MACHINE LEARNING IN PHYSICS
TUTORIAL 1

HARRISON B. PROSPER

PHY6938

Recap

- As noted last week, machine learning (ML) models, $f(x;\omega)$, are *trained* (that is, fitted to data) by minimizing the empirical risk function

$$R(\omega) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, f_i)$$

which is a generalization of fitting using least-squares.

- We also saw that the mathematical quantity approximated by our model, $f(x;\omega)$, is determined solely by the loss function, $L(y_i, f_i)$, and the probability distribution, $p(x,y)$, of the data.

Recap

Quadratic loss

$$L(y, f) = (y - f)^2$$

$$f(x, \omega^*) = \int y \, p(y \mid x) \, dy$$

Binary cross entropy

$$L(y, f) = -[y \log f + (1 - y) \log(1 - f)]$$

$$f(x, \omega^*) = p(y = 1 \mid x) = \frac{p(x|y = 1)\epsilon}{p(x|y = 1)\epsilon + p(x|y = 0)}$$

$$\epsilon = \pi(y = 1)/\pi(y = 0)$$

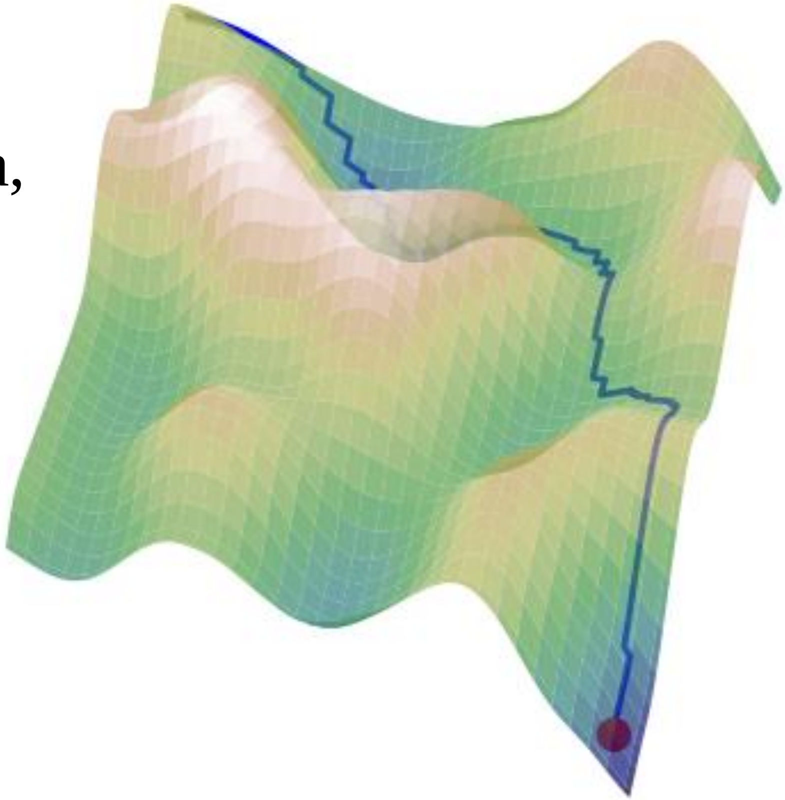
Recap

Stochastic Gradient Descent

1. At the current point ω_j, compute a *noisy* approximation to the gradient of

$$R(\omega) \approx \frac{1}{n}\sum_{i=1}^{n} L(y_i, f_i)$$

by using a **batch** of **training** data, where $\boldsymbol{n} \ll \boldsymbol{N}$.



2. Move to the next position ω_{j+1} in the landscape using

$$\omega_{j+1} = \omega_j - \eta \nabla R$$

TUTORIAL 1

Tutorial 1

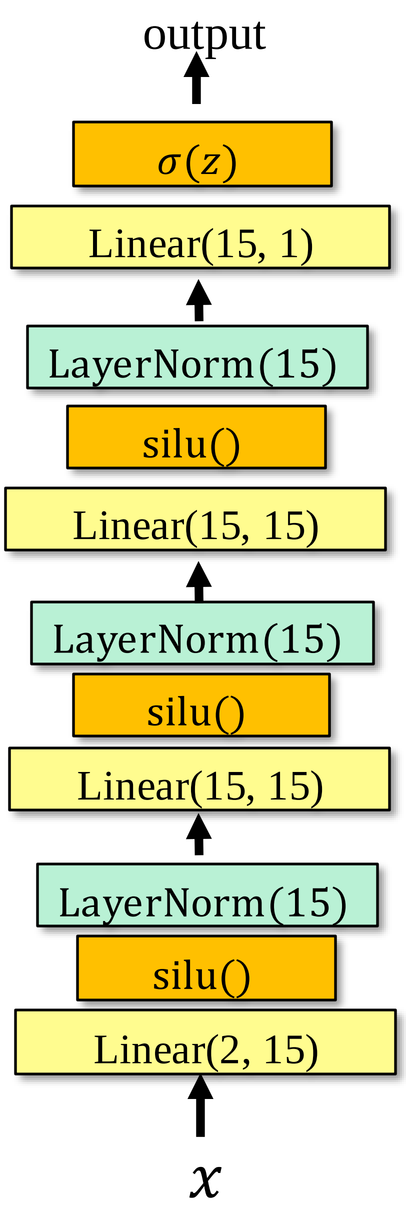
Task

- Classify points in our 2-sample dataset.

Method

- Train model $f(x;\omega)$ using the binary cross entropy loss and find the best-fit parameter $\widehat{\omega} = \text{argmin}_{\omega} R(\omega)$.

- Compare $f(x;\widehat{\omega})$ with the exact classifier.



Tutorial 1: Dataset

Sample 1

$\mu_1 = (0.25, -0.25)$

$cov_1 = \begin{pmatrix} 0.20 & 0.24 \\ 0.24 & 0.40 \end{pmatrix}$

Sample 2

$\mu_2 = (-0.10, 0.10)$

$cov_2 = \begin{pmatrix} 0.60 & 0.40 \\ 0.40 & 0.30 \end{pmatrix}$