Logistic Regression & Neural Networks

Saeed Saremi

Assigned reading: 5.3.1, 5.4.$\{1, 2, 3, 4\}$, 6.$\{1, 2\}$

September 19, 2024

a summary of the previous lecture

- We "separated" the problem of classification, of learning the mapping $f : \mathcal{X} \to \{0, \ldots, K-1\}$, into an inference step and the decision step.[1]
- We first took a generative viewpoint, assuming $X|k \sim \mathcal{N}(\mu_k, \Sigma)$ and found out for two-class classification the posterior $p(k|x)$ takes the form of the logistic function:

$$p(k|x) = \frac{1}{1 + e^{-z}},$$

where z is linear function:

$$z = \theta^\top x + \theta_0,$$

where θ is depends on $\{\mu_k\}_{k=0}^{K-1}$ and Σ.
- In the generative approach μ_k and Σ are estimated from the data.
- We learnt than one can parametrize the posterior distribution directly and by some considerations, logistic function again appears. In this discriminative approach, the parameters (θ, θ_0) should be estimated directly with maximum likelihood.

[1] For simple criterion of minimizing the probability of misclassification, we arrive at

$$k = \operatorname*{argmax}_k p(k|x).$$

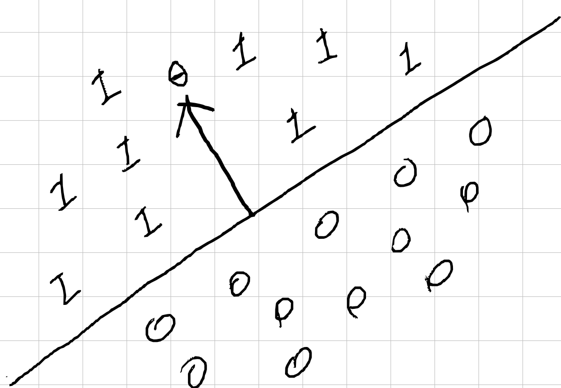
outline

- on the geometrical meaning of θ in the logistic regression
- multi-class ($K > 2$) extension of the logistic function
- the cross-entropy loss for discriminative models and its gradient
- neural networks

The GEOMETRY of θ in LOGISTIC regression

$$p_\theta(1|x) = \frac{1}{1+\exp(-\theta^T x - \theta_0)} \qquad (1)$$

$$\left.\begin{aligned} x &= x_{\parallel} + x_{\perp} \\ \theta^T x_{\perp} &= 0 \end{aligned}\right\} \Rightarrow$$ adding $\alpha x_\perp$ to x does not change $p_\theta(1/x)$



- θ is perpendicular to the decision boundary.
- with parametrization given in Eq. (1), θ "points" to class "1".
- θ_0 shifts the decision boundary.

MULTI-CLASS $(K>2)$ CLASSIFICATION

As before we start with the generative approach:

$$X \mid k \sim \mathcal{N}(\mu_k, \Sigma)$$

$X \in \mathbb{R}^d$, $k \in \{1, \dots, K\}$, $\mu_k \in \mathbb{R}^d$, $\Sigma \in \mathbb{R}^{d \times d}$

$$p(k \mid x) = \frac{p(x \mid k)\, p(k)}{\sum_{j=1}^{K} p(x \mid j)\, p(j)}$$

$$= \frac{e^{a_k}}{\sum_{j=1}^{K} e^{a_j}}$$

$$p(x \mid k) = \frac{1}{(2\pi)^{d/2}} \frac{1}{|\Sigma|} e^{-\frac{1}{2}(x-\mu_k)^T \Sigma^{-1} (x-\mu_k)}$$

$$a_k = (\underbrace{\Sigma^{-1}\mu_k}_{\theta_k})^T x \underbrace{- \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log p(k)}_{\theta_0}$$

$$a_k = \theta_k^T x + \theta_{0,k}$$

$$(a_1 \cdots a_K) \longmapsto \left(\frac{e^{a_1}}{\sum e^{a_j}}, \dots, \frac{e^{a_K}}{\sum e^{a_j}} \right)$$

SOFTMAX FUNCTION

If $a_k \gg a_j$, $j \neq k$: $(a_1 \cdots a_K) \mapsto (0\,0 \cdots \underset{\text{index } k}{1} \cdots 0)$

LEARNING in DISCRIMINATIVE models

We start with our linear two-class logistic regression model:

$$p_\theta(1|x) = \frac{1}{1+\exp(-\theta^T x - \theta_0)}$$

The log-likelihood ~~of~~ this model for any $y \in \{0,1\}$ is given by:

$$\log\left(p_z^{\,y}(1-p_z)^{1-y}\right),$$

where z depends on θ & crucially p only depends on z.

We define the loss (something we minimize) as the negative log likelihood

$$\mathcal{L}(\theta) = -y \log p_z - (1-y)\log(1-p_z)$$

We need to know the gradient $\nabla_\theta \mathcal{L}(\theta)$ to optimize ~~for~~ parameters θ.

We use the chain rule:

$$\nabla_\theta \mathcal{L}(\theta) = \partial_z\left(-y \log p_z - (1-y)\log(1-p_z)\right)\nabla_\theta z$$

Second, we use the following

$$\partial_z p_z = \partial_z \left(\frac{1}{1+e^{-z}}\right) = \frac{e^{-z}}{(1+e^{-z})^2} = p_z(1-p_z)$$

We therefore have:

$$\nabla_\theta \mathcal{L}(\theta) = \left(-y\,\frac{\cancel{p_z}(1-\cancel{p_z})}{\cancel{p_z}} + (1-\cancel{y})\,\frac{p_z\,(\cancel{1-p_z})}{\cancel{1-p_z}}\right)\nabla_\theta z$$

$$= (p_z - y)\,\nabla_\theta z \longrightarrow x \quad (\text{for linear model})$$

The final answer has a nice interpration

$$\theta \leftarrow \theta - \epsilon\,(p_z - y)\,x$$

$$\begin{cases} \theta \leftarrow \theta - \tilde{\epsilon}\,x & y=0 \\ \theta \leftarrow \theta + \tilde{\epsilon}\,x & y=1 \end{cases}$$

(This is expected from our geometrical picture of the logistic regression.)

θ is "pushed" in the direction of x for $y=1$.

The reverse happens for $y=0$.

NEURAL NETWORKS

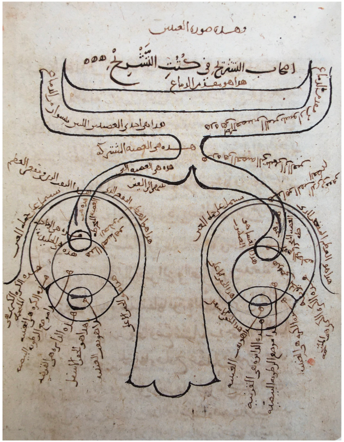


Figure: The oldest known drawing of the nervous system by Ibn al-Haytham (published in 1083)

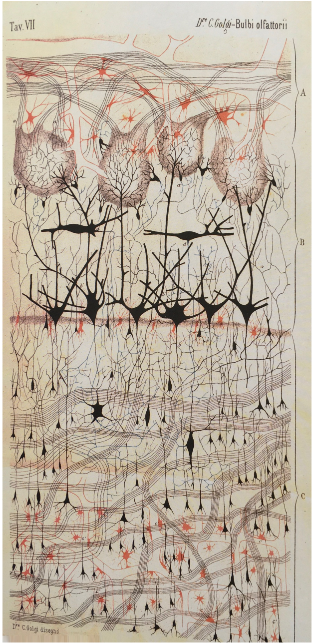


Figure: Olfactory bulb, Camillo Golgi, 1875

Santiago Ramón Y Cajal & the neuron doctrine

Neuron as the discrete distinct entities in the brain as opposed to a continuous network.

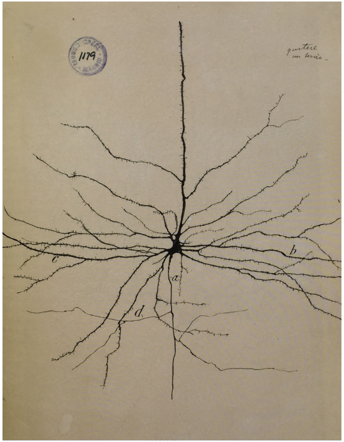


Figure: *pyramidal neuron.* Cajal, 1899

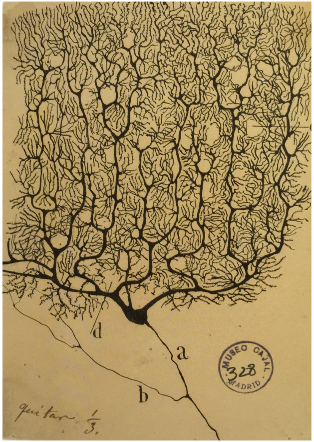


Figure: *purkinje neuron.* Cajal, 1899

The Nobel Prize in Physiology or Medicine 1906

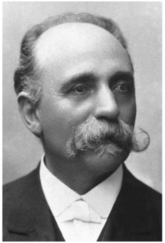


Photo from the Nobel Foundation archive.

Camillo Golgi

Prize share: 1/2

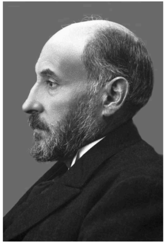


Photo from the Nobel Foundation archive.

Santiago Ramón y Cajal

Prize share: 1/2

The Nobel Prize in Physiology or Medicine 1906 was awarded jointly to Camillo Golgi and Santiago Ramón y Cajal "in recognition of their work on the structure of the nervous system"

network of neurons: axons, dendrites, and synapses

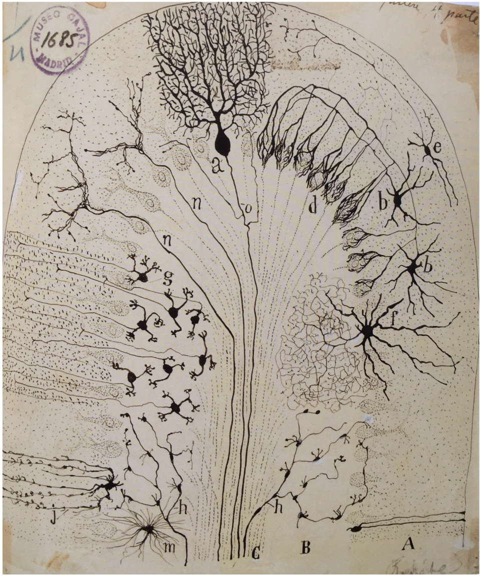


Figure: *network of neurons.* Cajal, 1899

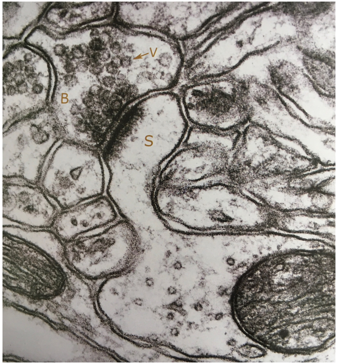


Figure: *Synapse.* Spacek and Harris, 2000

electricity in the brain

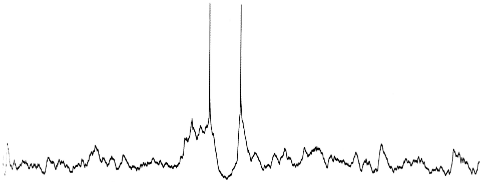


Figure: *Whole-cell recording in an awake rat.* Contantinople and Bruno, 2009.

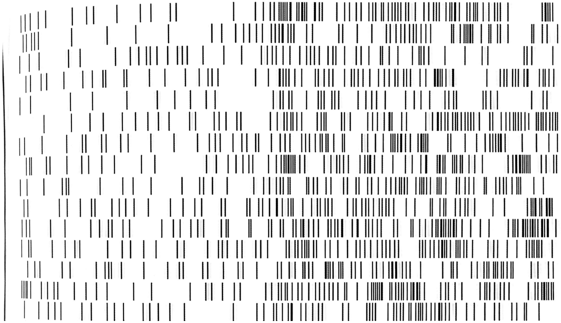


Figure: *Action potentials in a live monkey brain.* Saez and Salzman, 2009. (Each row, 4 seconds.)

biological neural networks v.s. deep neural networks

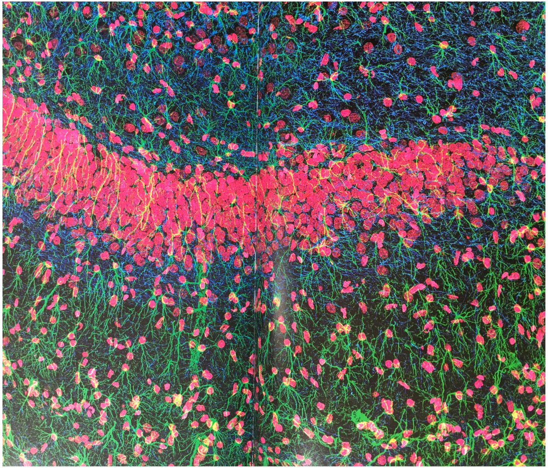


Figure: human brain: 10^{11} neurons, 10^{15} synapses

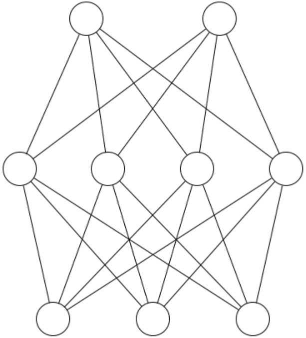


Figure: `GPT-4`: 10^6 neurons, 10^{11} parameters (weights), 100 layers

? What are we missing in this comparison?