

Title: PSI 2018/2019 - Machine Learning - Lecture 12

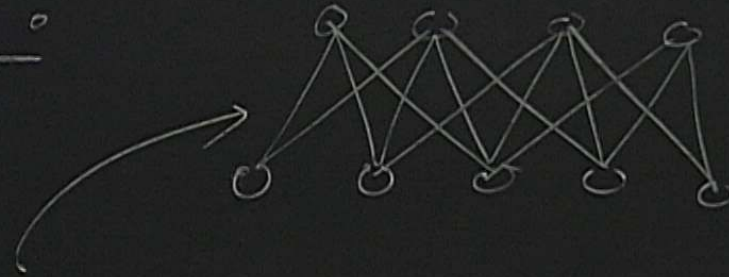
Speakers: Roger Melko

Collection: PSI 2018/2019 - Machine Learning (Hayward Sierens)

Date: April 09, 2019 - 9:00 AM

URL: <http://pirsa.org/19040007>

RBM :



$n$  hidden units

$m$  visible units

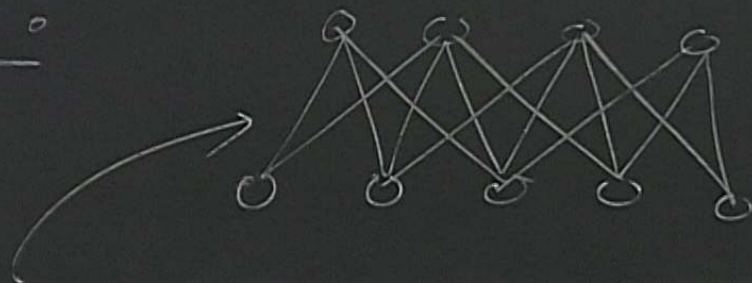
fully  
connected between layers

joint distribution  $p_{\lambda}(\vec{v}, \vec{h}) = \frac{1}{Z_{\lambda}} e^{-E(\vec{v}, \vec{h})}$

$$E(\vec{v}, \vec{h}) = - \sum_{i=1}^m b_i v_i - \sum_{j=1}^n c_j h_j - \sum_{i,j} w_{ij} v_i h_j$$

$$Z_{\lambda} = \sum_{\vec{v}, \vec{h}} e^{-E} \quad (\text{partition function})$$

RBM:



$n$  hidden units  
 $m$  visible units

fully  
connected between layers

$$\lambda = (W, \vec{b}, \vec{c})$$

Today: Training an RBM

$$\lambda = (W, b, z)$$

$$Z_\lambda = \sum_{\vec{x}} e^{-\dots} \quad (\text{partition function})$$

# Today: Training an RBM

## Cost function

$$D_{KL}(P \parallel P_\lambda) = \sum_{\vec{x}} P(\vec{x}) \log \frac{P(\vec{x})}{P_\lambda(\vec{x})}$$

target (unknown) distribution  $\uparrow$   
 RBM distribution  $\downarrow$

KL divergence.

$D_{KL} > 0$  and  
 $D_{KL} = 0$  iff  $P = P_\lambda$



note

$$D_{KL} = \underbrace{\sum_{\vec{x}} P(\vec{x}) \log P(\vec{x})}_{\text{entropy of } P, \text{ data } D} - \underbrace{\sum_{\vec{x}} P(\vec{x}) \log P_\lambda(\vec{x})}_{\text{depends on } \lambda}$$

As always, approximate the "trace" with training samples drawn from P

$D_{KL}(P \parallel P_\lambda) = \sum_{\vec{x}} P(\vec{x}) \log \frac{P(\vec{x})}{P_\lambda(\vec{x})}$  KL divergence.  $D_{KL} = 0 \iff P = P_\lambda$   
 target (unknown) distribution  $\rightarrow$  RBM distribution

note  $D_{KL} = \underbrace{\sum_{\vec{x}} P(\vec{x}) \log P(\vec{x})}_{\text{entropy of } P, \text{ data } D} - \underbrace{\sum_{\vec{x}} P(\vec{x}) \log P_\lambda(\vec{x})}_{\text{depends on } \lambda}$  } As always, approximate the "trace" with training samples drawn from  $P$

$$D = \{\vec{x}\}, \quad P_{\text{data}}(\vec{x}) = \frac{1}{|D|} \sum_{\vec{x}_i \in D} \delta_{\vec{x}, \vec{x}_i}, \quad D_{KL} \approx -H_D - \frac{1}{|D|} \sum_{\vec{x} \in D} \log P_\lambda(\vec{x})$$

We typically drop the entropy of the dataset  $H_D$ , what remains is Maximum Likelihood Estimation

Let's define  $C_\lambda = -\frac{1}{|D|} \sum_{\vec{x} \in D} \log P_\lambda(\vec{x})$  and gradient descent  $\lambda \leftarrow \lambda - \eta \nabla_\lambda C_\lambda$   $\eta = \text{learning rate (constant)}$   
 for RBM  $\rightarrow$

Recall  $\mathcal{D} = \{ \vec{v}_1, \vec{v}_2, \vec{v}_3, \dots, \vec{v}_M \}$ ,  $M = |\mathcal{D}|$ ,  $\rightarrow$  lets pull out one  $\vec{v}_k = \vec{v}$

For that single training example  
$$\log p_{\lambda}(\vec{v}) = \log \frac{1}{Z} \sum_{\vec{h}} e^{-E(\vec{v}, \vec{h})} = \log \sum_{\vec{h}} e^{-E} - \log \sum_{\vec{h}} e^{-E}$$

Recall  $\mathcal{D} = \{ \vec{v}_1, \vec{v}_2, \vec{v}_3, \dots, \vec{v}_M \}$ ,  $M = |\mathcal{D}|$ .  $\rightarrow$  lets pull out one  $\vec{v}_k = \vec{v}$

For that single training example

$$\log p_{\lambda}(\vec{v}) = \log \frac{1}{Z} \sum_{\vec{h}} e^{-E(\vec{v}, \vec{h})} = \log \sum_{\vec{h}} e^{-E} - \log \sum_{\vec{h}} e^{-E}$$

the gradient  $\frac{\partial \log p_{\lambda}(\vec{v})}{\partial \lambda} = \frac{\partial}{\partial \lambda} \left( \log \sum_{\vec{h}} e^{-E} \right) - \frac{\partial}{\partial \lambda} \left( \log \sum_{\vec{h}} e^{-E} \right)$

for that single training example

$$\log p_{\lambda}(\vec{v}) = \log \frac{1}{Z} \sum_h e^{-E(\vec{v}, h)} = \log \sum_h e^{-E} - \log \sum_{\vec{v}, h} e^{-E}$$

the gradient

$$\frac{\partial \log p_{\lambda}(\vec{v})}{\partial \lambda} = \frac{\partial}{\partial \lambda} \left( \log \sum_h e^{-E} \right) - \frac{\partial}{\partial \lambda} \left( \log \sum_{\vec{v}, h} e^{-E} \right)$$

$$= \frac{-1}{\sum_h e^{-E}} \sum_h e^{-E} \cdot \frac{\partial E}{\partial \lambda} + \sum_{\vec{v}, h} \frac{1}{\sum_{\vec{v}, h} e^{-E}} e^{-E} \frac{\partial E}{\partial \lambda} \quad \text{dropped } \vec{v}, h$$

note

Using the expression for the conditional probability distribution

$$p(h|\vec{v}) = \frac{p(\vec{v}, h)}{p(\vec{v})} = \frac{\frac{1}{Z} e^{-E}}{\frac{1}{Z} \sum_h e^{-E}} = \frac{e^{-E}}{\sum_h e^{-E}}$$

$$\frac{\partial \log p(\vec{v})}{\partial \lambda} = - \sum_h p(h|\vec{v}) \frac{\partial E}{\partial \lambda} + \sum_{\vec{v}, h} p(\vec{v}, h) \frac{\partial E}{\partial \lambda}$$

↑↑↑↑↑↑↑

Keep in mind the gradients of  $E$

are easy  $\frac{\partial E}{\partial W_{ij}} = -v_i h_j$  etc.

Now recall the graph structure  
of the RBM

diagonal of  $E$

$\sigma_i h_i$ ; etc.


graph structure

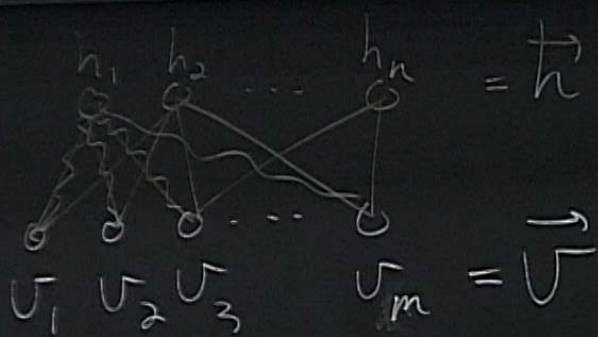
RBM

$$\begin{array}{cccc} h_1 & h_2 & \dots & h_n \\ 0 & 0 & \dots & 0 \\ \sigma_1 & \sigma_2 & \sigma_3 & \sigma_m \end{array}$$

$$\frac{2 \log P(\vec{v})}{2\lambda} = - \sum_{\vec{h}} p(\vec{h}|\vec{v}) \frac{\partial E}{\partial \lambda} + \sum_{\vec{v}, \vec{h}} p(\vec{v}, \vec{h}) \frac{\partial E}{\partial \lambda}$$

$\uparrow \quad \uparrow \quad \uparrow \quad \uparrow \quad \uparrow \quad \uparrow \quad \uparrow$





still a single  
training example.

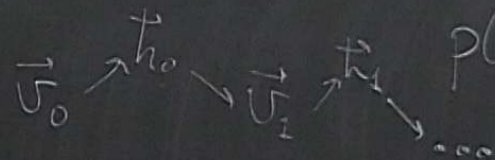
Each hidden  $h_i$  is independent of the other  $h_{-i}$

$$p(\vec{h} | \vec{v}) = \prod_i p(h_i | \vec{v})$$

$$p(\vec{v} | \vec{h}) = \prod_i p(v_i | \vec{h})$$

The conditional prob. can be calculated analytically.  $p(h_3=1 | \vec{v}) = \sigma\left(\sum_{j=1}^m W_{3j} v_j + a_3\right)$

Note: the "restriction" on weights allows  
"Block Gibbs" sampling



$$p(v_i=1 | \vec{h}) = \sigma\left(\sum_{j=1}^n W_{ij} h_j + b_i\right)$$

Back to  $\oplus$ : Look at the first term for  $\lambda = W_{ij}$

$$\sum_{k \in \mathcal{K}} p(k | \vec{u}) \frac{2E}{2w_{ij}} = \sum_{k \in \mathcal{K}} p(k | \vec{u}) u_j h_i$$

and factor  $p(\vec{h} | \vec{v}) = p(h_i | \vec{v}) p(h_{-i} | \vec{v})$

Note: The restriction on weights allows "Block Gibbs" sampling  $\vec{v}_0 \rightarrow h_0 \rightarrow \vec{v}_1 \rightarrow h_1 \rightarrow \dots$

Back to  $\otimes$ : Look at the first term for  $\lambda = W_{ij}$

$$p(v_i = 1 | \vec{h}) = \sigma\left(\sum_{j=1} W_{ij} h_j + b_i\right)$$

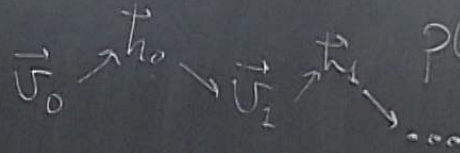
$$\sum_{\vec{h}} p(\vec{h} | \vec{v}) \frac{\partial E}{\partial W_{ij}} = \sum_{\vec{h}} p(\vec{h} | \vec{v}) v_j h_i$$

and factor  $p(\vec{h} | \vec{v}) = p(h_i | \vec{v}) p(\vec{h}_{-i} | \vec{v})$

$$= \sum_{h_i} \sum_{\vec{h}_{-i}} p(h_i | \vec{v}) p(\vec{h}_{-i} | \vec{v}) h_i v_j = \sum_{h_i} p(h_i | \vec{v}) h_i v_j \underbrace{\sum_{\vec{h}_{-i}} p(\vec{h}_{-i} | \vec{v})}_{(p(h_i=0 | \vec{v}) + p(h_i=1 | \vec{v})) \dots}$$

The conditional prob. can be calculated analytically.  $p(h_j=1|\vec{v}) = \sigma\left(\sum_{i=1}^n w_{ij} v_i + c_j\right)$

Note: the "restriction" on weights allows  
"Block Gibbs" sampling



$$p(v_i=1|h) = \sigma\left(\sum_{j=1}^n w_{ij} h_j + b_i\right)$$

Back to  $\otimes$ : Look at the first term for  $\lambda = w_{ij}$

$$\begin{aligned} \sum_{h_i} p(h_i|\vec{v}) \frac{\partial E}{\partial w_{ij}} &= \sum_{h_i} p(h_i|\vec{v}) v_j h_i \quad \text{and factor } p(h_i|\vec{v}) = p(h_i|\vec{v}) p(h_{-i}|\vec{v}) \\ &= \sum_{h_i} \sum_{h_{-i}} p(h_i|\vec{v}) p(h_{-i}|\vec{v}) h_i v_j = \sum_{h_i} p(h_i|\vec{v}) h_i v_j \underbrace{\sum_{h_{-i}} p(h_{-i}|\vec{v})}_{(p(h_{-i}=0|\vec{v}) + p(h_{-i}=1|\vec{v})) \dots = 1} \\ &= \sum_{h_i} p(h_i|\vec{v}) h_i v_j \end{aligned}$$

$$= \sum_{h_i} \sum_{h_{-i}} p(h_i | \vec{U}) p(h_{-i} | \vec{U}) h_i U_j = \sum_{h_i} p(h_i | \vec{U}) h_i U_j \underbrace{\sum_{h_{-i}} p(h_{-i} | \vec{U})}_{(p(h_i=0|\vec{U}) + p(h_i=1|\vec{U}))}$$

$$= \sum_{h_i} p(h_i | \vec{U}) h_i U_j$$

$$= (p(h_i=0|\vec{U}) \cdot 0 + p(h_i=1|\vec{U}) \cdot 1) U_j = p(h_i=1|\vec{U}) U_j$$

$$\begin{aligned}
 &= \sum_{h_i} \sum_{\tilde{h}_i} p(h_i | \vec{v}) p(\tilde{h}_i | \vec{v}) h_i v_j = \sum_{h_i} p(h_i | \vec{v}) h_i v_j \underbrace{\sum_{\tilde{h}_i} p(\tilde{h}_i | \vec{v})}_{(p(h_i=0 | \vec{v}) + p(h_i=1 | \vec{v}))} \\
 &= \sum_{h_i} p(h_i | \vec{v}) h_i v_j
 \end{aligned}$$

$$= (p(h_i=0 | \vec{v}) \cdot 0 + p(h_i=1 | \vec{v}) \cdot 1) v_j = p(h_i=1 | \vec{v}) v_j$$

What about the 2<sup>nd</sup> term in  $\otimes$ ?  $\sum_{\vec{v}, \vec{h}} p(\vec{v}, \vec{h}) \frac{\partial E}{\partial \lambda}$  if you write  $p(\vec{v}, \vec{h}) = p(\vec{v}) p(\vec{h} | \vec{v})$  or  $p(\vec{h}) p(\vec{v} | \vec{h})$

i.e.  $\sum_{\vec{v}, \vec{h}} p(\vec{v}, \vec{h}) = \sum_{\vec{v}} p(\vec{v}) \sum_{\vec{h}} p(\vec{h} | \vec{v}) \frac{\partial E}{\partial \lambda}$

← RBM simplifies the "inner" trace.  
 - but the outer trace is still over  $2^m$  states

Recall

$$C_\lambda = -\frac{1}{|\mathcal{D}|} \sum_{\vec{v} \in \mathcal{D}} \log P_\lambda(\vec{v})$$

$$\nabla_\lambda C_\lambda = -\frac{1}{|\mathcal{D}|} \sum_{\vec{v} \in \mathcal{D}} \frac{2}{2\lambda} \log P_\lambda(\vec{v})$$

Recall

$$C_\lambda = -\frac{1}{\|\mathcal{D}\|} \sum_{\vec{v} \in \mathcal{D}} \log p_\lambda(\vec{v})$$

$$\nabla_\lambda C_\lambda = -\frac{1}{\|\mathcal{D}\|} \sum_{\vec{v} \in \mathcal{D}} \frac{\partial}{\partial \lambda} \log p_\lambda(\vec{v})$$

$$\frac{\partial \log p_\lambda(\vec{v})}{\partial \lambda} = -\sum_k p(k|\vec{v}) \frac{\partial E}{\partial \lambda} + \sum_{\vec{v} \in \mathcal{H}} p(\vec{v}|k) \frac{\partial E}{\partial \lambda}$$

$$\text{ie } \nabla_\lambda C_\lambda = \left\langle \frac{\partial E}{\partial \lambda} \right\rangle_{p(\lambda|\vec{v})} - \left\langle \frac{\partial E}{\partial \lambda} \right\rangle_{p(\vec{v}|k)} \quad \lambda \leftarrow \lambda - \eta \nabla_\lambda C_\lambda$$

ie  $\nabla_{\lambda} C_{\lambda} = \left\langle \frac{\partial E}{\partial \lambda} \right\rangle_{\rho(\vec{v}, \vec{h})} - \left\langle \frac{\partial E}{\partial \lambda} \right\rangle_{\rho(\vec{v}, \vec{h})}$

16  $\frac{\partial C_{\lambda}}{\partial W_{ij}} = \left\langle V_i h_j \right\rangle_{\rho} - \left\langle V_i h_j \right\rangle_{\rho(\vec{v}, \vec{h})}$

↑  $W_{ij}$  increases according to the data

↓ decreases when generated by the model

"positive phase"

"negative phase"

How to calculate "negative phase"?

- expectation value of a joint dist.

$$\langle U_{ij} \rangle_{p(\vec{w}, \vec{h})} \sim \frac{1}{N_{\text{mc}}} \sum_{\vec{w}, \vec{h}} (U_{ij})$$

produced by RBM

Fudging  $\rightarrow$  approximate  
this with known MCMC step