Recap: Linear Regression: Closed Form Solution

$$w^* = \arg\min_w \frac{1}{n}\sum_{i=1}^{n} \left(y_i - w^T x_i \right)^2 = \arg\min_w \overbrace{\frac{1}{n}\left\| Y - Xw \right\|_2^2}^{L}$$

where $X = \begin{bmatrix} \leftarrow x_1^T \rightarrow \\ \vdots \\ \leftarrow x_n^T \rightarrow \end{bmatrix}_{n\times(d+1)}$ $\quad w = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix}_{(d+1)\times 1}$ $\quad Y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}_{n\times 1}$

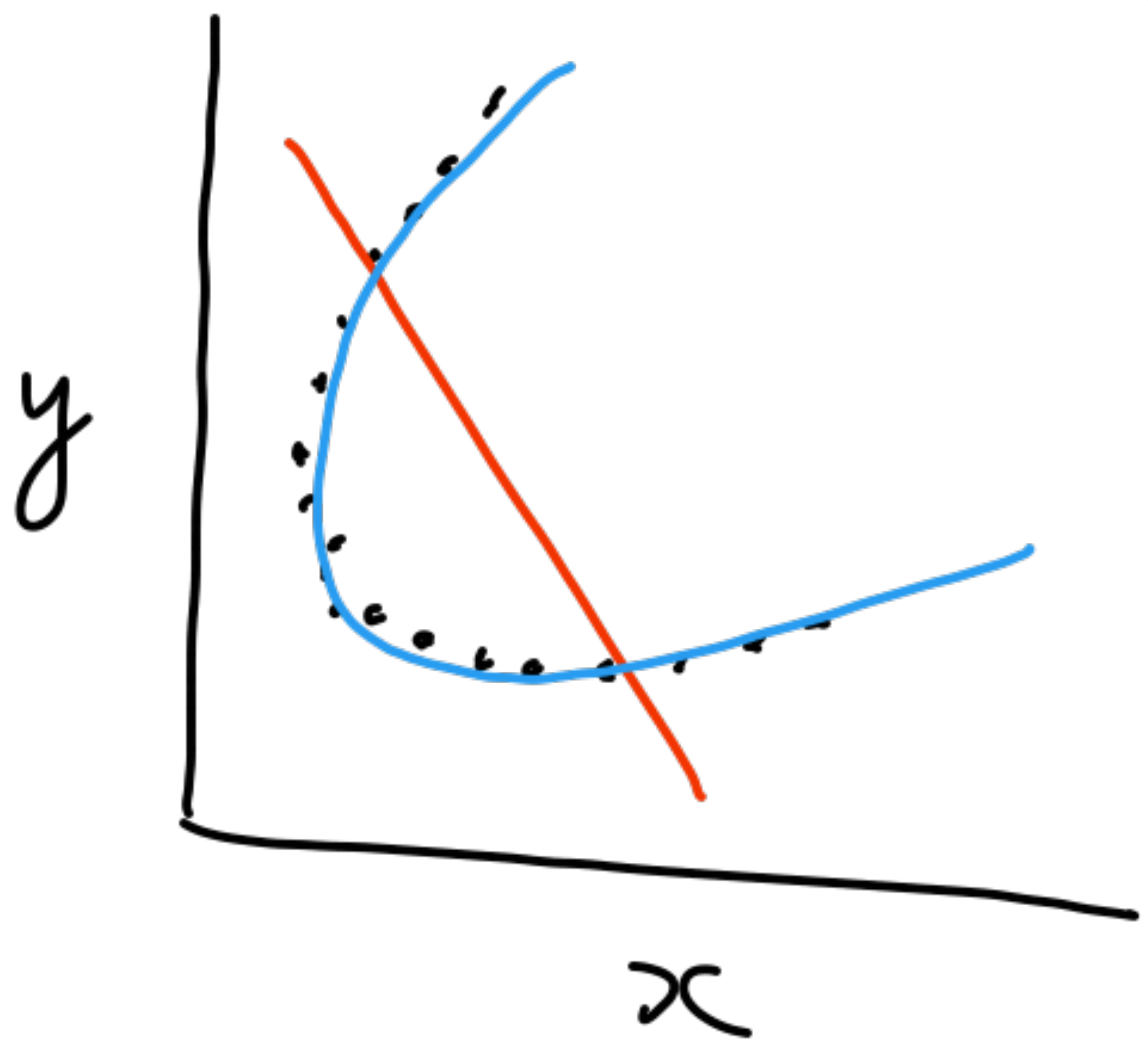
$x_i = \begin{bmatrix} 1 \\ x_{i1} \\ \vdots \\ x_{id} \end{bmatrix}_{(d+1)\times 1}$

$$\begin{bmatrix} \frac{\partial L}{\partial w_0} \\ \frac{\partial L}{\partial w_1} \\ \vdots \\ \frac{\partial L}{\partial w_d} \end{bmatrix}$$

$$\nabla_w L = 0 \Rightarrow -\frac{2}{n}\sum_i \left(y_i - w^T x_i \right) x_i = 0 \Rightarrow -2\sum_i y_i x_i + 2\sum_i \left(w^T x_i \right) x_i = 0$$

$$\Rightarrow 2X^T Y = 2X^T X w \Rightarrow \boxed{w = \left(X^T X \right)^{-1} X^T Y}$$

Consider the following dataset



$$h_w(x) = w_0 + w_1 x$$

$$h_w(x) = w_0 + w_1 x + w_2 x^2$$

More generally, can we transform the datapoints to a different dimensionality so as to enable a better regression fit

Do this with "BASIS FUNCTIONS"

BASIS FUNCTION

Define $\phi : \mathcal{X} \to \mathbb{R}^m$, $\phi(x) = \begin{bmatrix} 1 \\ \phi_1(x) \\ \phi_2(x) \\ \vdots \\ \phi_m(x) \end{bmatrix}_{(m+1) \times 1}$ Scalars

$x \in \mathbb{R}^d = \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_d \end{bmatrix}$

Typically $m \gg d$

but $d > m$ as well if there are correlated/spurious features in $\mathbb{R}^d$ that we want to eliminate

Common types of basis functions

① Polynomial basis ; for 1-D points, $\phi_j(x) = x^j$

② Radial basis function (RBF) ; for 1D points,
(Gaussian basis)

$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{\sigma_j^2}\right\}$$

Where μ_j and σ_j are predefined

3. Piecewise linear basis
4. Periodic basis ($\sin x$, $\cos x$, etc.)
5. Fourier basis

⋮

What is the loss function with basis functions?
& the corresponding optimization problem

$$w^* = \underset{w}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \left(y_i - w^T \phi(x_i)\right)^2 = \underset{w}{\operatorname{argmin}} \|y - \Phi w\|_2^2$$

where $\Phi = \begin{bmatrix} \longleftarrow \phi(x_1)^T \longrightarrow \\ \vdots \\ \longleftarrow \phi(x_n)^T \longrightarrow \end{bmatrix}_{n \times (m+1)}$

$$\boxed{w^* = (\Phi^T \Phi)^{-1} \Phi^T y}$$

GRADIENT DESCENT (GD) FOR LINEAR REGRESSION

GD is a first-order iterative algorithm used to find local optima of a differentiable function

GD is derived from gradient descent

gradient: $\nabla_w L = \begin{bmatrix} \frac{\partial L}{\partial w_0} \\ \frac{\partial L}{\partial w_1} \\ \vdots \\ \frac{\partial L}{\partial w_d} \end{bmatrix}$, gradient is the direction of fastest increase in L

descent : We are interested in minimizing a loss fn

So, update w in the reverse direction of the gradient

Apply GD in order to estimate w which parameterizes a loss function L

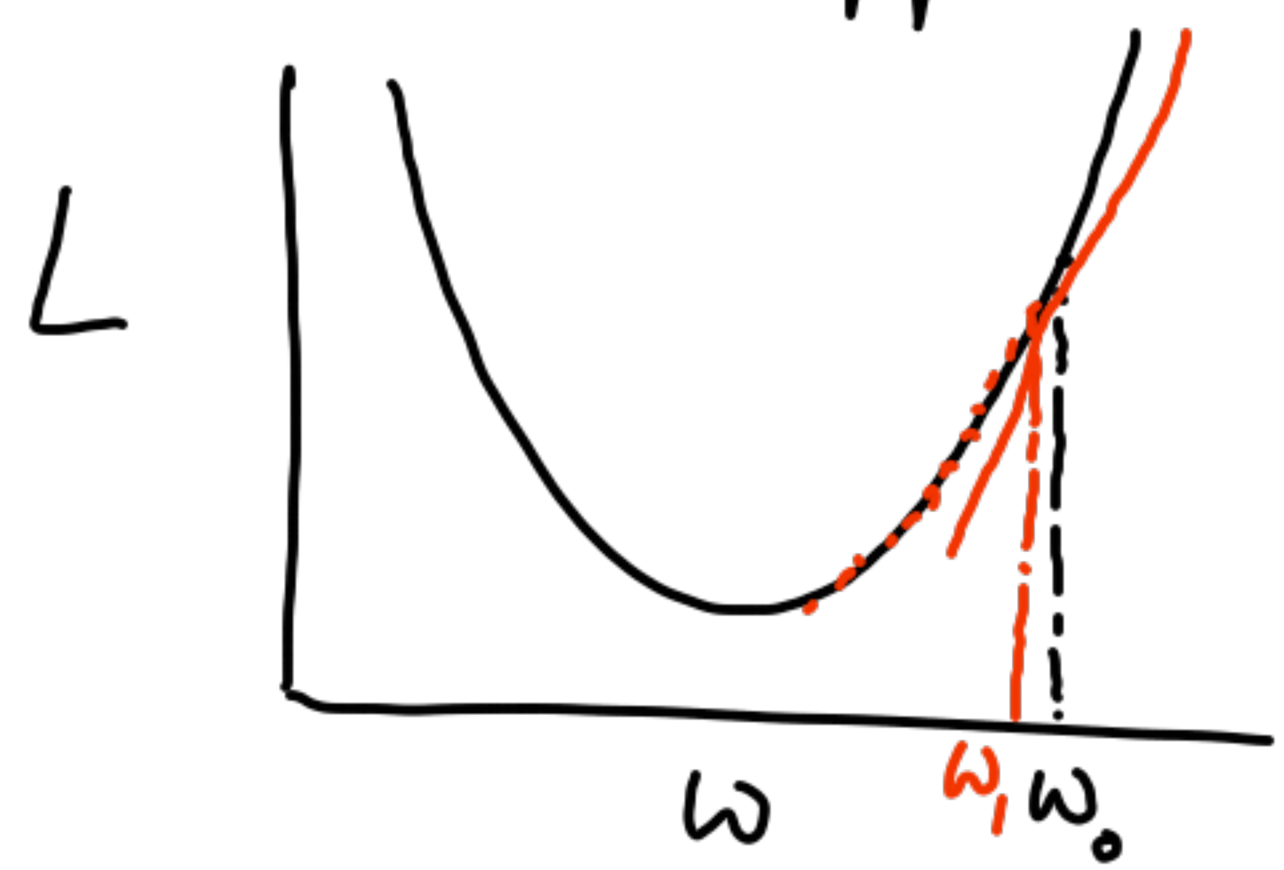
General template of a GD-style algorithm:

- Initialize w
- Repeat
  - choose a descent direction
  - choose a step size
  - update w using an update rule
- Exit repeat if some stopping criterion is met

GRADIENT DESCENT

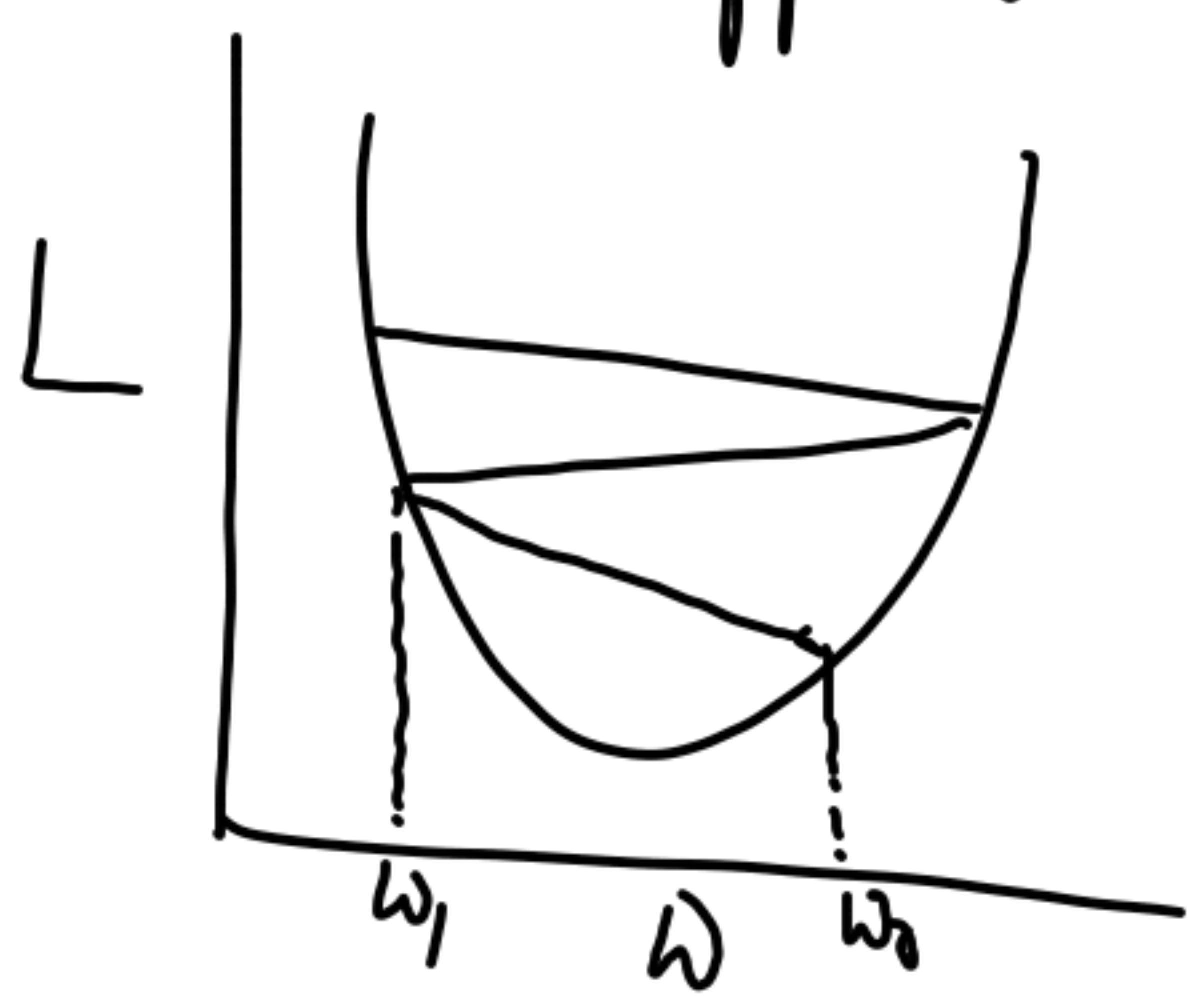
- $w \leftarrow w_0$ (w_0 can be all 0's, random uniform, random Gaussian, etc.)
- Repeat
  - Descent direction: $-\nabla_w L$
  - step size is determined by $\eta > 0$ ("LEARNING RATE")
  - $w_{t+1} \leftarrow w_t - \eta \nabla_{w_t} L$
- Exit when $\|w_{t+1} - w_t\|^2 < \epsilon$

(1) What happens with GD if the step size is too small?



Going to take many iterations to converge

(2) What happens when the step size is too large?



Could delay arriving at the final solution

At worst, the loss could diverge