(Non-examinable)

Probability estimation question

Iain is juggling.

Assume he has a 1% chance of dropping each catch.

(not really independent, but let's pretend)

Estimate without a computer/calculator the probability he manages 100 catches in a row on first attempt.

$N = 100$ catches, $f = 0.01$ drop prob.

$$P(N \text{ catches}) = \prod_{n=1}^{N} (1-f)$$

$$= (1-f)^N \qquad \approx 1 - Nf \ ? \quad \Rightarrow 0 \text{ here} \ \times$$

$$\log P(N \text{ catches}) = N \log(1-f)$$

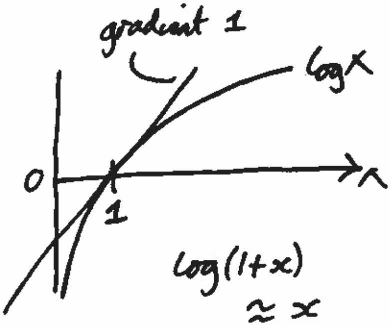
$$\approx -Nf$$

$$\approx -1 \text{ here}$$

$$P(100 \text{ catches}) \approx e^{-1}$$

$$= 1/e$$

$$\approx \underline{\underline{0.37}}$$



$\log(1+x) \approx x$ for small x

"about a third"

In Matlab/Octave or Numpy

$$\log(1 - 1e{-20}) \text{ "=" } 0 \qquad 10^{-20}$$

$$\text{log1p}(-1e{-20}) = 1e{-20}$$

"$\log(1+x)$"

Recipe: fit parameters to data

Loss and function: $L(y^{(n)}, f(\underline{x}^{(n)}, \underline{w})) = L_n$

Square loss: $(y^{(n)} - f_n)^2$

Negative log probability: $-\log p(y^{(n)} \mid \underline{x}^{(n)}, \underline{w})$

Cost

$$C = \sum_n L_n \quad \text{sum over training set}$$

Learning direction

$$-\frac{1}{N} \nabla_{\underline{w}} C = \underbrace{-\frac{1}{N} \sum_n \nabla_{\underline{w}} L_n}_{\substack{\text{Monte Carlo approx:} \\ -\nabla_{\underline{w}} L_n \text{ for random } n}}$$

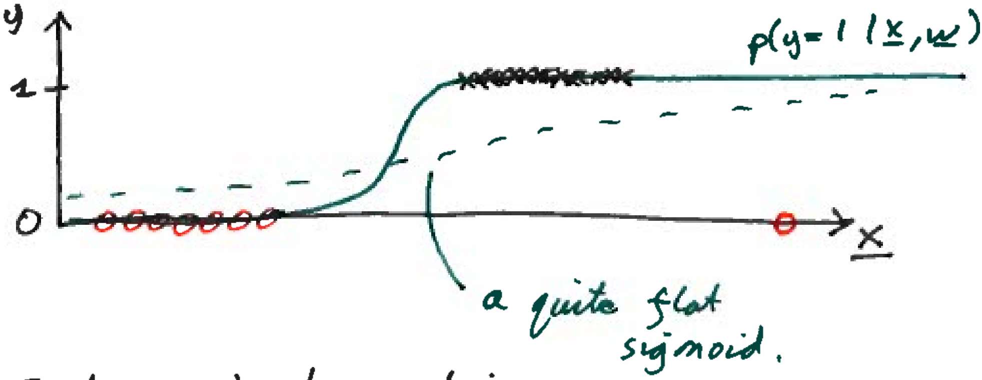
$\Rightarrow$ Identify probabilistic model of y, $p(y \mid \underline{x}, \underline{w})$

$\Rightarrow$ Update parameters:

$$\underline{w} \leftarrow \underline{w} + \eta \nabla_{\underline{w}} \log p(y^{(n)} \mid \underline{x}^{(n)}, \underline{w})$$

S.G.D. on loss. S.G. Ascent on log likelihood

Robust Logistic Regression



Each example has a binary variable

$m^{(n)} \in \{0, 1\}$ hidden variable / latent variable

I will assume:

$$P(m\,|\,\varepsilon) = \begin{cases} 1-\varepsilon & m=1 \\ \varepsilon & m=0 \end{cases}$$

$\varepsilon \approx 0.01$?

Model for labels:

$$P(y=1\,|\,\underline{x},\underline{w},m) = \begin{cases} \sigma(\underline{w}^T\underline{x}) & m=1 \\ 1/2 & m=0 \end{cases}$$

Need Likelihood of $\underline{w}$, ε

$$P(y=1|\underline{x},\underline{w},\varepsilon) = \sum_{m\in\{0,1\}} P(y=1,m|\underline{x},\underline{w},\varepsilon)$$

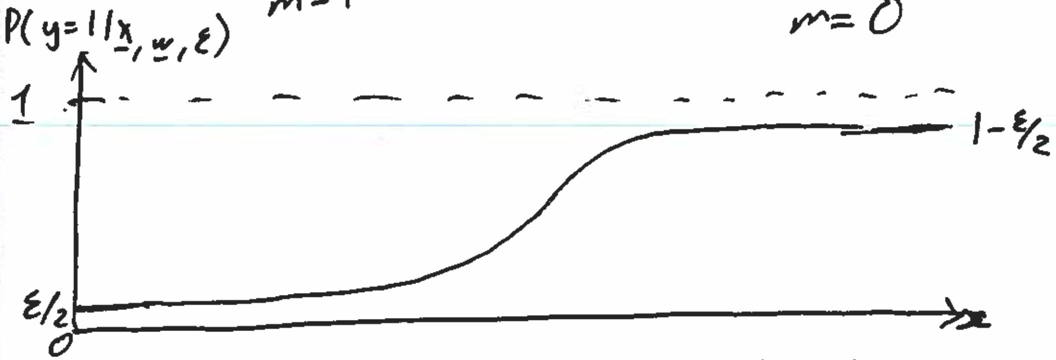
(Sum Rule)

$$= \sum_m P(y=1|\underline{x},\underline{w},\cancel{\varepsilon},m)\,P(m|\cancel{\underline{x}},\cancel{\underline{w}},\varepsilon)$$

For this model

(Product Rule)

$$= \underbrace{(1-\varepsilon)\,\sigma(\underline{w}^T\underline{x})}_{m=1} + \underbrace{\varepsilon/2}_{m=0}$$

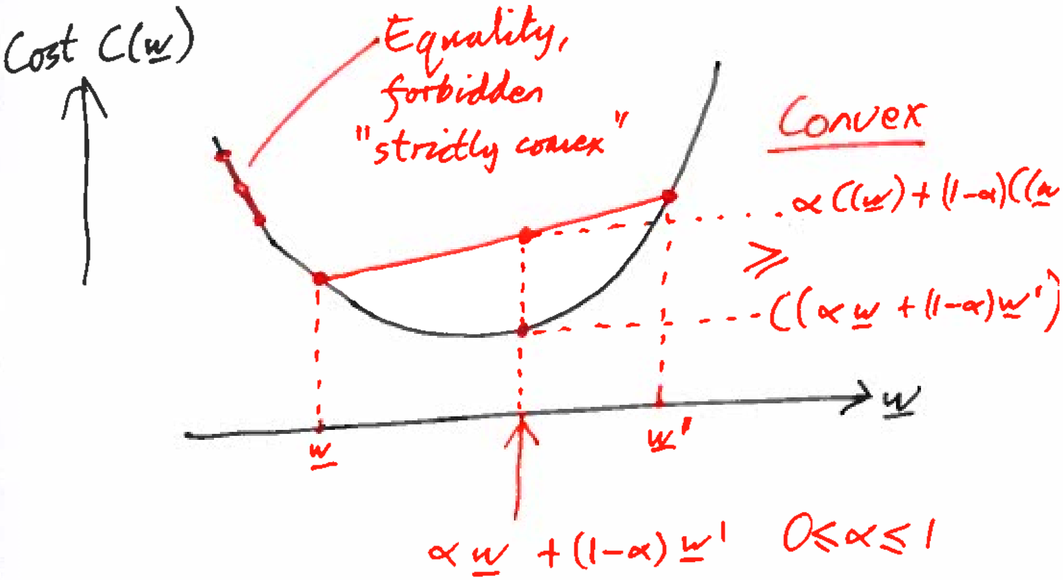


$$\nabla_{\underline{w}} \log P(y^{(n)}|\underline{x}^{(n)},\underline{w},\varepsilon) = \ldots \text{calculus/algebra}$$

$$= \frac{1}{1+\frac{1}{2}\left(\frac{\varepsilon}{1-\varepsilon}\right)\frac{1}{\sigma_n}}\,\nabla_{\underline{w}} \log \sigma_n$$

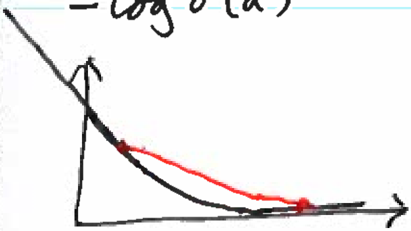
$(P(y^{(n)}|\underline{w},\underline{x}^{(n)})$
Logistic regression

Convex Optimization Problems



Logistic Regression

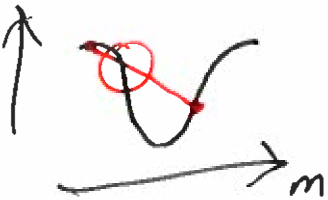
$-\log \sigma(a)$



Cost is sum convex functions

$\Rightarrow$ convex

$-N(x; m, \sigma^2)$



Robust logistic regression

NLL

How do we fit ε ?

By hand? Grid of settings $\varepsilon \in [0,1]$

maybe on a log scale

$0.1, 0.01, 0.001, \ldots$

Jointly fit

$$\underline{\theta} = \begin{bmatrix} \underline{w} \\ \varepsilon \end{bmatrix}, \quad \nabla_{\underline{\theta}} C$$

$\rightarrow$ Do SGD

Doesn't work because $\varepsilon \in [0,1]$

Trick: reparameterize model

$$\varepsilon = \sigma(b) \quad \leftarrow -\infty < b < \infty$$

Optimize $\begin{bmatrix} \underline{w} \\ b \end{bmatrix}$

$$\frac{\partial C}{\partial b} = \frac{\partial C}{\partial \varepsilon} \underbrace{\frac{\partial \varepsilon}{\partial b}}_{\varepsilon(1-\varepsilon)}$$