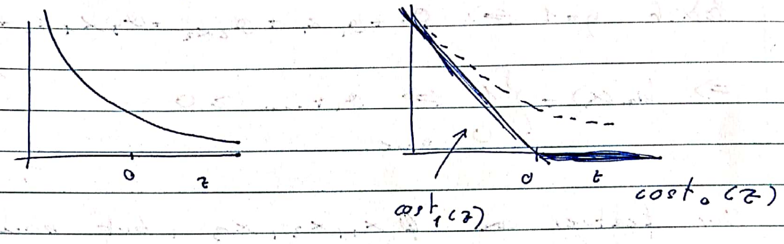
Week 7 | SPV

• Alternative view of logistic regression

We'll transform this in this

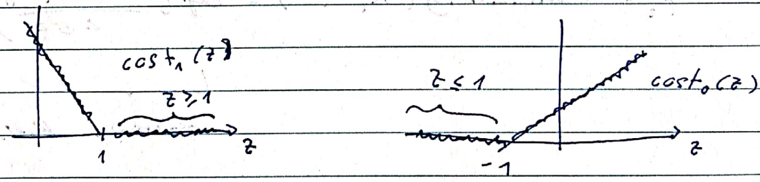


Logistic reg:

$$\frac{\partial}{\partial\theta}\left[\frac{1}{m}\left[\sum_{i=1}^{m} y^i\left(-\log h_\theta(x^i)\right) + (1-y^i)\left(\left(-\log(1-h_\theta(x^i))\right)\right)\right] + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2\right]$$

SPV

$$\frac{\partial}{\partial\theta}\left[C\sum_{i=1}^{m}\left[y^i cost_1(\theta^T x^i) + (1-y^i) cost_0(\theta^T x^i)\right] + \frac{1}{2}\sum_{i=1}^{n}\theta_j^2\right]$$



if $y=1 \Rightarrow \theta^T x \geq 1$
if $y=0 \Rightarrow \theta^T x \leq -1$

it's what we want; not just $\theta^T x \geq$ or ≤ 0, then we create a proper margin (decision boundary)

KERNELS

- Non linear decision boundary

Predict $y=1 \iff \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1 x_2 + \ldots \geq C$

$$\Rightarrow h_\theta(x) = \begin{cases} 1 \iff \theta_1 + \ldots \geq 0 \\ 0 \text{ else} \end{cases}$$

Our features are x_1, x_2, x_3, but now we'll use others $\left\{ f_1 = x_1, f_2 = x_2, f_3 = x_1 x_2 \ldots \right.$ for example

→ Given x, we'll compute new features depending on proximity to landmarks l^1, l^2, l^3

$$f_1 = \text{similarity}(x, l^1) = \exp\left(-\frac{\|x - l^1\|}{2\sigma^2}\right)$$

$$f_2 = \text{similarity}(x, l^2) = \underbrace{\exp\left(-\frac{\|x - l^2\|}{2\sigma^2}\right)}_{\text{gaussian kernels}}$$

- Kernels and similarity
  - if $x \approx l^i \Rightarrow f_1 \approx 1$
  - if x far from $l^i \Rightarrow f_1 \approx 0$

$$\Rightarrow h_\theta(f) = \begin{cases} 1 \iff \theta_1 f_1 + \theta_2 f_2 + \theta_3 f_3 \ldots \geq 0 \\ 0 \text{ else} \end{cases}$$

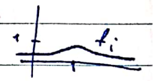
• SVM with kernels

$$\frac{\partial}{\partial\theta} C \left[\sum_{y=1}^{m} y^i \, cost_1(\theta^T f^i) \right] + (1-y^i)\, cost_0(\theta^T f^i) + \frac{1}{2}\sum_{j=1}^{m} \theta_j^2$$

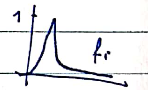
$$C = \frac{1}{\lambda} \begin{cases} C >>> \Rightarrow \text{lower bias, high variance} \\ C <<< \Rightarrow \text{higher bias, low variance} \end{cases}$$

σ^2 :

- $\sigma^2 >>$ ⇒ f_i will vary more smoothly
  ⇒ higher bias, low variance



- $\sigma^2 <<$ ⇒ f_i will vary less smoothly
  ⇒ lower bias, high variance



• Using SPV packages
  - Choice of C
  - Choice of kernel

Ex: linear kernel ⇒ $y=1 \Leftrightarrow \theta^T x \geq 0$

→ • Feature scaling before using Gaussian kernel!!!

• SPV multiclass classification
  → 1 vs all method

• Logistic regression VS SVM

$n \equiv$ n° of features ; $m \equiv$ n° of training examples

1 • If $n >> m \Rightarrow$ LR or SVM without kernel (linear)

2 • If $n > m \Rightarrow$ SVM with gaussian kernel

3 • If $n < m \Rightarrow$ Create/add more features & go to 1

In python problems this is not typical at all. Training set always way bigger than number of features.

?

Neural networks likely to perform nice in every circumstance, but may be slower to train

Scanned with CamScanner