RBM Classifier for MNIST

ISPR - Midterm 2
Assignment 3

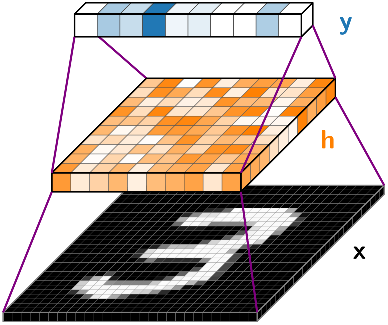
Filippo Baroni

May 3, 2021

RBM Classifier[1]

Model

- Visible units $\mathbf{x}$ (input).
  Size: $28 \times 28 \implies 784$.
- Half-visible units $\mathbf{y}$ (output).
  Size: 10.
- Hidden units $\mathbf{h}$.
  Size: 100.

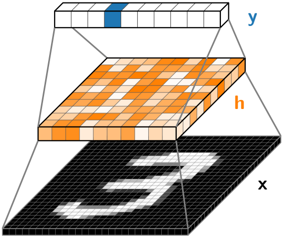


Energy

$$E(\mathbf{x}, \mathbf{y}, \mathbf{h}) = -\mathbf{h}^t\mathbf{W}\mathbf{x} - \mathbf{h}^t\mathbf{U}\mathbf{y} - \\ - \mathbf{b}^t\mathbf{x} - \mathbf{c}^t\mathbf{h} - \mathbf{d}^t\mathbf{y}$$

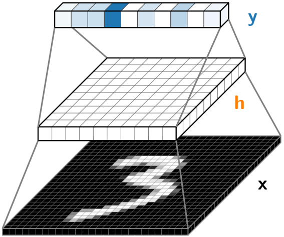
[1] Larochelle, Hugo and Bengio, Yoshua, "Classification using Discriminative Restricted Boltzmann Machines". In: *Proceedings of the 25th International Conference on Machine Learning*. ICML '08.

Training and Prediction



Training

- Treat **y** as visible units.
- Use one-hot encoding for **y**.
- Contrastive divergence, 1 step.
- SGD with momentum.



Prediction

- Treat **y** as hidden units.
- Compute the exact conditional distribution

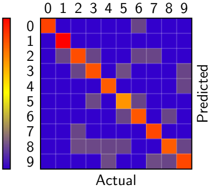
$$p(y|\mathbf{x}) \propto e^{d_y} \prod_{j=0}^{9} \left(1 + e^{c_j + U_{jy} + \sum_i W_{ji} x_i}\right).$$

- Pick the digit y^* that maximizes $p(y^*|\mathbf{x})$.

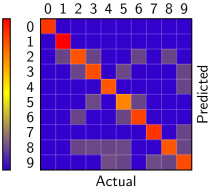
Results

RBM Classifier

Test
accuracy:
92.9%

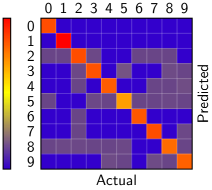


Train
accuracy:
92.7%

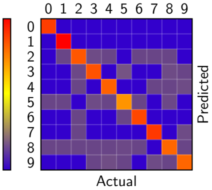


RBM + `sklearn.linear_model`

Test
accuracy:
90.4%



Train
accuracy:
89.6%



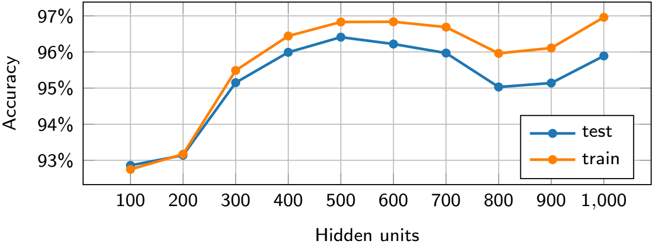
Neuron activations

Experiments

- **Gibbs sampling**

Input	Outputs (20 iterations)

- **More hidden units**



Appendix - Code

One epoch of training

```
# [...] initialize x0 and y0
h0P = sigmoid(self.c + self.W @ x0 + self.U @ y0)
wakeW = h0P @ x0.T
wakeU = h0P @ y0.T
h0 = sample(h0P)
x1 = sample(sigmoid(self.b + self.W.T @ h0))
y1 = sample(sigmoid(self.d + self.U.T @ h0))
h1P = sigmoid(self.c + self.W @ x1 + self.U @ y1)
dreamW = h1P @ x1.T
dreamU = h1P @ y1.T
# [...] perform gradient descent
```

Prediction

```
x = data.T
t = (self.c + self.U)[:, :, None] + (self.W @ x)[:, None, :]
P = self.d + np.sum(np.log1p(np.exp(t)), axis = 0)
return np.argmax(P, axis = 0)
```