

# Ограничена Болцман Машина (restricted boltzmann machine)

Урош Стегић 447/2016

Математички факултет

26. септембар 2018.

# Садржај

Увод

Процес учења

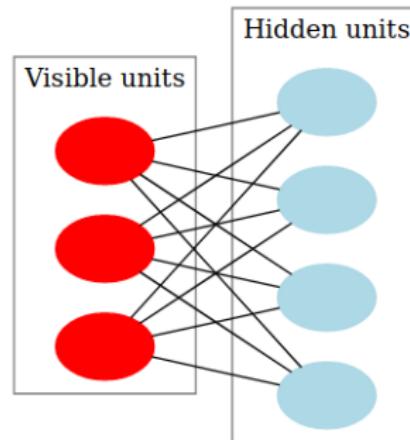
Закључак

## Врста модела

- Ненадгледано учење
- Генеративни модел
- Параметризован модел
- Пробабилистички графички модел

## Компоненте

- видљиви слој -  $v$
- скривени (латентни) слој -  $h$
- параметри модела -  $\theta$



## Опис проблема

- проналажење густине заједничке расподеле:  $p(v, h)$
- енергија:  $E(v, h; \theta) = -\sum_{r=1}^R v_r b_r - \sum_{k=1}^K h_k c_k - \sum_{k=1}^K \sum_{r=1}^R v_r h_k W_{rk}$
- болцманова расподела:  $p(v, h; \theta) = \frac{1}{Z(\theta)} e^{-E(v, h, \theta)}$
- функција партиционисања:  $Z(\theta) = \sum_v \sum_h e^{-E(v, h, \theta)}$

## Оптимизациони циљ

- Минимизација негативног логаритма веродостојности:  
$$-\frac{1}{N} \sum_{i=1}^N \log p(v_i | \theta)$$
- Трик: маргинална расподела  $p(v_i | \theta)$  преко слободне енергије  $F(v; \theta)$
- Градијент:  $\mathbb{E}_{\mathbf{h}} \left[ \frac{\partial E(v^{(i)}, h)}{\partial \theta} | v^{(i)} \right] - \mathbb{E}_{\mathbf{v}, \mathbf{h}} \left[ \frac{\partial E(v, h)}{\partial \theta} \right]$

# Contrastive Divergence

---

**Algorithm 27.3:** CD-1 training for an RBM with binary hidden and visible units

---

```
1 Initialize weights  $\mathbf{W} \in \mathbb{R}^{R \times K}$  randomly;  
2  $t := 0$ ;  
3 for each epoch do  
4    $t := t + 1$  ;  
5   for each minibatch of size  $B$  do  
6     Set minibatch gradient to zero,  $\mathbf{g} := \mathbf{0}$  ;  
7     for each case  $\mathbf{v}_i$  in the minibatch do  
8       Compute  $\boldsymbol{\mu}_i = \mathbb{E}[\mathbf{h}|\mathbf{v}_i, \mathbf{W}]$ ;  
9       Sample  $\mathbf{h}_i \sim p(\mathbf{h}|\mathbf{v}_i, \mathbf{W})$ ;  
10      Sample  $\mathbf{v}'_i \sim p(\mathbf{v}|\mathbf{h}_i, \mathbf{W})$ ;  
11      Compute  $\boldsymbol{\mu}'_i = \mathbb{E}[\mathbf{h}|\mathbf{v}'_i, \mathbf{W}]$ ;  
12      Compute gradient  $\nabla_{\mathbf{W}} = (\mathbf{v}_i)(\boldsymbol{\mu}_i)^T - (\mathbf{v}'_i)(\boldsymbol{\mu}'_i)^T$  ;  
13      Accumulate  $\mathbf{g} := \mathbf{g} + \nabla_{\mathbf{W}}$ ;  
14   Update parameters  $\mathbf{W} := \mathbf{W} + (\alpha_t/B)\mathbf{g}$ 
```

---

# Имплементација

```
void rbm::fit(const arma::mat& data, double lr, unsigned int epochs_count)
{
    arma::mat biased_data = arma::join_horiz(arma::ones(data.n_rows), data);
    auto act = sigmoid();
    std::random_device rd;
    std::mt19937 rng(rd());
    std::uniform_real_distribution<> uniform(0, 1);
    for(unsigned int epoch = 0; epoch < epochs_count; epoch++) {
        // positive phase
        arma::mat positive_logits = biased_data*m_weights;
        arma::mat positive_activations = act.evaluate(positive_logits);
        positive_activations.col(0) = arma::ones<arma::mat>(positive_activations.n_rows, 1);
        arma::mat sampled_latent = where(
            positive_activations,
            [&uniform=uniform, &rng=rng](double d) { return d > uniform(rng); },
            [](bool b, double d) { return b ? 1 : 0; }
        );
        arma::mat positive_association = biased_data.t() * positive_activations;

        // negative phase
        arma::mat negative_logits = sampled_latent*m_weights.t();
        arma::mat negative_activations = act.evaluate(negative_logits);
        negative_activations.col(0) = arma::ones<arma::mat>(negative_activations.n_rows, 1);

        arma::mat negative_latent_logits = negative_activations * m_weights;
        arma::mat negative_latent_activations = act.evaluate(negative_latent_logits);
        arma::mat negative_association = negative_activations.t() * negative_latent_activations;

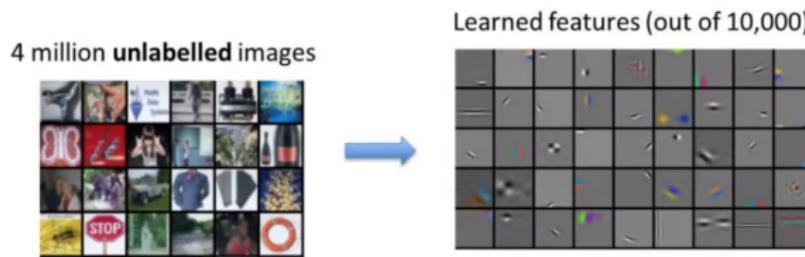
        m_weights += lr * ((positive_association - negative_association)/data.n_rows);

        std::cout << "epoch: " << epoch << " error: " << arma::accu(arma::pow((biased_data - negative_activations), 2))
    }
}
```

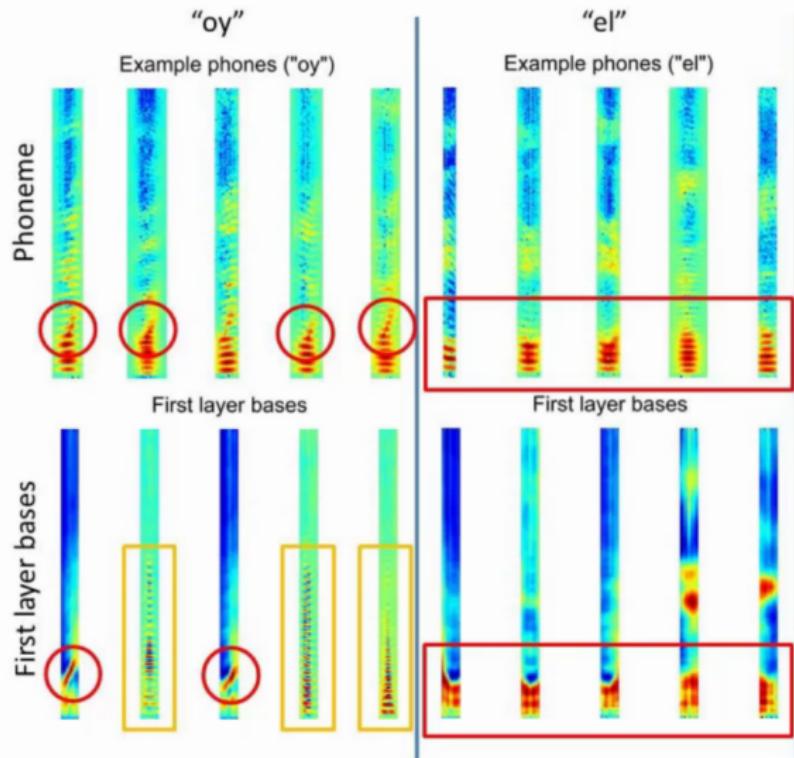
## Primene

- Мреже дубоког уверења
- Налажење угњеждених репрезентација
- Системи за препроучивање

## Визуелизација - слике



# Визуелизација - звук



## Референце

- A practical guide to training RBM, G. Hinton,  
<http://www.cs.toronto.edu/>
- Machine Learning - A probabilistic perspective, K. Murphy

# Хвала на пажњи