Classification problems

Examples:
- Text categorization
- Image recognition

A general setup

$K$ categories

**Input data:** $\{(z_i, c_i)\}, \ i = 1, 2, \ldots, n$

$z_i \in \mathbb{R}^D$ - vector of values

$c_i \in \{1, 2, \ldots, K\}$ - label
A general setup

\[ \text{Input data: } \{(z_i, c_i)\}, \ i = 1,2,\ldots,n \]
\[ z_i \in \mathbb{R}^D - \text{vector of values} \]
\[ c_i \in \{1,2,\ldots,K\} - \text{label} \]

\[ \text{Two classes: } j \text{ and NOT } j \]

\[ \text{Input data: } \{(z_i, y_i)\}, \ i = 1,2,\ldots,n \]
\[ z_i \in \mathbb{R}^D - \text{vector of values} \]
\[ y_i \in \{1,-1\} - \text{label} \]
A general setup

<table>
<thead>
<tr>
<th>K categories</th>
<th>Feature Space</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input data:</strong> {(z_i, c_i)}, (i = 1,2,\ldots,n)</td>
<td></td>
</tr>
<tr>
<td>(z_i \in \mathbb{R}^D) - vector of values</td>
<td></td>
</tr>
<tr>
<td>(c_i \in {1,2,\ldots,K}) - label</td>
<td></td>
</tr>
<tr>
<td><strong>Input data:</strong> {(x_i, y_i)}, (i = 1,2,\ldots,n)</td>
<td></td>
</tr>
<tr>
<td>(x_i = \phi(z_i) \in \mathbb{R}^d) - vector of values</td>
<td></td>
</tr>
<tr>
<td>(y_i \in {1,-1}) - label</td>
<td></td>
</tr>
</tbody>
</table>

Two classes: \(j\) and NOT \(j\)

**Input data:** \{(z_i, y_i)\}, \(i = 1,2,\ldots,n\)

\(z_i \in \mathbb{R}^D\) - vector of values

\(y_i \in \{1,-1\}\) - label
A general setup

**K categories**

Input data: \( \{(z_i, c_i)\}, i = 1,2,\ldots,n \)
- \( z_i \in R^D \) - vector of values
- \( c_i \in \{1,2,\ldots,K\} \) - label

**Feature Space**

Input data: \( \{(x_i, y_i)\}, i = 1,2,\ldots,n \)
- \( x_i = \phi(z_i) \in R^d \) - vector of values
- \( y_i \in \{1,-1\} \) - label

Two classes: \( j \) and **NOT** \( j \)

Input data: \( \{(z_i, y_i)\}, i = 1,2,\ldots,n \)
- \( z_i \in R^D \) - vector of values
- \( y_i \in \{1,-1\} \) - label

Hope: separable by a hyperplane
Text categorization


Reuters Corpus Volume 1 (RCV1):
manually categorized archive of news stories

❖ Over 800,000 stories
❖ Most stories < 1000 words (less than two A4 pages)
❖ Feature space is defined by a vocabulary of 47,152 words

Apart from the terrible memories this stirs up for me personally (coding stories through the night etc.),
I can’t find fault with your account.
– Reuters editor commenting on a draft of section 2 of this paper.
Statement of problem

Find a “good” hyperplane \( w^\top x - b = 0 \) such that

\[
\begin{align*}
\begin{cases}
  w^\top x_i - b > 0, & y_i = 1, \\
  w^\top x_i - b < 0, & y_i = -1.
\end{cases}
\end{align*}
\]

Prediction function: \( h(x, w, b) := w^\top x - b \)

Evaluate \( \text{sign} (h(x_i, w, b)) \) to attribute \( x_i \) to category 1 or -1
Support-vector machines


\[
\begin{cases}
    w^\top x_i - b \geq 1, & y_i = 1, \\
    w^\top x_i - b \leq -1, & y_i = -1.
\end{cases}
\]

\[
\frac{1}{2} \|w\|^2 \rightarrow \text{min}
\]

subject to

\[
y_i (w^\top x_i - b) \geq 1
\]
A smooth loss function

Prediction function: \[ h(x, w, b) := w^\top x - b \]

Log-loss function: \[ l(h, y) := \log(1 + \exp[-yh(x_i, w, b)]) \]

Minimization problem:
\[
\min_{(w, b) \in \mathbb{R}^d \times \mathbb{R}} \frac{1}{n} \sum_{i=1}^{n} l[h(x_i, w, b), y_i] + \frac{\lambda}{2} \|w\|^2.
\]

\textbf{positive parameter}

Tikhonov regularization
Image recognition

Sample digit images from MNIST database of handwritten characters

Example from Bottou-Curtis-Nocedal
Deep neural networks (DNNs)

Input: \( \{(x_i^{(0)}, y_i)\}, \quad i = 1, \ldots, n \)

Map to a feature space by a composition of functions:

\[
x_i^{(j)} = s \left( W_j x_i^{(j-1)} + b_j \right) \in \mathbb{R}^{d_j}, \quad j = 1, \ldots, J
\]

\( W_j = \text{matrix}, \ b_j = \text{vector}, \)

\( s(\ldots) = \text{a nonlinear activation function} \)

\( J = \text{the number of layers} \)

Example with three layers, i.e., \( J = 3 \):

\[
x^{(3)} = s \left( W_3 s \left( W_2 s \left( W_1 x^{(0)} + b_1 \right) + b_2 \right) + b_3 \right)
\]
Popular activation functions

Rectified Linear Unit (ReLU):
\[ s(x) = \max\{0, x\} \]

Sigmoid:
\[ s(x) = \frac{1}{1 + e^{-x}} \]
Optimization problem for DNNs

Vector of parameters:

\[ w := \{(W_1, b_1), (W_2, b_2), \ldots, (W_J, b_J)\} \]

Minimization problem:

\[
\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} l \left[ h(x_i, w), y_i \right],
\]

\( h(x_i, w) \) is a prediction function

\( l(h(\cdot), y) \) is a loss function
Examples of DNNs for image recognition

- **AlexNet (8 layers)** (2012)
- **ResNet (up to 152 layers)** (residual NN, matlab) (2016)

Thanks to Avi Schwarzschild, AMSC PhD student