Chapter 2

Regression, Probability, and Classification

Chapter 2

Regression, Probability, and Classification

...

2.1 Empirical Risk Minimization Unpacked

$$\min_{\vec{\theta} \in \Theta} J(\vec{\theta}) := \frac{1}{m} \sum_{i=1}^{m} L(wx_i + b, y_i)$$

2.2 Key facts about gradient with respect to a vector

If $\vec{v} \in \mathbb{R}^d$ is a vector, then

$$\nabla_{\vec{\theta}} \left(\vec{\theta}^T \vec{v} \right) = \vec{v}$$

Chain rule: If $f : \mathbb{R} \to \mathbb{R}$ has derivative $\frac{df}{dt}(\cdot)$, then

$$\begin{aligned} \nabla_{\vec{\theta}} \left(f \left(\vec{\theta}^T \vec{v} \right) \right) &= \frac{df}{dt} \left(\nabla_{\vec{\theta}} \left(\vec{\theta}^T \vec{v} \right) \right) \\ &= \frac{df}{dt} \left(\vec{\theta}^T \vec{v} \right) \vec{v} \end{aligned}$$

<MINTED>

2.3 Outlier and Nonlinear Models

- Recall linear model was sensitive to large noise. What about the quadratic case?

2.4 Summary

We've studied

- Linear models
- Polynomials models
- Square losses
- Absolute values losses
- **Next**: neural networks

2.5 Polynomials of degree d

- Models of growing complexity.
- Richer class of function.
- More parameters.
- More expressive.

Neural networks width/depth is analogous to the degree of a polynomial.

Table 2.1

	Expressiveness	Prone-ness to overfitting
More degree	High	Higher-ish
Less degree	Low	Low

Polynomial suffers from numerical issues when degree is higher. This affects NN less (why NN tolerates numerical issues better??).

2.6 Activation Functions

Rectified linear unit or ReLU.

$$\text{relu}(z) = \max(0, z)$$

<MINTED>

2.7 One-neuron Network

Model params $\vec{\theta} = [a\ b\ w] \in \vec{\Theta}$

$$f(x; \theta) = a \times \text{relu}(w \times x + b)$$

2.8 Two-neuron Network

Model params $\vec{\theta} = \left[\vec{a}\ \vec{b}\ \vec{w}\right] \in \vec{\Theta}$ where $\vec{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$, $\vec{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$, $\vec{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$

$$f(x;\theta) = a_1 \times \text{relu}(w_1 \times x_1 + b_1) + a_2 \times \text{relu}(w_2 \times x_2 + b_2)$$

2.9 Gradient Descent (GD)

Let $\epsilon_k > 0$ be learning rates, $k = 1, 2, \ldots$

- Initialize $\vec{\theta}$
- While not converged (k = iteration counter):
  - Compute gradient ...
  - Computer update $\vec{\theta} \leftarrow \vec{\theta} \ldots$

2.10 Empirical Risk Minimization Unpacked

$$J(\vec{\theta}) \ldots$$

$$\nabla_{\vec{\theta}} J_i(\theta) = \nabla_{\vec{\theta}} L\left(f(x_i;\vec{\theta}), y_i\right)$$

Observation: more parameters seems to make optimization easier. In other words, ...

2.11 Probability

We want to connect the optimization we did so far to something statistically grounded. Diffusion model was invented in or around 2015 based on statistical principles. Optimization didn't come out of the blue, they were based on statistical ideas.

- Probability distribution: $p(\vec{x})$
- $\vec{x}$ belongs with some set χ
- Sampling from the distribution

$$\vec{x}^{(1)}, \ldots, \vec{x}^{(m)}, \sim p(\vec{x})$$

- i.i.d stands from "independently and identically distributed"

2.11.1 Example: Gaussian

- $\chi = \mathbb{R}$
- $p(x)$ is the "probability density function" for the "standard Gaussian" distribution

  <MINTED>
- We say that $x \sim p(\vec{x})$ is a *continuous random vector*
- What does it mean to have access to a distribution? It means we can sample from it.

2.11.2 Example: Uniform Distribution

-

2.11.3 Example: Bernoulli Distribution

- $\Gamma = \{0, 1\}$
  - 1 = head, 0 = tail in a coin toss
- $p(y)$ is the "probability mass function" for the "Bernoulli" distribution for a fair coin toss

  <MINTED>
- Height of the density is how likely your sample lands here.
- We say that $y \sim p(y)$ is a *discrete random variable.*
- $p(y)$ is the "probability mass function" for the "Bernoulli" distribution for a $q \in (0, 1)$ biased coin

  <MINTED>

2.12 Joint probability

- Joint probability distribution: $p(\vec{x}, y)$
- $\vec{x}$ belongs to some set χ
- $\vec{y}$ belongs to some set γ
- Sampling from ...

How to construct a Joint probability distribution $p(\vec{x}, y)$?

- Sample $\vec{x}$ from some density $p(\vec{x})$

  <MINTED>

- Pick some function $g : \chi \rightarrow \gamma$

  <MINTED>

- Add some noise

2.13 Conditional Distribution

Conditional probability: $p(y|\vec{x})$ probability of y given $\vec{x}$

2.14 Gaussian/normal distribution

- Assumption: $p_{\text{data}}(y|\vec{x})$ is distributed according to $y = \vec{w}^T\vec{x} + b + \epsilon$, where
- Gaussian distribution with mean μ and variance σ^2

$$\epsilon \sim N\left(\mu, \sigma^2\right)$$

- y is deterministic function if not for noise ϵ
- The probability density function (PDF)

$$\begin{aligned} N(\epsilon; \mu, \sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\epsilon-\mu)^2}{2\sigma^2}\right) \\ N(\epsilon; \mu, \sigma^2) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-(\vec{w}^T\vec{x}+b)-\mu)^2}{2\sigma^2}\right) \end{aligned} \tag{2.1}$$

2.15 Maximum Likelihood

Probability of observing the data.

$$\prod_{i=1}^{m} p_{\text{model}}\left(y^{(i)}|\vec{x}^{(i)}; \vec{\theta}\right) = \prod_{i=1}^{m} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\left(y^{(i)} - f\left(\vec{x}^{(i)}; \vec{\theta}\right)\right)^2}{2\sigma^2}\right) \tag{2.2}$$

2.16 Joint probability for classification?

How to construct a Joint probability distribution $p_{\text{model}}(\vec{x}, y)$?

- Sample $\vec{x}$ from some density $p(\vec{x})$

  <MINTED>

- Suppose $\gamma = \{1, \ldots, K\}$

- Pick some function $f(x; \vec{\theta}) : \chi \rightarrow \mathbb{R}^K$

  <MINTED>

- The softmax function

  <MINTED>

- Draw labels

  <MINTED>