Linear Regression:

- Classification: $(x, f(x))$ $\quad f(x) \in \{0, 1\}$
  - Half-spaces
  - Decision trees
- Real-valued labels (x, y) $y \in \mathbb{R}$.

X and Y two random variables

we want to predict the value/label.

we get to see X.

- we want to predict Y; we don't see x.

  $(x, y) \sim D$.

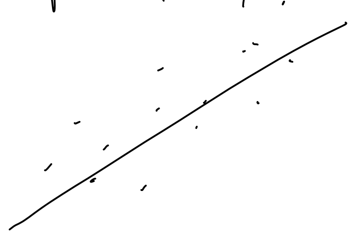
  - Optimal guess for y is $E[Y]$.
  - Measure loss by using square-loss: $(\text{prediction} - Y)^2$
- We observe X, we want to predict Y.

  optimal prediction $E[Y \mid X] = \underbrace{f(X)}_{\text{Regression Function}}$

  - Obstacle: $f(x)$ could be unknown, or hard to compute.

<u>Linear Regression asks the following question:</u>

☆ Given X, what linear function of X should we use to predict Y?



- we want to learn coefficients β_0 and β_1 such that

$$\underset{(x,y) \sim D}{E}\left[\left(Y - \underbrace{(\beta_0 + \beta_1 X)}_{Y' = \text{prediction}}\right)^2\right] \text{ is minimized.}$$

Draw a training set of size m :

$(x^1, y^1), \ldots, (x^m, y^m)$ "simple linear regression"

$$\underbrace{\min_{\beta_0, \beta_1} \frac{1}{m} \sum_{j=1}^{m} \left(y^j - (\beta_0 + \beta_1 x^j) \right)^2}_{\mathcal{L} \, = \text{ cost function.}}$$

How to find β_0 and β_1?

Take derivative wrt β_0, β_1. Set them equal to 0.

$$\frac{\partial \mathcal{L}}{\partial \beta_0} = \frac{1}{m} \sum_{j=1}^{m} \left(y^j - \beta_0 - \beta_1 x^j \right)(2)(-1) = 0$$

$$\frac{\partial \mathcal{L}}{\partial \beta_1} = \frac{1}{m} \sum_{j=1}^{m} \left(y^j - \beta_0 - \beta_1 x^j \right)(2)(-x^j) = 0$$

Eliminating -2

$$\frac{1}{m} \sum_{j=1}^{m} \left(y^j - \beta_0 - \beta_1 x^j \right) = 0 \Rightarrow \bar{y} - \frac{\cancel{m}\beta_0}{\cancel{m}} \beta_1 \bar{x} = 0$$

$\Rightarrow \beta_0$ in terms of β_1, $\bar{y}$, $\bar{x}$: $\boxed{\beta_0 = \bar{y} - \beta_1 \bar{x}}$

$$\frac{1}{m} \sum_{j=1}^{m} \left(y^j - \beta_0 - \beta_1 x^j \right)(x^j) = 0 \Rightarrow$$

$$\frac{1}{m} \sum_{j=1}^{m} \left(x_j y_j - \beta_0 x_j - \beta_1 (x^j)^2 \right)$$

β_1 will not involve β_0 : substitute $\beta_0 = \bar{y} - \beta_1 \bar{x}$:

$$\overline{xy} - \beta_0 \bar{x} - \beta_1 \overline{x^2} = \overline{xy} - (\bar{y} - \beta_1 \bar{x})\bar{x} - \beta_1 \overline{x^2}$$

$$\overline{xy} - \bar{x}\bar{y} + \beta_1 (\bar{x})^2 - \beta_1 \overline{x^2} = 0$$

$$\beta_1 = \frac{\overline{xy} - \bar{x} \cdot \bar{y}}{\overline{x^2} - (\bar{x})^2} = \boxed{\frac{cov(x,y)}{Var(x)}}$$

where, $cov(x,y) = E[X \cdot Y] - E[X]E[Y]$

This is the simple linear regression. Now let's look at multiple variables/features, $x_1, \ldots, x_n$

$x \in \mathbb{R}^n$. x is n dimensional vector.

$y \in \mathbb{R}$ (scalar).

Fitting a line to n-dimensional data.

Consider m by n matrix X.

$$X = m \underbrace{\begin{bmatrix} & & \\ & & \end{bmatrix}}_{n}$$

- Each row is equal to x^i drawn from D.
- Each column: Each point is in $\mathbb{R}^n$.

$\vec{y} \in \mathbb{R}^m \leftarrow$ labels for these m points $(m \times 1)$

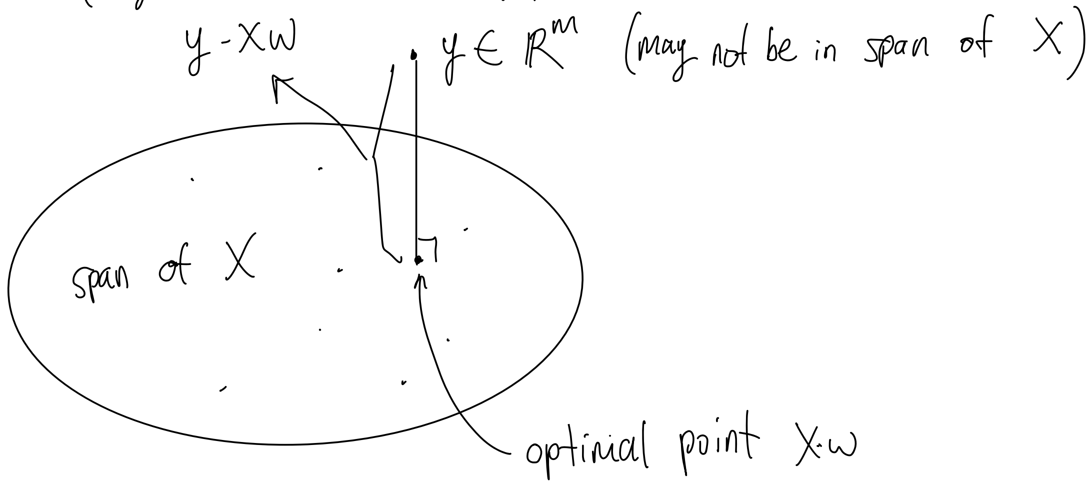
Goal: Find a vector $\underbrace{w \in \mathbb{R}^n}_{n \times 1}$: $\min_w \| \underbrace{X \cdot \vec{w}}_{m \times 1} - \vec{y} \|_2^2$

$X^1 = x^1_1, \dots, x^1_n$

$(y - (x^1_1 w_1 + \dots + x^1_n w_n))^2$

$\min_w \|X \cdot \vec{w} - \vec{y}\|_2^2$. How to find $\vec{w}$?

$X \cdot w$ is a vector in the span of the columns of X (e.g. # of features), $n \times 1$.



vector $y - Xw$ is orthogonal to X:

$X^T \cdot (y - Xw) = 0$. orthogonal = inner product = 0.

$\Rightarrow X^T y - X^T X w = 0$

$\Rightarrow X^T y = X^T X w$

$\Rightarrow \boxed{(X^T X)^{-1} X^T y = w \quad \text{Normal Equations.}}$

Issues:

1) what if $X^T X$ is not invertible?
"Pseudo-inverse".

2) What is the running time for computing w?
crude estimate: $O(n^3 + m \cdot n^2)$.
Can improve using gradient descent (later).

Maximum Likelihood:

Assume "simple linear regression case".

x; Assume $y = \beta_0 + \beta_1 x + \epsilon$

random noise variable,
$\epsilon \sim N(0, \sigma^2)$

Drawn $x^1, \ldots, x^m$ and $y^1, \ldots, y^m$.
we want to understand:
for a fixed choice of β_0 and β_1,
(σ^2 is known),

what is the probability that we see $(x^1, y^1), \ldots, (x^m, y^m)$.

Depends on β_0 and β_1.

<u>Likelihood Function</u>:

Probability of seeing training set given a choice β_0 and β_1 of our parameters.

$$\underbrace{\prod_{i=1}^{m} p(y^i | x^i ; \beta_0, \beta_1)}_{} =$$

$$\text{Gaussian: } \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left[- \frac{(y^i - \overbrace{(\beta_0 + \beta_1 x^i)}^{\mu})^2}{2\sigma^2} \right]$$

likelihood of our training set.

Choose β_0 and β_1 that maximize the likelihood.

Instead of directly maximizing the likelihood, we max the log of it.

$$\log(L(\beta_0, \beta_1)) = \log \prod_{i=1}^{m} p(y^i | x^i ; \beta_0, \beta_1)$$

$$= \sum_{i=1}^{m} \log(p(y^i | x^i ; \beta_0, \beta_1))$$

$$= \underbrace{-\frac{m}{2}\log 2\pi}_{\text{constant}} \underbrace{- m\log\sigma}_{\text{constant}} - \frac{1}{2\sigma^2}\underbrace{\sum_{i=1}^{m}\left(y^i - (\beta_0 + \beta_1 x^i)\right)^2}_{\text{least-squares estimate for simple linear regression}}$$

Two interpretations for coefficients in linear regression:

- Geometric: coefficients of the line that minimizes squared distance from the line to our labels.

- Statistical: coefficients give you the maximum likelihood estimator for a training set generated per the assumption.

  $y \sim N(\beta_0 + \beta_1 x, \epsilon)$.

Read ch 9 of ML book.