Unsupervised learning of distributions on binary vectors using two layer networks
Yoav Freund, David Haussler, 1994

Presented By: Yaoguang Zhai
OBJECTIVE

- Q: find correlations of bits in binary vectors
  - $\bar{x}_i \in \{\pm 1\}^n$, $n > 50$
  - Independently generated vectors
  - E.g., in binary image of handwritten digits

- A: (at 1994)
  - PCA and GMM
  - Hopfield Nets, BM, Bayes Networks, Markov random field
  - ……
  - Restricted Boltzman Machine (RBM), a.k.a. Influence Combination Model in this paper
RECAP. OF HOPFIELD NET

The global energy:
\[ E = -\sum_i s_i b_i - \sum_{i<j} s_i s_j w_{ij} \]

- Memorize a configuration by creating an energy minimum
RECAP. OF BOLTZMAN MACHINE

- Energy based on joint distribution of visible and hidden units.
- Probability is linear to \( \exp(-E) \)

\[
p(v, h) \propto e^{-E(v, h)}
\]

- Update the network with MC-MC
RESTRICTED BOLTZMAN MACHINE: STRUCTURE

- Bipartite graph
- Input vector
- Hidden units
- Weighted connections

Diagram:

- Input nodes connected to hidden nodes through weighted connections.
RESTRICTED BOLTZMAN MACHINE: INTUITIVE

• Inputs are NOT independent

• E.g., one Sunday afternoon ...

Go to Café?

Fred’s 3yr Kid
Fred’s Wife
Fred
Fred’s Ex-Wife
Fred’s Ex-GF

Café in the town

Coffee House
RESTRICTED BOLTZMAN MACHINE: BINARY MODEL

- $x_i \in \{-1, +1\}, i = 1, ..., n$
- $h_j \in \{0, +1\}, j = 1, ..., m$
- $w_i^{(j)}, \theta^{(j)} \in R$, $\phi_B = \{(w^{(j)}, \theta^{(j)})\}$
- $E \left( \bar{x}, \bar{h} \middle| \phi_B \right) = -\sum_{j=1}^{m}(w^{(j)} \cdot \bar{x} + \theta^{(j)})h_j$
- $Pr \left( \bar{x}, \bar{h} \middle| \phi_B \right) = \frac{1}{Z_B} \cdot \exp(-E \left( \bar{x}, \bar{h} \middle| \phi_B \right))$
- $Pr(\bar{x} \mid \phi_B) = \sum_{\bar{h}} Pr(\bar{x}, \bar{h} \mid \phi_B)$, mixture of $2^m$ Bernoulli dist.
- For a fix $\bar{h} = (h_1, h_2, ..., h_m)$,
  $$Pr \left( \bar{x} \mid \bar{h}, \phi_B \right) = \frac{Pr \left( \bar{x}, \bar{h} \middle| \phi_B \right)}{Pr \left( \bar{h} \middle| \phi_B \right)} = C \cdot \prod_{i} \exp(k_i x_i)$$
- Any distribution over $\{0, +1\}^n$ can be represented by a RBM.
LEARNING ALGORITHM ON BINARY MODEL

• Maximum likelihood estimation by gradient decent
  • Finding $\phi_B$ to maximum the likelihood of samples:
  • $\log - likelihood = \prod_{\tilde{x} \in S} \ln(Pr(\tilde{x}, \tilde{h}|\phi_B)) = \frac{1}{Z_B} \exp(-E(\tilde{x}, \tilde{h}|\phi_B))$

• OPT1: SGD

• OPT2: Calculate explicitly the gradient and use batched GD:
  • Time complexity: $O(|S|n + 2^m)$ as $2^m$ of energies are needed

• OPT3: as in 2, but only calculate dominate terms in the energy
  • Assume small number of terms are activated at the same time
  • Time complexity: $O(m^k)$ as more than $k$ hidden units activated is negligible.
LEARNING ALGORITHM: PROJECTION PURSUIT

• Greedy algorithm that generates hidden units one by one
• Main points:
  • Search “best“ projection from high-dim to low-dim
    • Gaussian: 😞
    • Non-Gaussian : 😲
  • At one time, find (greedily) the “best“ projection
  • In next iteration, expand the projection space dim
  • E.g, PCA and SVD are special cases of PP
LEARNING ALGORITHM: PROJECTION PURSUIT

• Scheme:
  • Recall $P(\tilde{x}) = \frac{1}{Z_B} \prod (e^{\theta_i + \tilde{w}_i \cdot \tilde{x}})$
  • Initiate $p_0(\tilde{x})$
  • $G$ is a family of functions from $R$ to $R$, $G = \{ g: R \to R | g(t) = \frac{1}{Z} (e^{t||\tilde{w}_i||_2}) ; Z \in R \}$
  • $A$ is a set of unit length vector in $R^n$, $\tilde{a}_i = \frac{\tilde{w}_i}{||\tilde{w}_i||_2}$
  • $N$th order projection:

$$PP_m = \left\{ \frac{1}{Z} p_0(\tilde{x}) \prod_{i=1}^{m} g_i (\tilde{a}_i \cdot \tilde{x}) | \tilde{a}_i \in A; g_i \in G; Z = \int_{R^n} p_0(\tilde{x}) \prod_{i=1}^{m} g_i (\tilde{a}_i \cdot \tilde{x}) d\tilde{x} \right\}$$

• find $p_i$ that maximally increasing log-likelihood:

$$LL(p|S) = \sum_{\tilde{x} \in S} \ln p(\tilde{x}), p \in PP_m$$
RESTRICTED BOLTZMANN MACHINE: REAL-VALUE

- $x_i \in R, i = 1, \ldots, n$
- $h_j \in \{0, +1\}, j = 1, \ldots, m$
- $w_i^{(j)}, \theta^{(j)} \in R$, $\phi_R = \{(w^{(j)}, \theta^{(j)})\}$
- $E \left( \tilde{x}, \tilde{h} \middle| \phi_R \right) = - \sum_{j=1}^{m} (w^{(j)} \cdot \tilde{x} + \theta^{(j)}) h_j + \frac{1}{2} \| \tilde{x} \|_2^2$
- $Pr \left( \tilde{x}, \tilde{h} \middle| \phi_R \right) = \frac{1}{Z_R} \exp(-E \left( \tilde{x}, \tilde{h} \middle| \phi_R \right) )$
- $Pr(\tilde{x}|\phi_R) = \sum Pr(\tilde{x}, h|\phi_R)$, mixture of $2^m$ Gaussian.
- For a fix $\tilde{h} = (h_1, h_2, \ldots, h_m)$,

$$Pr \left( \tilde{x} \middle| \tilde{h}, \phi_R \right) = \frac{Pr \left( \tilde{x}, \tilde{h} \middle| \phi_R \right)}{Pr \left( \tilde{h} \middle| \phi_R \right)} = D \cdot \prod_{i} \exp(k_i x_i - x_i^2)$$
LEARNING ALGORITHM ON REAL-VALUED MODEL

- As in binary model
- But when \( m \) is large:

\[
\Pr \left( \hat{x} \mid \phi_R \right) = \frac{\exp \left( \sum_{i=1}^{m} h_i \theta_i + \frac{1}{2} \left\| \sum_{i=1}^{m} h_i \tilde{w}^i \right\|_2^2 \right)}{\sum_{\hat{x} \in \{0,1\}^m} \exp \left( \sum_{i=1}^{m} h_i \theta_i + \frac{1}{2} \left\| \sum_{i=1}^{m} h_i \tilde{w}^i \right\|_2^2 \right)}
\]

- We would like to find \( \hat{x} \) that the probability is large:
  - Let \( \Omega = (\tilde{w}^1, \ldots, \tilde{w}^m)^T, \tilde{\theta} = (\theta_1, \ldots, \theta_m)^T \)
  - The probability is proportional to

\[
g \left( \hat{x} \right) = \frac{1}{2} \left\| \hat{x}^T \Omega + \tilde{\theta}^T (\Omega \Omega^T)^{-1} \Omega \right\|_2^2 - \frac{1}{2} \tilde{\theta}^T (\Omega \Omega^T)^{-1} \tilde{\theta}
\]

- Transformed to finding a subset that is furthest away from a vector when \( m \) is large.
LEARNING ALGORITHM: EXPECTATION-MAXIMIZATION

• For single hidden node only
• Iteratively improve the estimates by max likelihood
• Scheme:
  • Initial the parameter setting \((\vec{w}_{\text{init}}, \vec{\theta}_{\text{init}})\)
  • For a step that with given samples \(S = \langle \vec{x}_1, ..., \vec{x}_N >\), update \((\vec{w}_{\text{old}}, \vec{\theta}_{\text{old}})\) with:
    \[
    \vec{w}_{\text{opt}} = \frac{\hat{E}(h\vec{x})}{\bar{E}(h)}
    \]
    \[
    \vec{\theta}_{\text{opt}} = -\ln \left( \frac{1 - \hat{E}(h)}{\bar{E}(h)} \right) - \frac{1}{2} \|\vec{w}_{\text{opt}}\|_2^2
    \]
  • Repeat until no significant differences between old and new parameter sets.
Figure 4.8: The weight vector, or image templates, found by the mixture model