Dense Associative Memory for Pattern Recognition

Dmitry Krotov[1] John J Hopfield[2]

[1]Institute for Advanced Study

[2]Princeton University

NIPS, 2016
Presenter: Jack Lanchantin

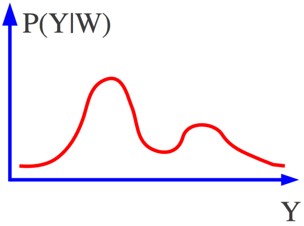
Outline

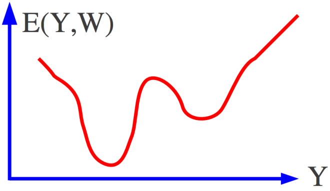
Outline

1. Introduction
   - Energy-based Learning
   - Associative Memory
2. Methods and Results
   - Classification with Dense Associative Memory
   - Duality b/w Neural Nets and Associative Memory
3. Conclusions

Energy-based Learning

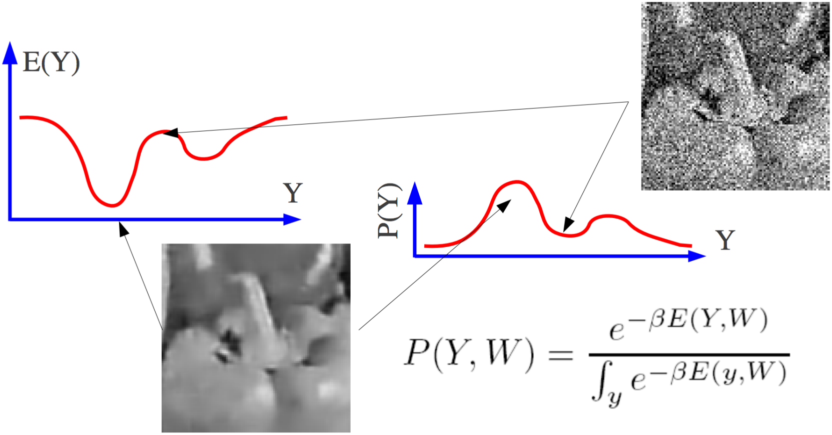
- **Energy function: viewed as a negative log probability density**
- **Probabilistic View:**
  - Produce a probability density function that:
  - has high value in regions of high sample density
  - has low value everywhere else (integral = 1).
- **Energy-Based View:**
  - produce an energy function E(Y,W) that:
  - has low value in regions of high sample density
  - has high(er) value everywhere else





Energy-based Learning

- **Make the energy around training samples low**
- **Make the energy everywhere else higher**



$$P(Y,W) = \frac{e^{-\beta E(Y,W)}}{\int_y e^{-\beta E(y,W)}}$$

Outline

Associative Memory

- N binary neurons with values ± 1
- A configuration of all neurons is denoted by vector σ.
- The model stores K memory vectors, denoted by ξ^μ
- The model is defined by an Energy function:

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sigma_i T_{i,j}\sigma_j, \quad T_{i,j} = \sum_{\mu=1}^{K}\xi_i^\mu \xi_j^\mu \tag{1}$$

Associative Memory

- N binary neurons with values $\pm$ 1
- A configuration of all neurons is denoted by vector σ.
- The model stores K memory vectors, denoted by ξ^μ
- The model is defined by an Energy function:

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sigma_i T_{i,j}\sigma_j, \quad T_{i,j} = \sum_{\mu=1}^{K}\xi_i^\mu \xi_j^\mu \tag{1}$$

- This energy model gets confused when many memories (i.e. large K) are stored because several memories produce contributions to the energy which are of the same order.

Dense Associative Memory: Higher Order Interactions in Energy Function

- New energy function:

$$E = -\sum_{u=1}^{K} F\Big(\sum_{i=1}^{N} \xi_i^u \sigma_i\Big) \tag{2}$$

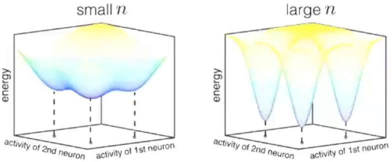
- Polynomial energy function: $F(x) = x^n$
- Rectified energy function: $F(x) = \begin{cases} x^n, & x \geq 0 \\ 0, & x < 0 \end{cases}$

Dense Associative Memory: Higher Order Interactions in Energy Function

- New energy function:

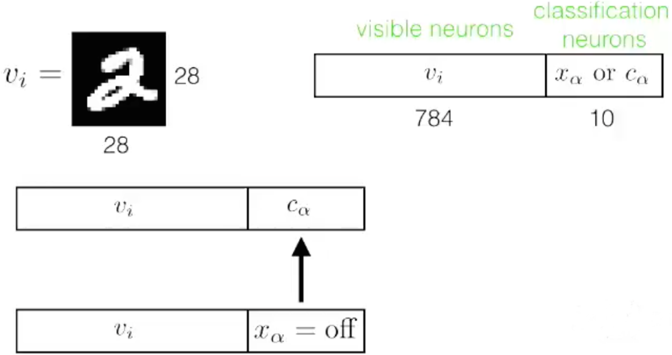
$$E = -\sum_{u=1}^{K} F\Big(\sum_{i=1}^{N} \xi_i^u \sigma_i\Big) \qquad (2)$$

- Polynomial energy function: $F(x) = x^n$
- Rectified energy function: $F(x) = \begin{cases} x^n, & x \geq 0 \\ 0, & x < 0 \end{cases}$
- As n increases, more memories can be packed into the same space because each term becomes sharper



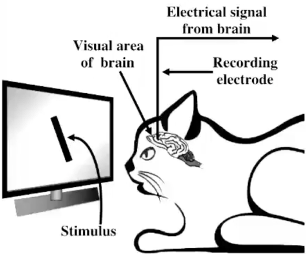
Outline

Classification with Dense Associative Memory

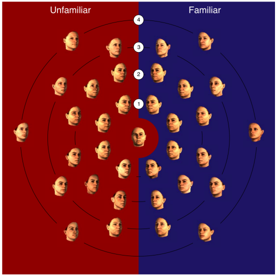


$$E = -\sum_{u=1}^{K} F\Big(\sum_{i}^{N} \xi_i^u v_i + \sum_{\alpha}^{10} \xi_\alpha^u v_\alpha\Big), \qquad F(x) = \begin{cases} x^n, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3}$$

Human Pattern Recognition: Feature Matching vs Prototype

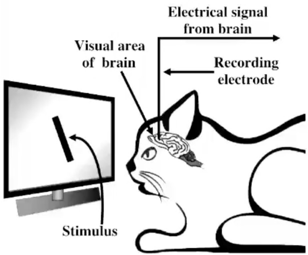


(a) Feature Matching
(Hubert & Wiesel 1959)

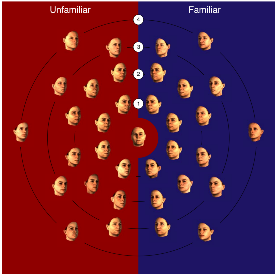


(b) Prototype Effect
(Solso & McCarthy, 1981)

Human Pattern Recognition: Feature Matching vs Prototype



(c) Feature Matching
(Hubert & Wiesel 1959)



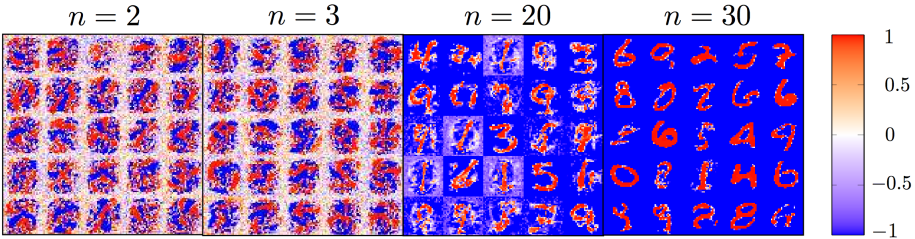
(d) Prototype Effect
(Solso & McCarthy, 1981)

- Instead of decomposing the input into a set of features and then matching these features, the whole image can be compared to memorized prototypes and classification can be made based on the similarity between the test image and prototypes.

Transitions

$$E = -\sum_{u=1}^{K} F\Big(\sum_{i}^{N} \xi_i^u v_i + \sum_{\alpha}^{10} \xi_\alpha^u v_\alpha\Big)$$

$$F(x) = \begin{cases} x^n, & x \geq 0 \\ 0, & x < 0 \end{cases}$$



1.5% 1.4% 1.6% 1.8%

feature detectors

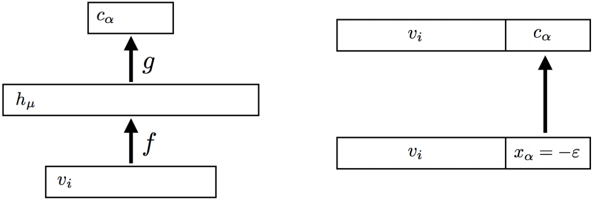
prototype detectors

Outline

Duality b/w Neural Nets and Associative Memory

$$\text{NN: } h_u = f\Big(\sum_{i}^{N} \xi_i^u v_i\Big)$$

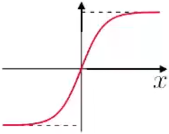
$$\text{DAM: } E = -\sum_{u=1}^{K} F\Big(\sum_{i}^{N} \xi_i^u v_i + \sum_{\alpha}^{10} \xi_\alpha^u v_\alpha\Big)$$

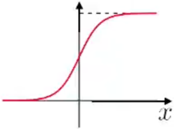


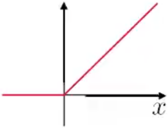
$$f(x) = F'(x) \tag{4}$$

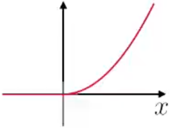
Duality with Activation Functions

AM: $n = 1$	$n = 1$	$n = 2$ standard Hopfield net	n DAM
$f(x) = \tanh(x)$	$f(x) = \frac{1}{1+e^{-x}}$	$f(x) = \mathrm{ReLU}$	$f(x) = \mathrm{ReP}_{n-1}$









ReLU
feature
small n

ReP_n for $n \approx 20$
prototype
large n

Conclusions

- Duality between Associative Memory and feed-forward neural nets.
- Feature to prototype transition in feed forwards nets can be induced by changing the power of the activation functions