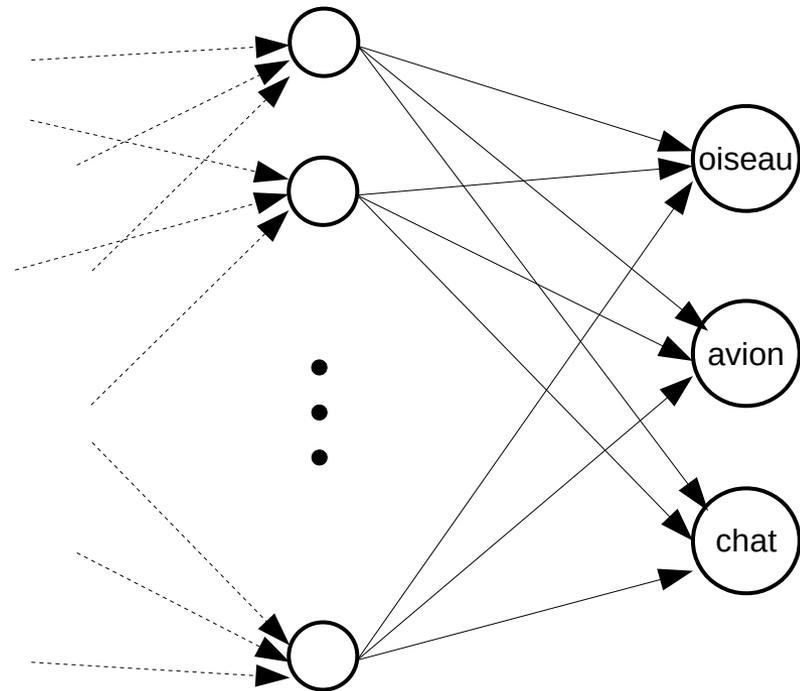


INTRODUCTION AUX RÉSEAUX DE NEURONES

Réseau multiclasse

Pascal Germain, 2019

Réseau avec sortie multiclasse

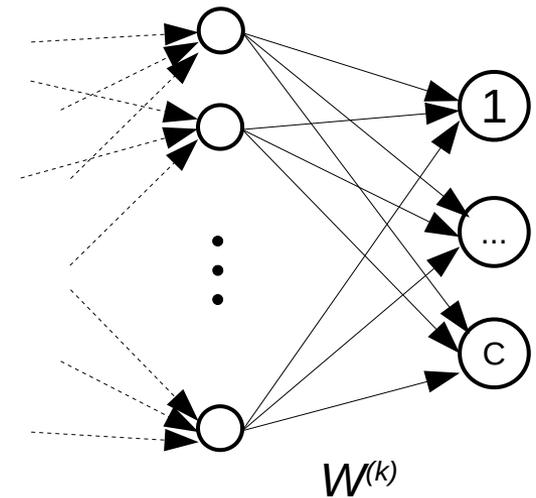


Réseau avec sortie multiclasse

Soit un problème à C classes.

1. Pour un exemple d'apprentissage (x, y) , on représente chaque classe par un entier

$$y \in \{1, \dots, C\}$$



Réseau avec sortie multiclasse

Soit un problème à C classes.

1. Pour un exemple d'apprentissage (x, y) , on représente chaque classe par un entier

$$y \in \{1, \dots, C\}$$

2. On converti y sous la forme d'un vecteur one-hot $\mathbf{y} \in \mathbb{R}^C$, possédant la valeur 1 à l'index correspondant à y , et les valeurs 0 autrement:

$$y = 1 \mapsto \mathbf{y} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad y = 2 \mapsto \mathbf{y} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

3. La couche de sortie sera donnée un vecteur $\hat{\mathbf{y}}$ obtenu en appliquant la fonction d'activation softmax aux valeurs $\mathbf{a} \in \mathbb{R}^C$ propagées par les couches précédentes:

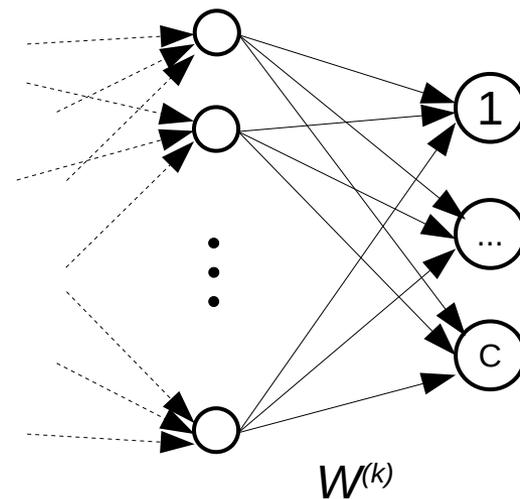
$$\hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_C \end{bmatrix} \quad \text{avec } \hat{y}_i = \text{softmax}(a_i) = \frac{e^{a_i}}{\sum_{j=1}^C e^{a_j}}$$

On interprète le vecteur \mathbf{y} comme une distribution de probabilité sur les classes.

4. Lors de l'optimisation du réseau, on minimise la perte du négatif log-vraisemblance:

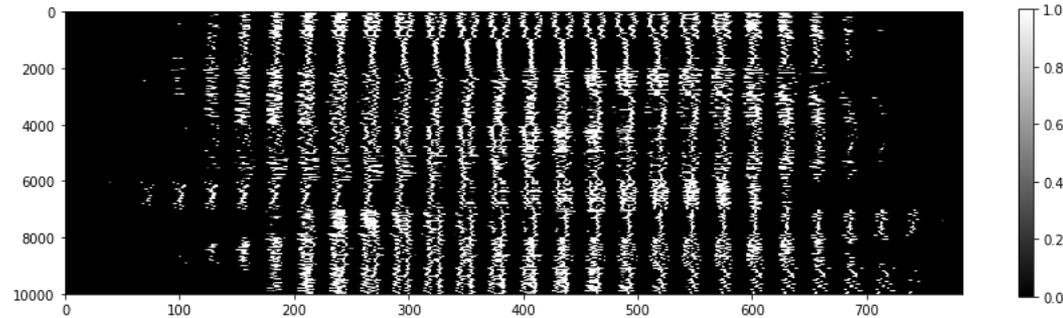
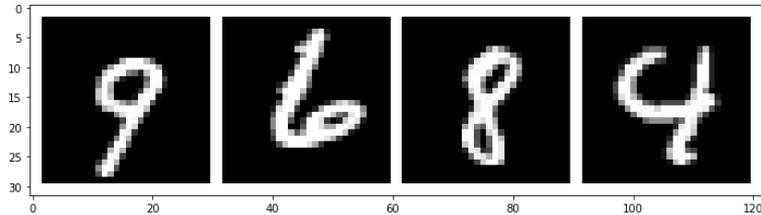
$$L_{\text{nlv}}(\hat{\mathbf{y}}, y) = -\ln(\hat{y}_y),$$

où \hat{y}_y correspond à la sortie d'index y du réseau (la probabilité associée à la classe de l'exemple).



Exemple : l'ensemble MNIST

Entrée : Image de 28 x 28 pixels sous la forme d'un vecteur de 784 dimensions

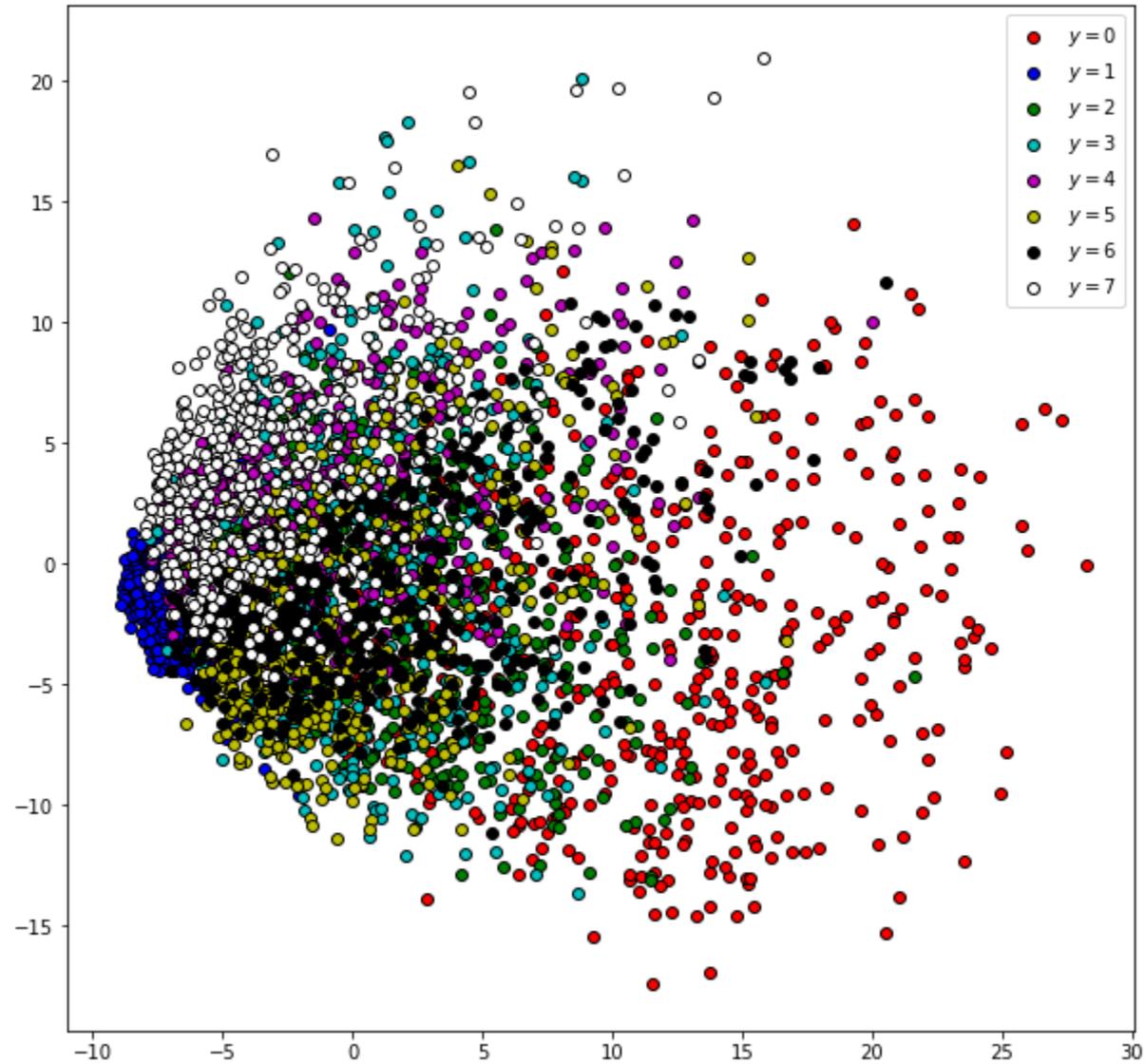


Sortie: {0,1,2,3,4,5,6,7,8,9}

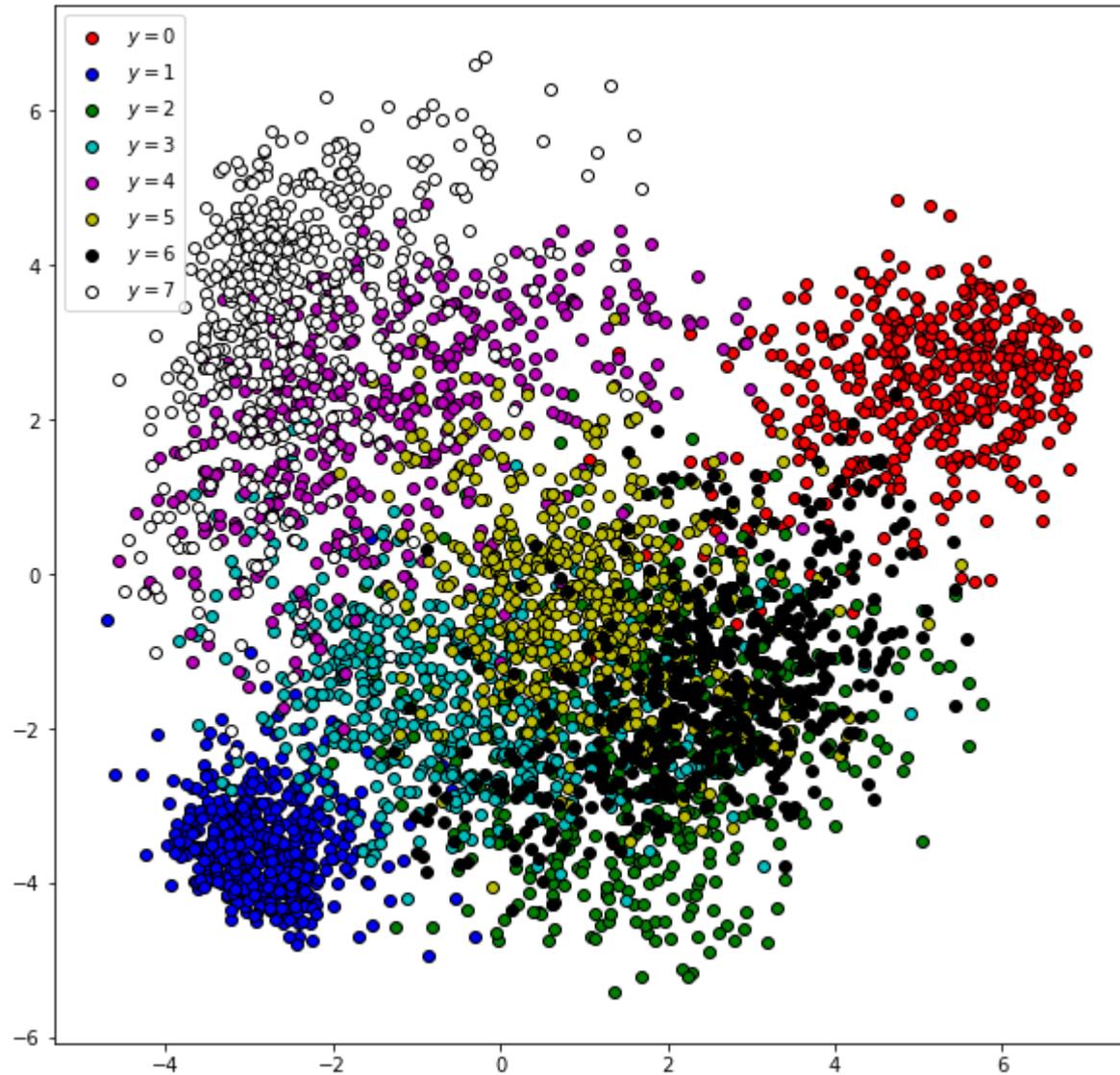
Architecture du réseau

Entrée : 784 neurones
Couche 1 : 50 neurones (tanh)
Couche 2 : 50 neurones (tanh)
Couche 3 : 50 neurones (tanh)
Sortie : 10 neurones (soft max)

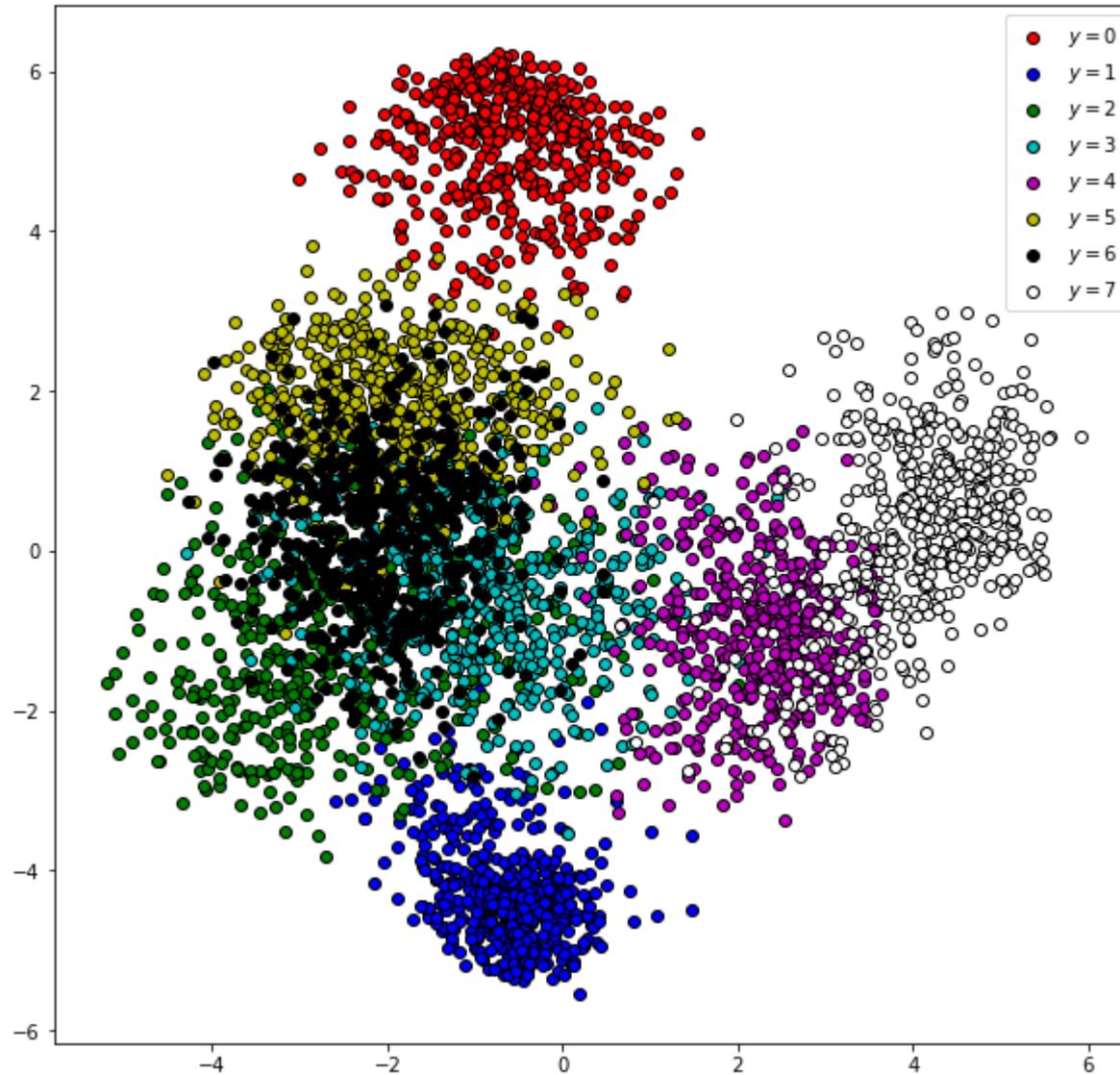
PCA sur les données brutes



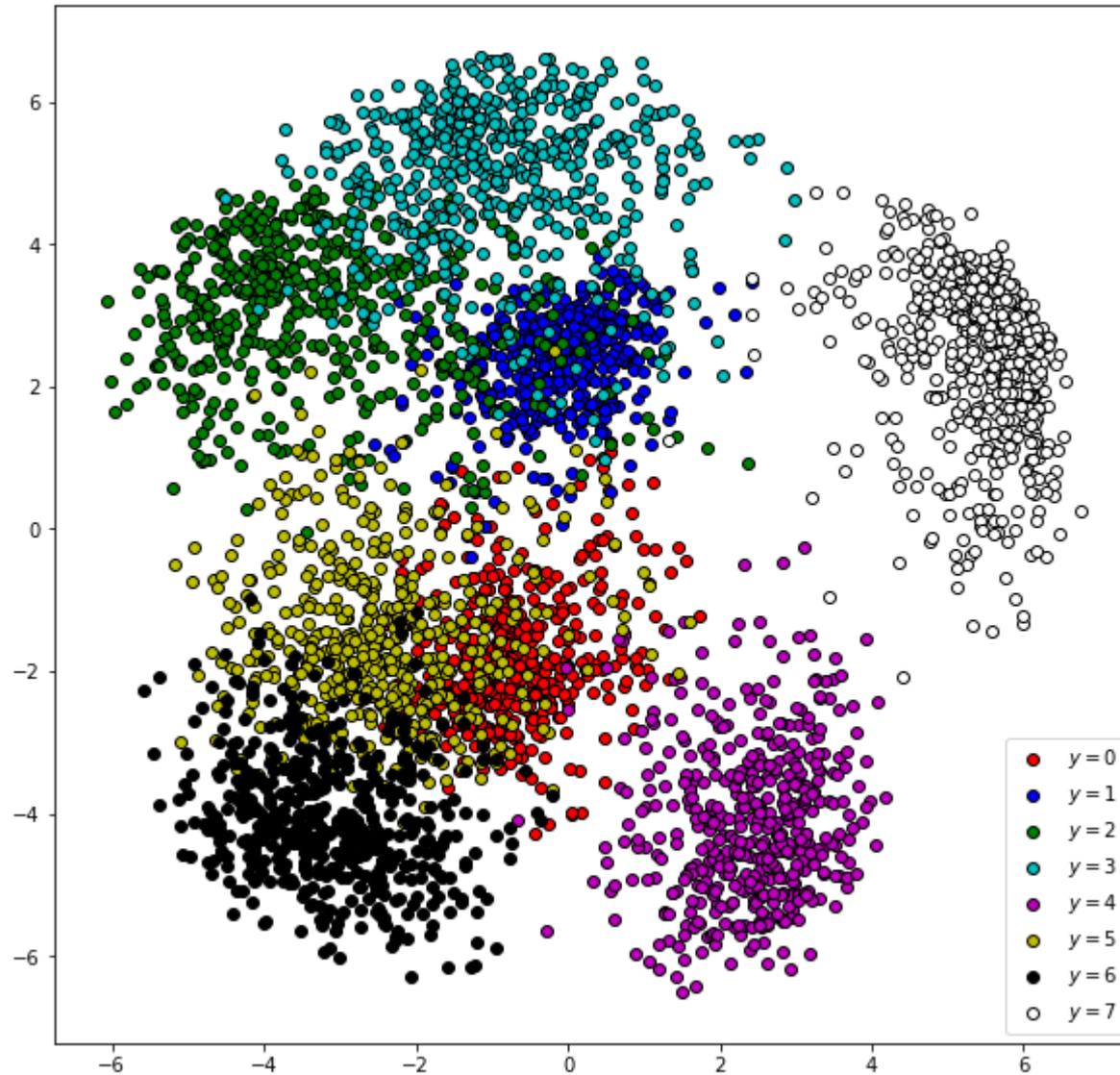
PCA sur les représentations (1^{ère} couche cachée)



PCA sur les représentations (2^e couche cachée)

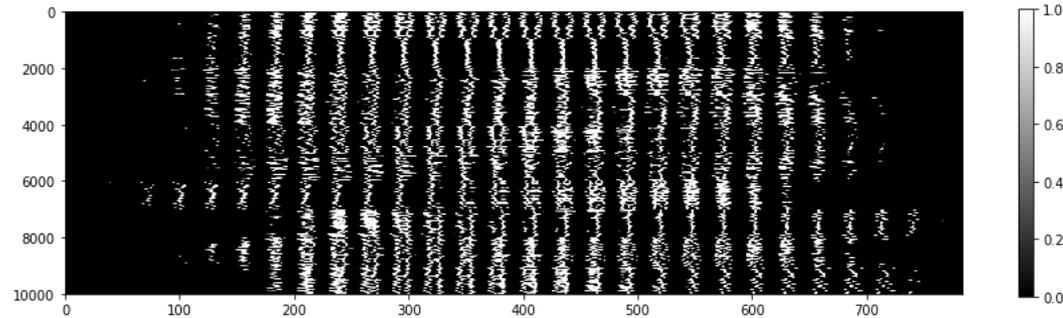
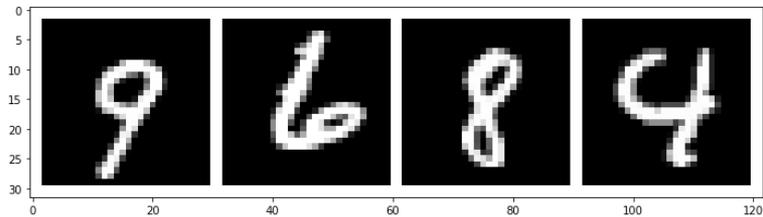


PCA sur les représentations (3^e couche cachée)



Exemple : l'ensemble MNIST

Entrée : Image de 28 x 28 pixels sous la forme d'un vecteur de 784 dimensions

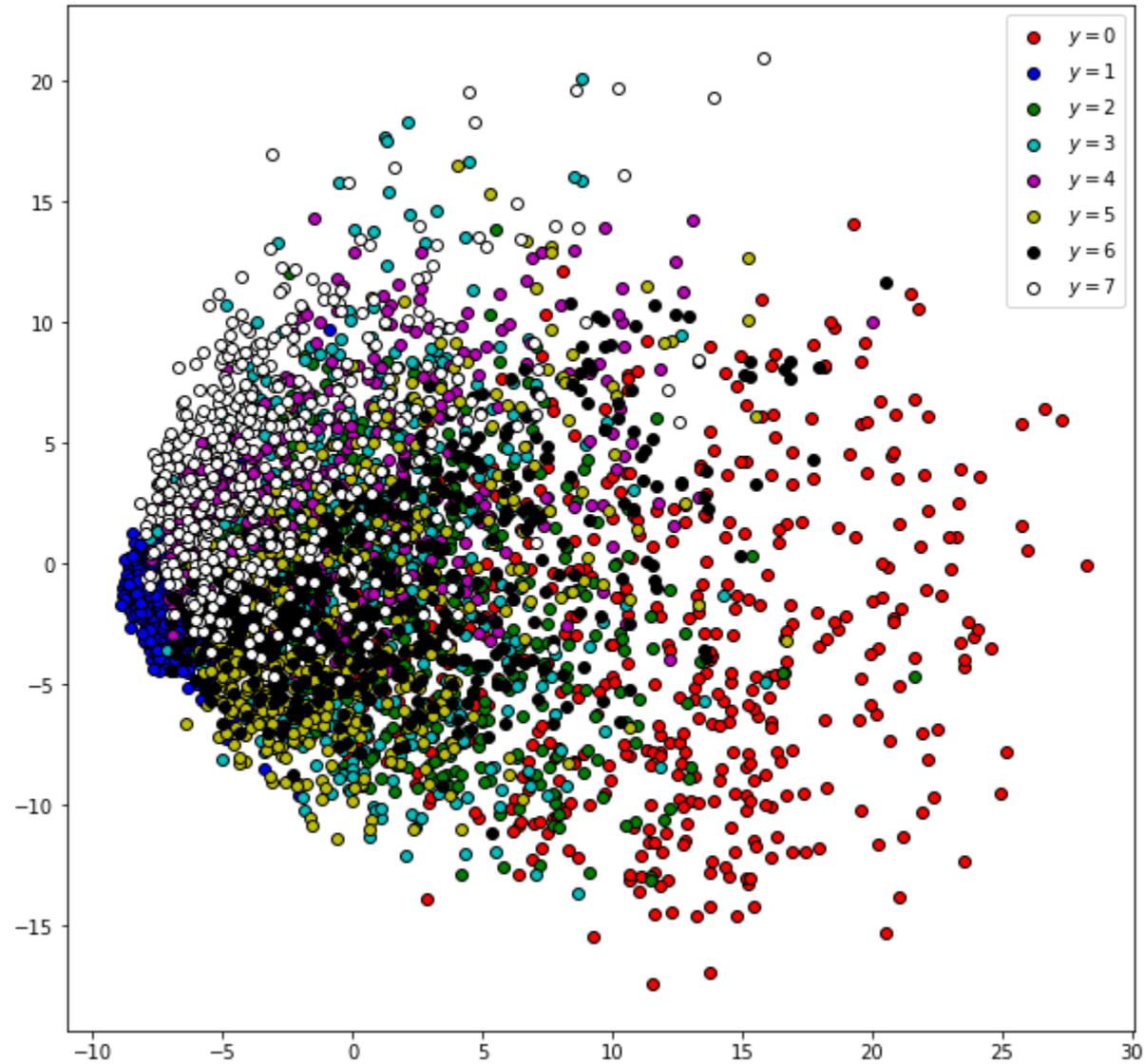


Sortie: {0,1,2,3,4,5,6,7,8,9}

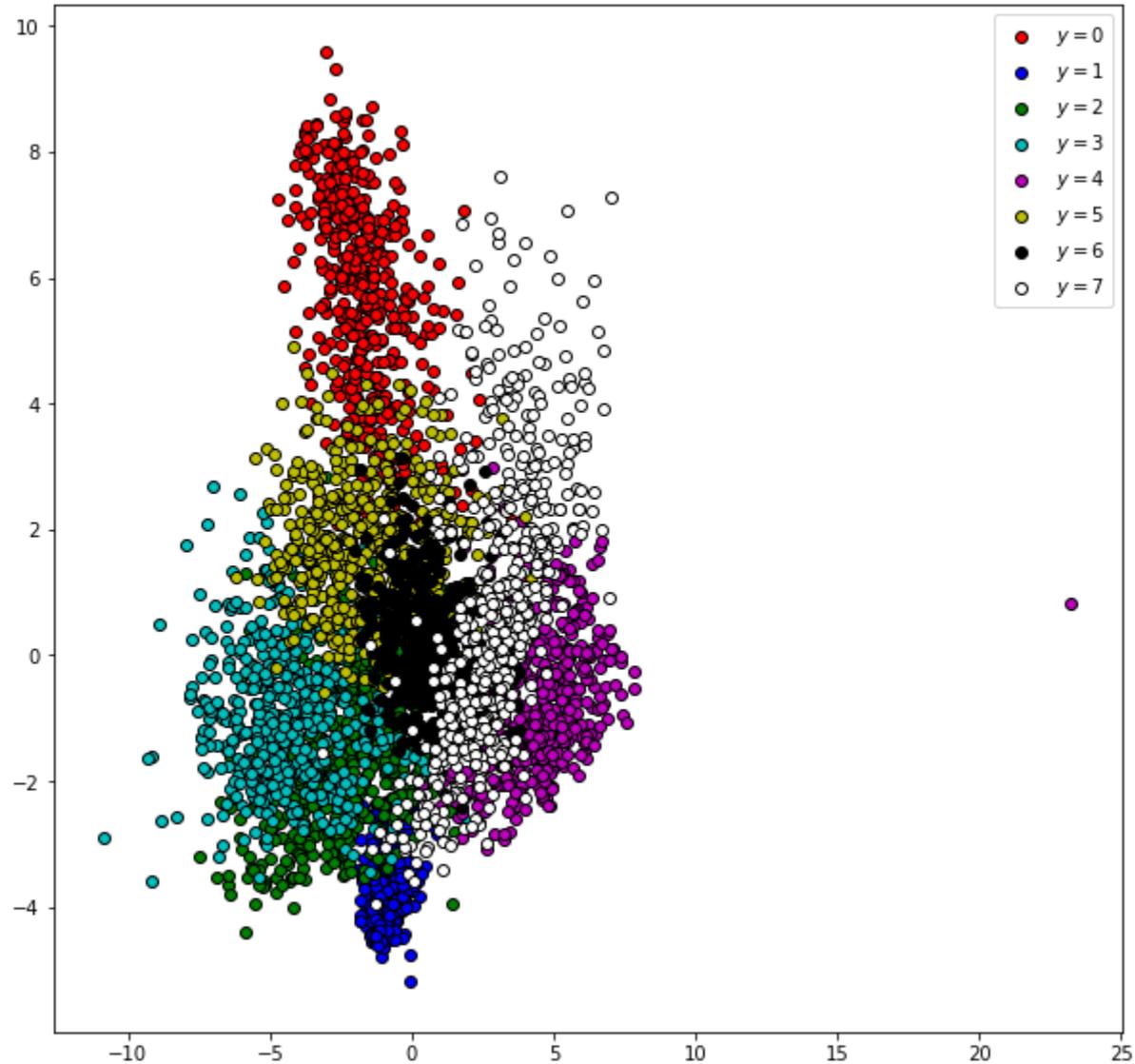
Architecture du réseau

Entrée : 784 neurones
Couche 1 : 50 neurones (relu)
Couche 2 : 50 neurones (relu)
Couche 3 : 50 neurones (relu)
Sortie : 10 neurones (soft max)

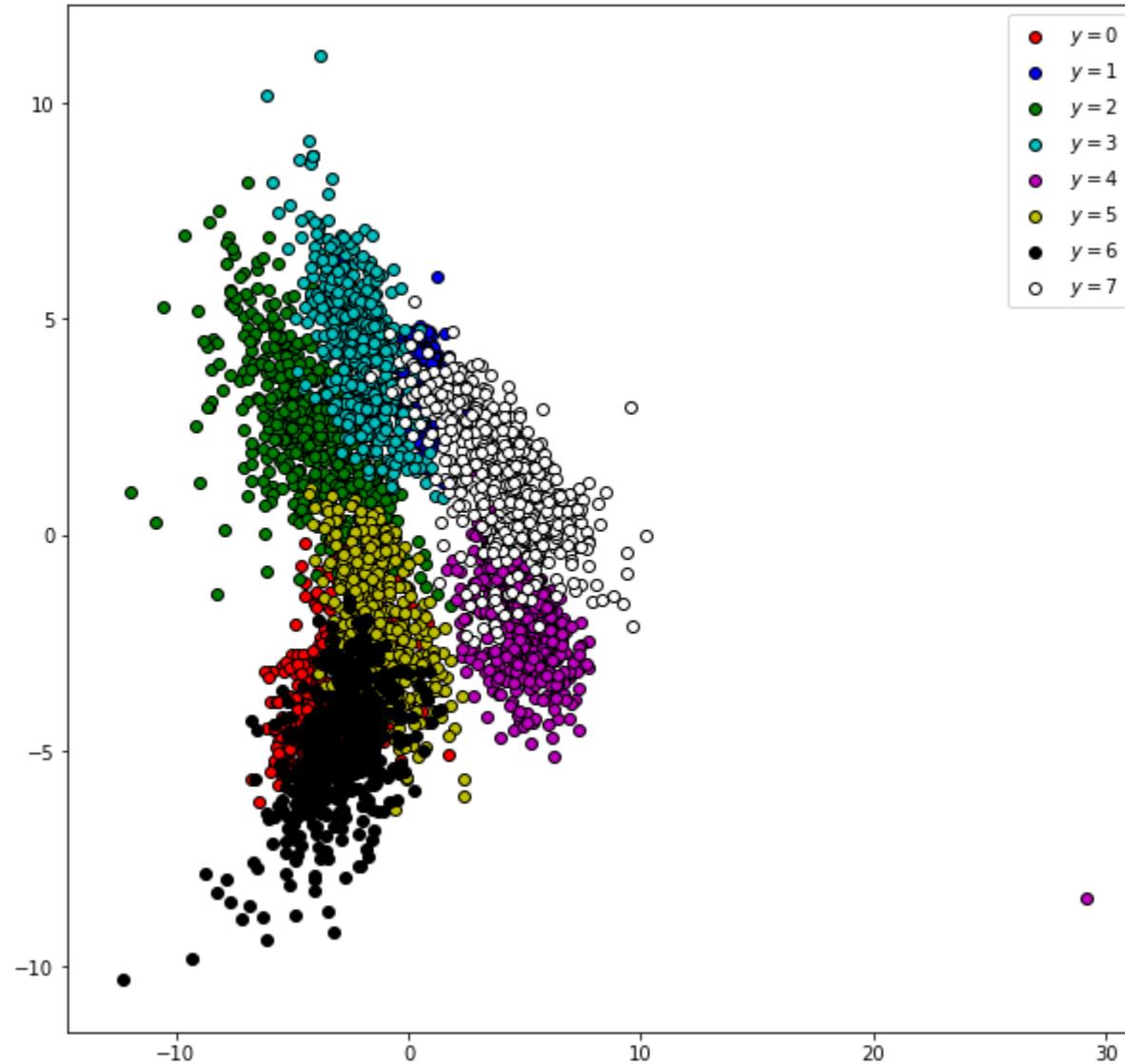
PCA sur les données brutes



PCA sur les représentations (1^{ère} couche cachée)



PCA sur les représentations (2^e couche cachée)



PCA sur les représentations (3^e couche cachée)

