Logistic Regression and Cross Entropy Loss

Binary Classification

$D = \{(\underline{x_1}, y_1), (\underline{x_2}, y_2), \ldots, (\underline{x_N}, y_N)\}$

$\underline{x_i} \in \mathbb{R}^{d+1}$

$y_i \in \{-1, +1\}$

$\text{Output}(\hat{P}_{\underline{w}}(-1 \mid x), \hat{P}_{\underline{w}}(1 \mid x))$

$$\hat{P}_{\underline{w}}(y \mid \underline{x}) = \frac{e^{y\underline{w}^T\underline{x}}}{1+e^{y\underline{w}^T\underline{x}}}$$

Loss Function: Log Loss

$$e_n(\underline{w}) = -\log \hat{P}_{\underline{w}}(y \mid \underline{x_n})$$

$$E_{in}(\underline{w}) = \frac{1}{N}\sum_{n=1}^{N} e_n(\underline{w})$$

$D' = \{\underline{(x_1}, P_1), \underline{(x_2}, P_2), \ldots, \underline{(x_n}, P_n)\}$

$P_i = (P_i(1), P_i(2))$

if $y_i = 1$, $P_i = (1, 0)$

if $y_i = -1$, $P_i = (0, 1)$

$$\begin{aligned} CE(P_i, \hat{P}_w) &= -(P_i(1)\log \hat{P}_w(1 \mid x_n)) + \\ &\quad P_i(2)\log \hat{P}_w(-1 \mid x_n) \\ &= e_n(\underline{w}) \end{aligned}$$

$$E_{in}(\underline{w}) = \frac{1}{N}\sum_{n=1}^{N} CE(P_n, \hat{P}_w(\underline{x_n}))$$

Knowledge Distillation

$$x \longrightarrow \boxed{\text{Powerful ML Model}} \longrightarrow (\hat{P}_i(1), \hat{P}_i(2))$$

(M)

$P_i(y=+1), P_i(y=-1)$

Limited Date

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

$$D \longrightarrow \boxed{\text{Logistic Regression}} \longrightarrow \text{Bad Accuracy}$$

$$x_i \longrightarrow \boxed{\text{Model } M} \longrightarrow \hat{P}_i$$

$$D' = \{(x_1, \hat{P}_1), (x_2, \hat{P}_2), \dots, (x_N, \hat{P}_N)\}$$

$$D' \longrightarrow \boxed{\text{Logistic Regression}} \longrightarrow \text{Improved Accuracy}$$

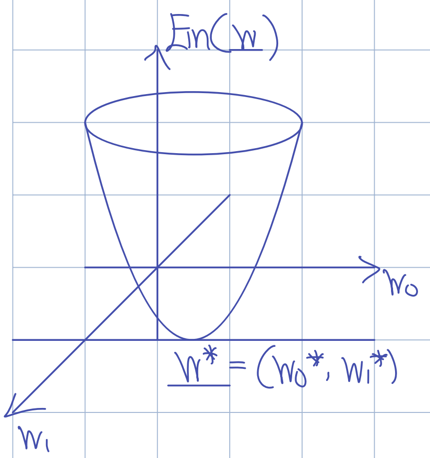
Cross Entropy Loss

Training for Logistic Regression

Minimize training error

$$E_{in}(\underline{w}) = \frac{1}{N}\sum_{n=1}^{N} e_n(\underline{w})$$

$$e_n(\underline{w}) = \log(1 + e^{-y_n \underline{w}^T x_n})$$

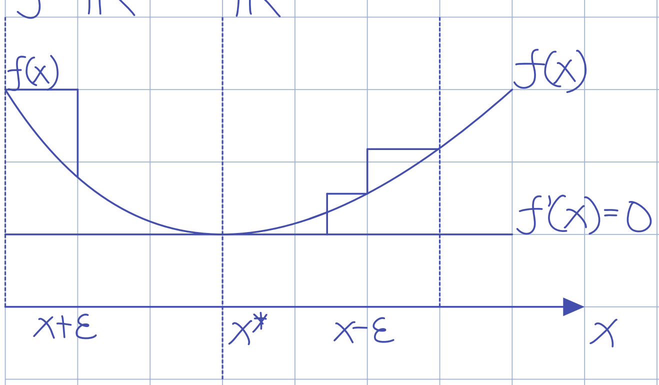


$\nabla E_{in}(\underline{w}) = 0 \longrightarrow$ No analytical solution

To Solve For $\underline{w}^*$

Solve numerically using Gradient Descent

$f: \mathbb{R} \longrightarrow \mathbb{R}$



If x^* minimizes $f(x)$

$f'(x^*) = 0$

Analytically computing x^* may not be tractable

Consider any $x \in \mathbb{R}$

if $x < x^*$, $f(x)$ is decreasing, $f'(x) < 0$

if $x > x^*$, $f(x)$ is increasing, $f'(x) > 0$

if $x = x^*$, $f'(x) = 0$

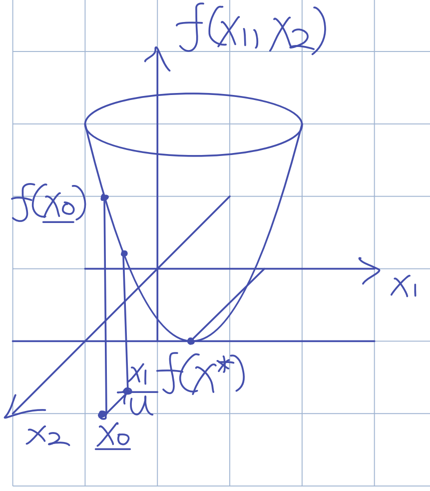
Gradient Descent for $f : \mathbb{R} \to \mathbb{R}$

1. Initialize $x = x_0$
2. If $f'(x) \approx 0$ then stop & output x
3. If $f'(x) > 0$ then $x = x - \varepsilon$
4. If $f'(x) < 0$ then $x = x + \varepsilon$
5. Go to step 2

ε = Step Size

Typically decrease ε as the iterations progress

$f(x_1, x_2)$



Let $\underline{x_0}$ be the initial point

Update Rule

$\underline{x_1} = \underline{x_0} + \delta\, \underline{u}$

$\underline{u}$ = direction vector

δ = step size along $\underline{u}$

δ will be fixed to a small constant (eg 10^{-3})

We will focus on selecting $\underline{u}$

$$f(\underline{x_1}) = f(\underline{x_0} + \delta \underline{u})$$

$$||\underline{u}|| = 1$$

pick $\underline{u}$ among all direction rectors that minimize $f(\underline{x_1})$

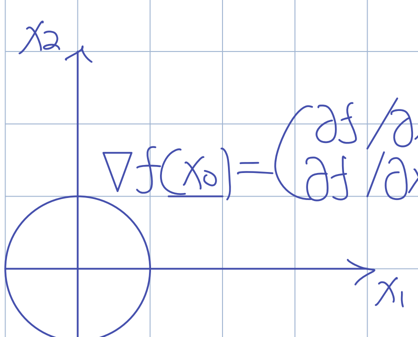
$$\underline{u}^* = \underset{\underline{u} \in \mathbb{R}^2}{\arg\min}\, f(\underline{x_0} + \delta \underline{u})$$

Since δ is small, we can apply the taylor series approximation

$$f(\underline{x_0} + \delta \underline{u}) \simeq f(\underline{x_0}) + \delta \underline{u}^T \nabla f(\underline{x_0})$$

$$\underset{\underline{u}, \|u\|_2 = 1}{\arg\min} \{ \underbrace{f(\underline{x_0})}_{\text{Constant}} + \delta \underline{u}^T \nabla f(\underline{x_0}) \}$$

$$= \underset{\underline{u}, \|u\|_2 = 1}{\arg\min} (\underline{u}^T \nabla f(\underline{x_0}))$$



$$\nabla f(\underline{x_0}) = \begin{pmatrix} \partial f / \partial x_1 \\ \partial f / \partial x_2 \end{pmatrix}$$

$$\underline{u}^* = \frac{-\nabla f(\underline{x_0})}{||\nabla f(\underline{x_0})||}$$

Update Rule

$$x_1 = x_0 - \delta \frac{\nabla f(x_0)}{\| \nabla f(x_0) \|}$$

$$x_t = x_{t-1} - \delta_t \frac{\nabla f(x_t)}{\| \nabla f(x_t) \|}$$