# Modèle Graphique : Machine de Boltzmann Restreinte

## Génération et Classification

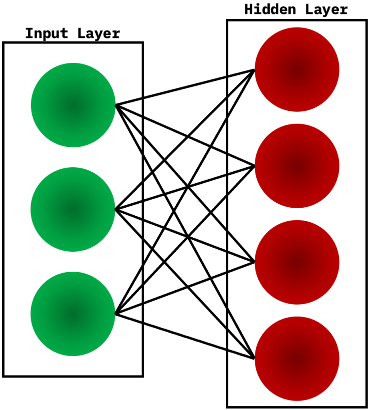
Leyth Akrout

Printemps 2021

# Outline

# Présentation du modèle

## Champ de Markov aléatoire




**p** variables visibles indépendantes **q** variables cachées indépendantes

# Présentation de la Méthode

## Densité

Théorème d'Hammersley-Clifford :

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

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

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

Grâce au modèle considéré on peut facilement calculer :

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

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

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

# Apprentissage

## Vraisemblance du Modèle

En prenant le logarithme de la densité et en dérivant selon un paramètre $\theta$ :

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

La deuxième partie dépend de Z et est donc difficilement calculable

# Apprentissage
## Contrastive Algorithm

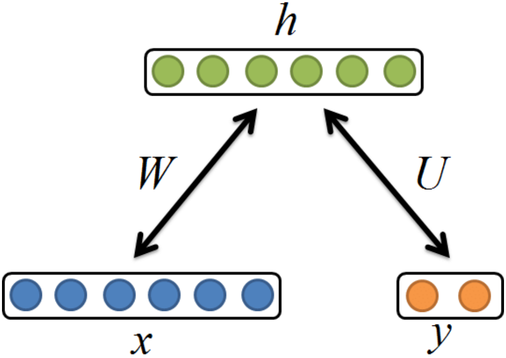
L'idée est de remplacer les espérences en faisant de l'échantillonage de Gibbs :

- On fixe K le nombres d' ´etapes de l' ´echantillonage 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 :

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

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 :

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

On peux montrer que :

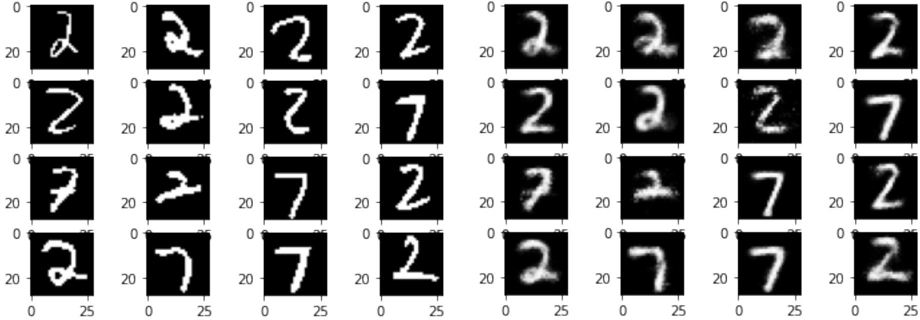
$$p(y|x) = \frac{e^{c_y} \prod_{j=1}^{n} (1 + e^{b_j + U_{jy} + \sum_i W_{ji} v_i})}{\sum_{y^*} e^{c_y^*} \prod_{j=1}^{n} (1 + e^{b_j + U_{jy^*} + \sum_i W_{ji} v_i})}$$

# Résultats

## Reconstructions avec 2 labels

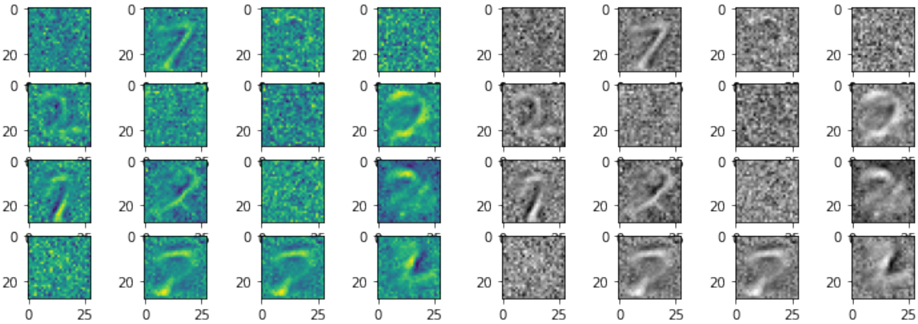
**Images MNIST** : -784 pixels ( variables visibles )
-100 Variables cachées

Images Originales — Images Reconstruites



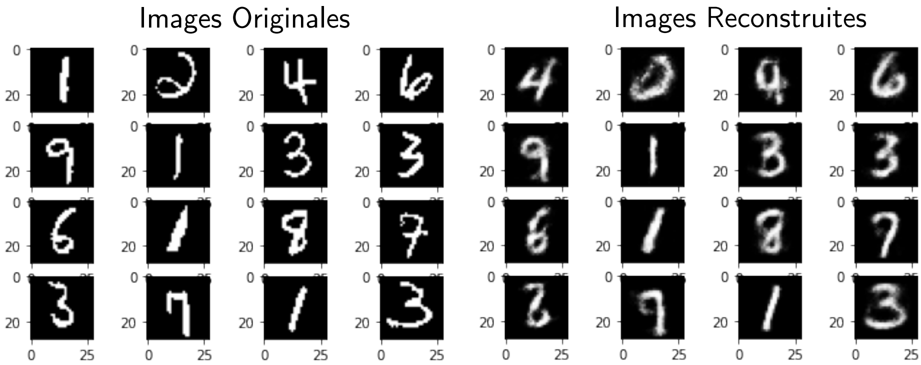
# Résultats

## Poids vecteurs cachées



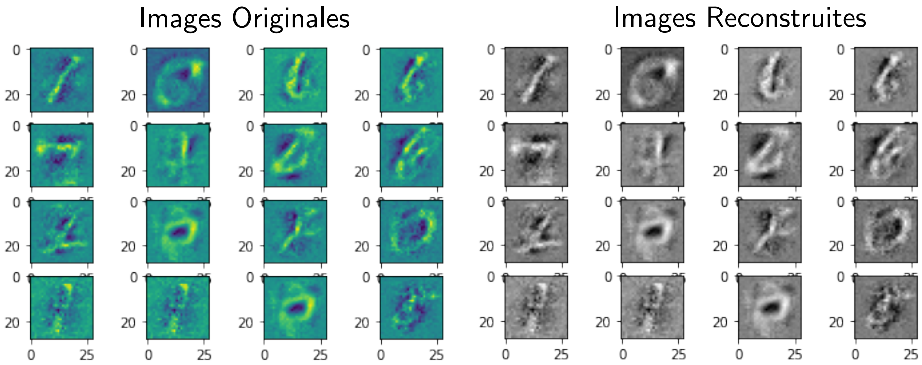
# Résultats

## Reconstructions avec 10 labels



# Résultats

## Reconstructions avec 2 labels



# Classification

## Classification 2 labels

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

# Classification

## Classification 10 labels

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