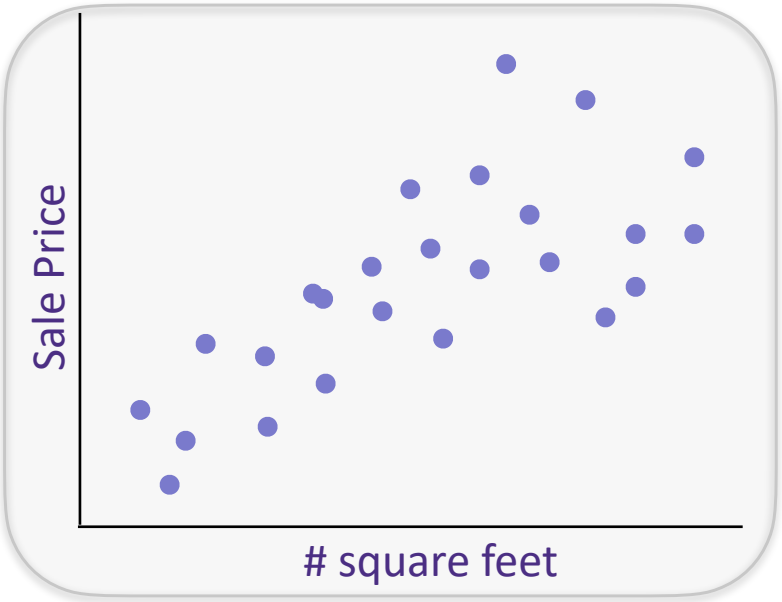
The regression problem, 1-dimensional

You want to sell your house that is 2,500 sq.ft.

Q. What is the right price?

Collect past sales data on zillow.com:

y = **House sale price** *and* x = **{# sq. ft.}**



Training Data: $\{(x_i, y_i)\}_{i=1}^{n}$ $x_i \in \mathbb{R}$ $y_i \in \mathbb{R}$

Process

1. Decide on a **model/hypothesis class**

assume house sale price is a linear function of square feet.

2. Find the function/model/hypothesis which explains/fits the data best

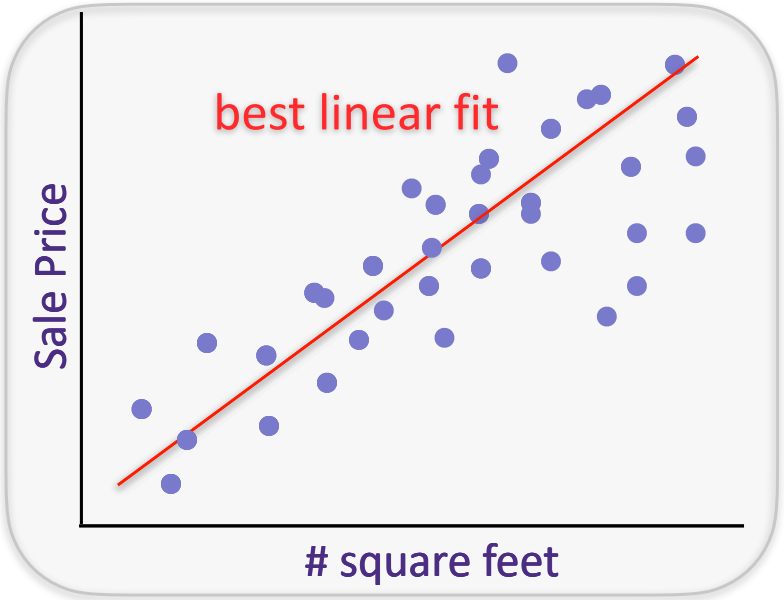
3. Use function to make prediction on new examples
   How much should you put your house on the market?

Fit a function to our data, 1-dimension

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft.}**



1. Training Data:

$$\{(x_i, y_i)\}_{i=1}^{n} \qquad x_i \in \mathbb{R} \quad y_i \in \mathbb{R}$$

2. Hypothesis/Model: linear

$$y_i = w \cdot x_i + \epsilon_i$$

3. Measure of good fit: ℓ_2-loss

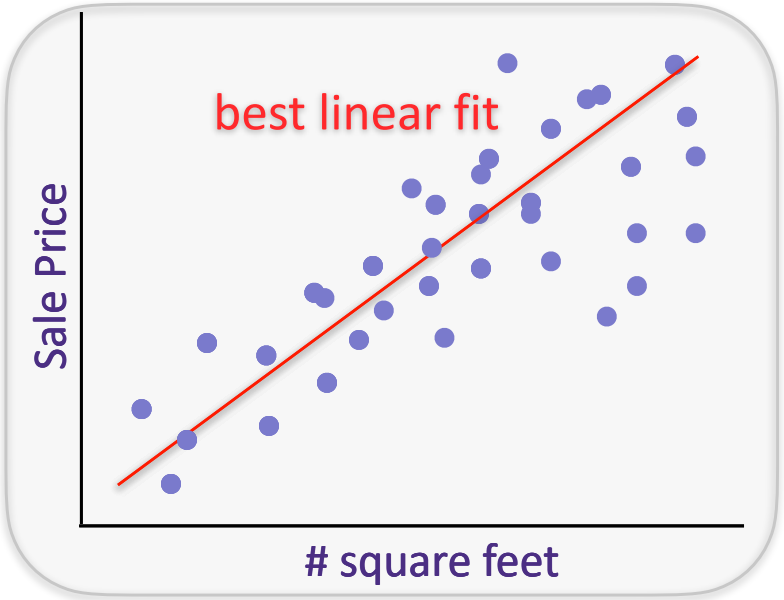
$$\min_{w \in \mathbb{R}} \sum_{i=1}^{n} (y_i - w x_i)^2 = \sum_{i=1}^{n} \varepsilon_i^2$$

The regression problem, d-dimensions

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft., zip code, date of sale, etc.}**



1. Training Data: $\{(x_i, y_i)\}_{i=1}^n$ $\quad x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$

2. Hypothesis/Model: linear

$$y_i = w^T x_i + \epsilon_i$$

3. Measure of good fit: ℓ_2-loss

$$\min_{w \in \mathbb{R}^d} \sum_{i=1}^n (y_i - w^T x_i)^2 = \sum_{i=1}^n \varepsilon_i^2$$

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input

n : # of examples/datapoints

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input

n : # of examples/datapoints

Linear Model:

$$\begin{aligned} y_1 &= x_1^T w + \epsilon_1 \\ y_2 &= x_2^T w + \epsilon_2 \\ &\vdots \\ y_n &= x_n^T w + \epsilon_n \end{aligned}$$

$$\mathbf{y} = \mathbf{X}w + \epsilon$$

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input
n : # of examples/datapoints

Linear Model:

$$\begin{aligned} y_1 &= x_1^T w + \epsilon_1 \\ y_2 &= x_2^T w + \epsilon_2 \\ &\vdots \\ y_n &= x_n^T w + \epsilon_n \end{aligned}$$

$$\mathbf{y} = \mathbf{X}w + \epsilon$$

ℓ_2-norm of a vector:
(also known as Euclidean norm)

$$\|\epsilon\|_2 = \sqrt{\epsilon_1^2 + \epsilon_2^2 + \cdots + \epsilon_d^2}$$

it follows that

$$\sum_{i=1}^{d} \epsilon_i^2 = \|\epsilon\|_2^2 = \epsilon^T \epsilon$$

ℓ_2-Loss: $\widehat{w}_{\text{LS}} = \arg\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - x_i^T w)^2$

this is also known as **Least Squares** solution

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input

n : # of examples/datapoints

Linear Model:

$$\begin{aligned} y_1 &= x_1^T w + \epsilon_1 \\ y_2 &= x_2^T w + \epsilon_2 \\ &\vdots \\ y_n &= x_n^T w + \epsilon_n \end{aligned} \qquad \mathbf{y} = \mathbf{X}w + \epsilon$$

ℓ_2-norm of a vector:

$$\|\epsilon\|_2 = \sqrt{\epsilon_1^2 + \epsilon_2^2 + \cdots + \epsilon_d^2}$$

it follows that

$$\sum_{i=1}^{d} \epsilon_i^2 = \|\epsilon\|_2^2 = \epsilon^T \epsilon$$

ℓ_2-Loss:

$$\widehat{w}_{\text{LS}} = \arg\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - x_i^T w)^2 = \arg\min_{w} \|\mathbf{y} - \mathbf{X}w\|_2^2$$

$$= \arg\min_{w} (\mathbf{y} - \mathbf{X}w)^T (\mathbf{y} - \mathbf{X}w)$$

The regression problem in matrix notation

$$\widehat{w}_{\text{LS}} = \arg\min_{w \in \mathbb{R}^d} (\mathbf{y} - \mathbf{X}w)^T (\mathbf{y} - \mathbf{X}w)$$

Set gradient w.r.t. w to zero to find the minima:

A few reminders on vector calculus

- Gradient of a function:

$$\nabla_w f(w) = \begin{bmatrix} \frac{df(w)}{dw_1} \\ \frac{df(w)}{dw_2} \\ \vdots \\ \frac{df(w)}{dw_d} \end{bmatrix}$$

- Example:

$f(w) = w^T w \implies \nabla_W f(w) = 2w$

$f(w) = (Aw)^T (Aw) \implies \nabla_W f(w) = 2AA^T w$

$f(w) = (Aw + b)^T (Aw + b) \implies \nabla_W f(w) = 2A^T (Aw + b)$

The regression problem in matrix notation

$$\begin{aligned}\widehat{w}_{LS} &= \arg\min_{w} ||\mathbf{y} - \mathbf{X}w||_2^2 \\ &= \arg\min_{w} (\mathbf{y} - \mathbf{X}w)^T(\mathbf{y} - \mathbf{X}w) \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}\end{aligned}$$

"Closed form" solution!

Questions?