Committee Machines

Committee machines are universal approximators.

- Static structures.
  - Ensemble averaging
  - boosting
  - bagging

- Dynamic structures.
  - Mixture of experts
  - Hierarchical mixture of experts
Experiment

Two classes distributed according to a two-dimensional Gaussian distribution with the following parameters:

Class 1:

\[ \mu_1 = [0, 0]^T \]
\[ \sigma_1^2 = 1 \]

and Class 2:

\[ \mu_2 = [2, 0]^t \]
\[ \sigma_2^2 = 4 \]
Ten neural networks (MLPs with 2 hidden neurons) were trained on the same 500 patterns. Each neural network had a different initial weight assignment.

The results:

<table>
<thead>
<tr>
<th>Classifier</th>
<th>% of correct classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Net 1</td>
<td>80.65</td>
</tr>
<tr>
<td>Net 2</td>
<td>76.91</td>
</tr>
<tr>
<td>Net 3</td>
<td>80.06</td>
</tr>
<tr>
<td>Net 4</td>
<td>80.47</td>
</tr>
<tr>
<td>Net 5</td>
<td>80.44</td>
</tr>
<tr>
<td>Net 6</td>
<td>76.89</td>
</tr>
<tr>
<td>Net 7</td>
<td>80.55</td>
</tr>
<tr>
<td>Net 8</td>
<td>80.47</td>
</tr>
<tr>
<td>Net 9</td>
<td>76.91</td>
</tr>
<tr>
<td>Net 10</td>
<td>80.38</td>
</tr>
</tbody>
</table>

The arithmetic average of the performance of all ten classifiers results in an
accuracy of 79.37%.

By voting, a classification accuracy of 80.27% is achieved (0.9% higher).

Bayes Optimal classifier for this distribution has a probability of being correct $p_c = 81.51$. 
Boosting By filtering

A weak learner is a learner which guarantees to predict better than random guessing.

Can a weak learner be used to learn arbitrarily well?

Consider the following approach using 3 classifiers:

1. The first classifier $C_1$ is generated by applying the weak learner to $N_1$ training examples.
2. $C_1$ is used as a filter: A fair coin is flipped:
   - If head turns up, a new example classified correctly by $C_1$ is ignored and an incorrectly classified example is collected for training classifier $C_2$.
   - If tail turns up, a new example classified incorrectly is ignored while a correctly classified example is collected.
   - Repeat this process until $N_1$ examples have been collected.
Generate a new classifier $C_2$ using the weak learner and the collected training examples.

3. Generate a third classifier by using the weak learner and a training sample of $N_1$ examples created by just retaining those examples which are differently classified by $C_1$ and $C_2$. 
Bagging (Boosting by resampling)

For \((i = 0; i < T; i++)\)

1. Re-sample the existing training set of \(N\) examples by bootstrapping: i.e. generate a new training sample \(S_i\) by drawing \(N\) examples randomly from the existing training set according to the uniform probability distribution among the \(N\) training examples.

2. Generate a classifier \(C_i\) from the generated sample \(S_i\).

endfor

3. Classify new examples by majority vote among the \(T\) generated classifiers.
AdaBoost

**Input:** Training sample $(x_i, d_i)_{i=1}^N$
Distribution $D$ over the sample and $T$ specifying the number of iterations

**Initialisation:** Set $D_1(i) = \frac{1}{N}$ for all $i$.

**Main Loop:** for $(n = 1; n < T; n++)$

Generate hypothesis $F_n$ based on $D_n$ and calculate error:

$\epsilon_n = \sum_{i\in\{i|F_n(x_i)\neq d_i\}} D_n(i)$
Set $\beta_n = \frac{\epsilon_n}{1-\epsilon_n}$

$D_{n+1}(i) = \frac{D_n(i)}{Z_n} \times a$, where $a = \beta_n$ if $F_n(x_i) = d_i$; $a = 1$ otherwise.
$Z_n$ is a normalisation constant.

**Output:**

$F_n(x) = \arg \max_d \sum_{n:F_n(x) = d} \log \frac{1}{\beta_n}$
AdaBoost

**Theorem** Suppose a weak learner, when called by AdaBoost, generates hypotheses with errors $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_T$ where error $\varepsilon_n$ is measured as follows:

$$
\varepsilon_n = \sum_{i:F_n(x_i) \neq d_i} D_n(i)
$$

Assume that $\varepsilon_n \leq \frac{1}{2}$ and let $\gamma_n = \frac{1}{2} - \varepsilon_n$. Then the following upper bound holds on the error of the final hypothesis:

$$
\frac{1}{N} \left| \left\{ i : F_{fin}(x_i) \neq d_i \right\} \right| \leq \Pi_{n=1}^{T} \sqrt{1 - 4\gamma^2} \leq \exp(-2 \sum_{n=1}^{T} \gamma_n^2)
$$
Arcing

Similar to AdaBoost: however, the individual training examples are re-weighted according to the following scheme for generating the \((k + 1)^{th}\) classifier:

\[
p_i = \frac{1 + m_i^4}{\sum_{j=1}^{N}(1 + m_j^4)}
\]

where \(m_i\) is the number of classifiers among classifier 1, ..., \(k\) which mis-classify example \(i\). The exponent of 4 was experimentally determined by Leo Breiman.
Most Probable Classification of New Instances

So far we have sought the most probable hypothesis given the data $D$ (also often denoted $h_{MAP}$)

Given new instance $x$, what is its most probable classification?

- $h_{MAP}(x)$ is not the most probable classification!

Consider:

- Three possible hypotheses:
  \[ P(h_1|D) = .4, \ P(h_2|D) = .3, \ P(h_3|D) = .3 \]
- Given new instance $x$,
  \[ h_1(x) = +, \ h_2(x) = -, \ h_3(x) = - \]
- What’s most probable classification of $x$?
Bayes optimal classification:

$$\arg\max_{v_j \in V} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D)$$

Example:

$$P(h_1|D) = .4, \quad P(-|h_1) = 0, \quad P(+-|h_1) = 1$$
$$P(h_2|D) = .3, \quad P(-|h_2) = 1, \quad P(+-|h_2) = 0$$
$$P(h_3|D) = .3, \quad P(-|h_3) = 1, \quad P(+-|h_3) = 0$$

therefore

$$\sum_{h_i \in H} P(+-|h_i)P(h_i|D) = .4$$
$$\sum_{h_i \in H} P(-|h_i)P(h_i|D) = .6$$

and

$$\arg\max_{v_j \in V} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D) = -$$
Mixture of Experts

An associate Gaussian mixture model is given by the following probability density function for $D$ (being the set of possible output values) for all $d \in D$:

$$f_D(d|x, k, \theta) = \sum_{k=1}^{K} g_k f_D(d|x, k, \theta)$$

$$= \frac{1}{\sqrt{2\pi}} \sum_{k=1}^{K} g_k \exp(-\frac{1}{2}(d - \gamma_k)^2)$$

with $x$ being the input vector, $\theta$ being the parameter vector (weights), $g_k$ being the gating value for expert $k$, and $y_k$ being the value produced by expert $k$. 