Recipe: fit parameters to data

**Loss and function:** \( L(y^n, f(x^n, w)) = L_n \)
- Square loss: \((y^n - f_n)^2\)
- Negative log probability: \(-\log p(y^n | x^n, w)\)

**Cost**
\[
C = \sum_n L_n \quad \text{sum over training set}
\]

**Learning direction**
\[
- \frac{1}{N} \nabla_w C = -\frac{1}{N} \sum_n \nabla_w L_n
\]

Monte Carlo approx:
\[-\nabla_w L_n \text{ for random } n\]

\(\Rightarrow\) Identify probabilistic model of \(y, p(y | x, w)\)

\(\Rightarrow\) Update parameters:
\[
w \leftarrow w + \eta \nabla_w \log p(y^n | x^n, w)
\]
S.G.D. on loss. S.G. Ascent on log likelihood
Robust Logistic Regression

Each example has binary variable

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

I will assume:

\[ p(m|\varepsilon) = \begin{cases} 1 - \varepsilon & m = 1 \\ \varepsilon & m = 0 \end{cases} \]

(eg. \( \varepsilon = 0.01 \))

Model for labels:

\[ p(y = 1 | x, w, m) = \begin{cases} \sigma(w^T x) & m = 1 \\ \frac{1}{2} & m = 0 \end{cases} \]
Need Likelihood of $w, \varepsilon$

\[ p(y=1 | x, w, \varepsilon) = \sum_{m \in \theta_{0, 13}} p(y=1, m | x, w, \varepsilon) \]  
(Sum Rule)

\[ = \sum_{m \in \theta_{0, 13}} p(y=1 | x, w, \varepsilon, m) p(m | x, w, \varepsilon) \]  
(Product rule)

For this model

\[ = \left(1 - \varepsilon\right) \sigma(w^T x) + \varepsilon \frac{1}{\nu} \]

\[ p(y=1 | x, w, \varepsilon) \]

\[ \begin{array}{c}
1 \\
\varepsilon/2 \\
\end{array} \quad \begin{array}{c}
x \\
1 - \varepsilon/2 \\
\end{array} \]

\[ \nabla_w \log p(y^{(n)} | x^{(n)}, w, \varepsilon) = \ldots \text{calculus/algbera} \]

\[ = \frac{1}{1 + \frac{1}{2} \left(\frac{\varepsilon}{1-\varepsilon}\right) \sigma_n(x)} \nabla_w \log \sigma_n \]

\[ p(y=1 | x, \varepsilon) \text{ for logistic regession} \]
How do we fit $E$?

Set it by hand?

Grid of settings $E \in [0, 1]$

Maybe on a log scale $0.1, 0.01, 0.001, ...$

For fixed $E$ the cost function is convex.

Cost $C(w)$

\[
C(\alpha w + (1-\alpha) w') \leq \alpha C(w) + (1-\alpha) C(w')
\]

$\alpha w + (1-\alpha) w'$, $0 \leq \alpha \leq 1$

Not convex
Use gradients to fit $E$

Jointly fit $\Theta = \begin{bmatrix} w \\ E \end{bmatrix}$, $D \& C$

1. Problem $C(E)$ is not convex.
   $\rightarrow$ Don't worry, do it anyway.

A real problem

$E \in [0, 1]$, $E$ is constrained.

Trick: reparameterize model

$E = \sigma(b), \quad b = \log \left( \frac{E}{1-E} \right) = \logit(E)$

$\uparrow \quad \uparrow$

$-\infty < b < \infty$

logistic sigmoid.

Derive $\frac{\partial C}{\partial b}$ and optimize $\begin{bmatrix} w \\ b \end{bmatrix}$
From the class forum:

Several RBFs fit by least squares → overfit

L2 regularized, still quite "wiggly" nowhere near data. Why?

[See also Tutorial 1 Q2c]