

Modèle Graphique : Machine de Boltzmann Restreinte

Génération et Classification

Leyth Akrouit

Université d'Evry

6 juin 2022

Outline

1 Présentation du Modèle

- Champ de Markov aléatoire
- Densité
- Probabilités

2 Apprentissage

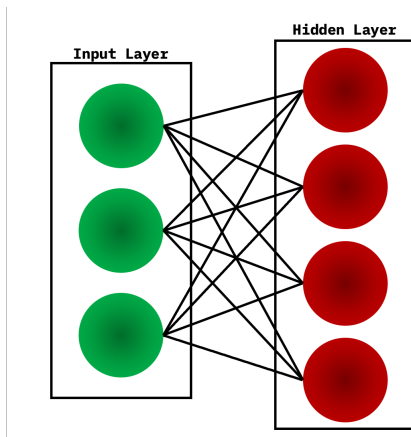
- Vraisemblance du Modèle
- Contrastive Algorithm
- Classification

3 Résultats

- Reconstructions
- Classification

Présentation du modèle

Champ de Markov aléatoire



p variables visibles indépendentes

q variables cachées indépendentes

Présentation de la Méthode

Densité

Théorème d'Hammersley-Clifford :

$$p(v, h) = \frac{1}{Z} \prod_{c \in \text{Cliques}} \psi_c(v_c * h_c)$$

$$\psi_c(v_c * h_c) = e^{-E(v_c, h_c)}$$

$$p(v, h) = \frac{1}{Z} \exp\left(\sum_{i \in \text{visible}} a_i v_i + \sum_{j \in \text{cache}} b_j h_j + \sum_{i \in \text{visible}} \sum_{j \in \text{cache}} v_i w_{ij} h_j \right)$$

Probabilités

Probabilités Conditionnelles

Grâce au modèle considéré on peut facilement calculer :

$$p(h_j = 1|v) = p(h_j = 1|v, h \setminus \{h_j\}) = \text{sigm}(b_j + \sum_i v_i w_{ij})$$

$$p(v_i = 1|h) = p(v_i = 1|h, v \setminus \{v_i\}) = \text{sigm}(a_i + \sum_j w_{ij} h_j)$$

Ces deux probabilité conditionnelles sont à la base de l'algorithme principal et de la génération de données

En prenant le logarithme de la densité et en dérivant selon un paramètre θ :

$$\frac{\partial \log(p(a, b, w))}{\partial \theta} = -\mathbb{E}_{h|v_i} \left(\frac{\partial}{\partial \theta} E(v_i, h) \right) + \mathbb{E}_{v, h} \left(\frac{\partial}{\partial \theta} E(v, h) \right)$$

La deuxième partie dépend de Z et est donc difficilement calculable

L'idée est de remplacer les espérances en faisant de l'échantillonnage de Gibbs :

- On fixe K le nombre d'étapes de l'échantillonnage de Gibbs.
- On choisit au hasard un élément $v^0 \in V$
- On génère h^0 à l'aide de $p(h|v)$
- On génère v^1 à l'aide de $p(v|h^0)$
- On génère h^1 à l'aide de $p(h|v^1)$
- On réitère les étapes 4 et 5 k fois

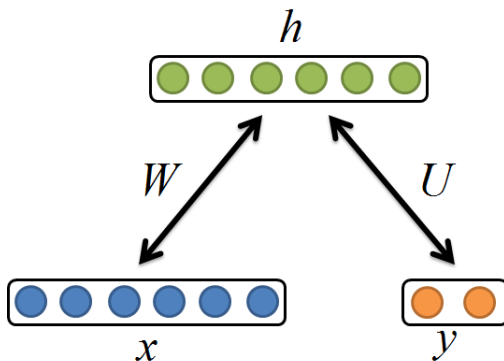
Enfin on actualise les paramètres :

$$\theta \leftarrow \theta + \lambda \left(\frac{\partial}{\partial \theta} E(v^0, h^0) - \left(\frac{\partial}{\partial \theta} E(v^k, h^k) \right) \right)$$

Classification

Classification

Pour faire de la classification il suffit de modifier légèrement le modèle en introduisant la variable de label y :



Classification

Classification

On obtient donc la fonction d'énergie suivante :

$$\begin{aligned} \sum_{i \in \text{visible}} a_i v_i + \sum_{j \in \text{hidden}} b_j h_j + \sum_{m \in \text{label}} c_m y_m + \sum_{i \in \text{visible}, j \in \text{hidden}} v_i w_{ij} h_j \\ + \sum_{j \in \text{hidden}, m \in \text{label}} y_j u_{jm} h_j \end{aligned}$$

On peut montrer que :

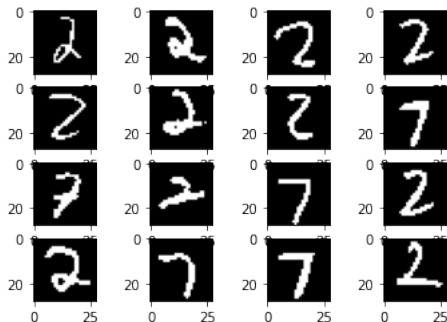
$$p(y|x) = \frac{e^{c_y} \prod_{j=1}^n (1 + e^{b_j + U_{jy} + \sum_i w_{ji} v_i})}{\sum_{y^*} e^{c_{y^*}} \prod_{j=1}^n (1 + e^{b_j + U_{jy^*} + \sum_i w_{ji} v_i})}$$

Résultats

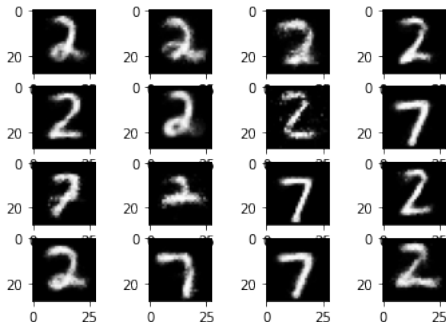
Reconstructions avec 2 labels

Images MNIST : -784 pixels (variables visibles)
-100 Variables cachées

Images Originales

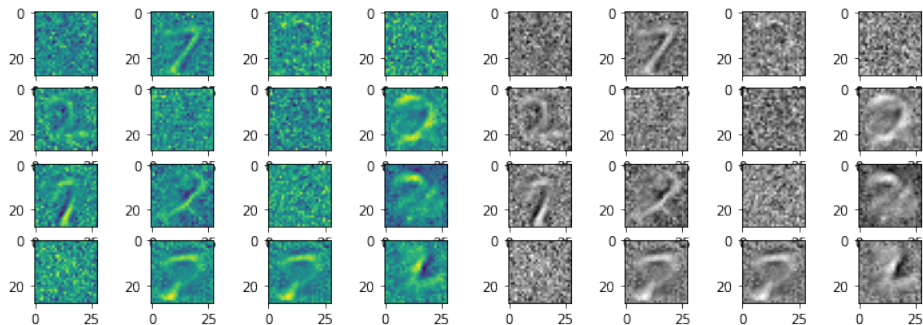


Images Reconstruites



Résultats

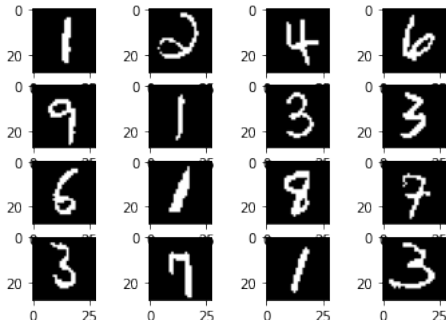
Poids vecteurs cachées



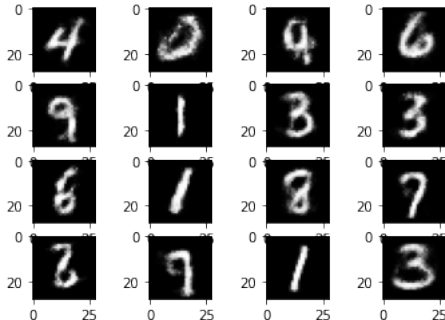
Résultats

Reconstructions avec 10 labels

Images Originales



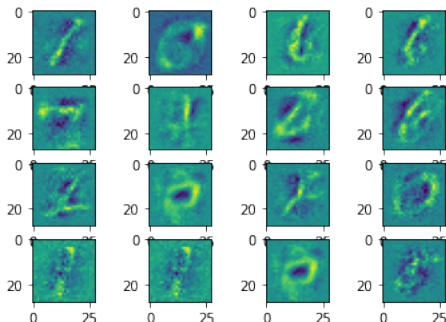
Images Reconstruites



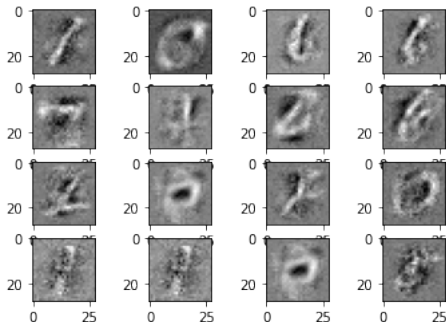
Résultats

Reconstructions avec 2 labels

Images Originales



Images Reconstruites



Classification

Classification 2 labels

- Accuracy train : 98.2%
- Accuracy validation : 97.2%

Classification

Classification 10 labels

- Accuracy train : 85.2%
- Accuracy validation : 85.4%