Learning with a single neuron

\[ \hat{y} = g(w \cdot x) \]

- Sigmoid function \( g(z) = \frac{1}{1+e^{-z}} \)
- For set of examples \((x_1, y_1), \ldots, (x_N, y_N)\)

Total loss \( \sum_{n=1}^{N} \frac{[g(w \cdot x_n) - y_n]^2}{2} \) can have exponentially \# of minima in weight space [Bu,AHW]
Want loss that is convex in $w$
Convex plus convex is convex!
Bregman Divergences lead to good loss functions

\[ g(w \cdot x) \]

transfer function \( g = \nabla G \)

\[
\int_{g^{-1}(y)}^{w \cdot x} (g(z) - y) \, dz = G(w \cdot x) - G(g^{-1}(y)) - (w \cdot x - g^{-1}(y))y
\]

\[
= \Delta_G(w \cdot x, g^{-1}(y))
\]

\[
\frac{\partial \Delta_G(w \cdot x, g^{-1}(y))}{\partial w} = (g(w \cdot x) - g(g^{-1}(y)))x
\]

\[
\hat{y}\ y
\]
Use $\Delta_G(w \cdot x, g^{-1}(y))$ as loss of $w$ on $(x, y)$
Called matching loss for $g$

<table>
<thead>
<tr>
<th>transfer func.</th>
<th>$G(z)$</th>
<th>match. loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>$g(z)$</td>
<td>$\frac{1}{2}z^2$</td>
<td>$\frac{1}{2}(w \cdot x - y)^2$</td>
</tr>
<tr>
<td>$z$</td>
<td>$\frac{1}{2}z^2$</td>
<td>square loss</td>
</tr>
<tr>
<td>$\frac{e^z}{1+e^z}$</td>
<td>$\ln(1 + e^z)$</td>
<td>$y \ln y + (1 - y) \ln(1 - y)$ + $\ln(1 + e^{w \cdot x}) - y \cdot w \cdot x$</td>
</tr>
</tbody>
</table>

In all cases, $\frac{\partial \text{match. loss}}{\partial w} = (\hat{y} - y)x$, where $\hat{y} = g(w \cdot x)$

Matching loss always convex in $w$

But convex losses **NON-ROBUST TO OUTLIERS**
Convex losses can’t handle outliers

- Any convex loss grows at least linearly
- We need the "wings to bend down", i.e. forget / give up on examples
- Non-convexity is needed to achieve robustness !!!
Fundamental example problem [LS,F]
Fundamental example problem [LS,F]

Diagram with points labeled as puller and penalizer, indicating large margins on both axes.
Fundamental example problem

[LS,F]