Autoencodeurs : comparaison avec d'autres méthodes de réduction de la dimension

Damien Babet Julie Djiriguian

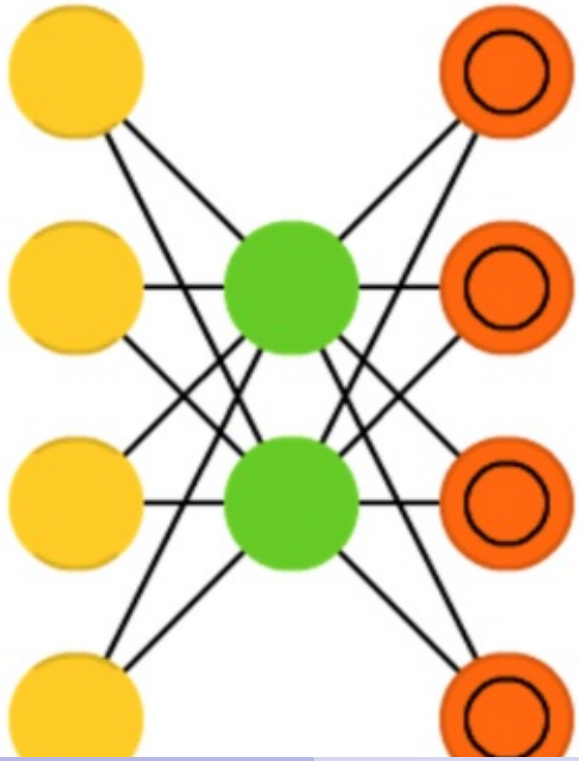
Projet de techniques avancées d'apprentissage

Plan

Définition d'un auto-encodeur

Auto-encodeur = réseaux de neurones constitués de deux phases :

- Une première phase de compression de l'information
- Une seconde phase de reconstitution de l'information initiale



Les AE : différents types et différents usages

- Différentes régularisations (faible nombre de cellule, sparsité, etc.)
- Input bruité : entraîné pour le débruitage des données
- Unité probabilistes : approximation de la distribution des inputs
- Stacked AE : initialisation pour les RN profonds

Le contexte de la factorisation matricielle

Matrice des données : $\mathbf{X} \in \mathbb{R}^{n*p}$

Approximation de faible rang $k < min(n,p)$: $\mathbf{X}_k \approx \mathbf{X}$

$\mathbf{X}_k$ peut être factorisé : $\mathbf{X}_k = \mathbf{PQ}, \mathbf{P} \in \mathbb{R}^{n*k}, \mathbf{Q} \in \mathbb{R}^{k*p}$

de manière non unique.

Réciproquement :

$\mathbf{P} \in \mathbb{R}^{n*k}$ est un encodage linéaire des données s'il existe $\mathbf{Q} \in \mathbb{R}^{k*p}$ tel que $\mathbf{X}_k = \mathbf{PQ}$.

La PCA donne la meilleure approximation (pour la norme de Frobenius) dans le cas général. Mais il existe de très nombreuses variantes avec des contraintes supplémentaires (données et facteurs non-négatifs, données pondérées, etc.)

Enjeu de l'analyse

Auto-encodeurs et factorisation de matrice partagent une fonction importante : la réduction de dimension

Dans quelle mesure les auto-encodeurs peuvent-ils concurrencer la PCA ?

Plan

Un AE linéaire équivaut à une PCA

Soit un AE **linéaire**, avec **1 couche cachée** de k unités, sans terme de biais, avec des poids $\mathbf{W}$ avec en entrée $\mathbf{X}$ = n vecteurs de taille p :

- Unités de la couche chachée : $h = \mathbf{XW}$
- Unités en sortie de l'AE : $\widehat{\mathbf{X}_k}^{AE} = h\mathbf{W}^T$

D'où : $\widehat{\mathbf{X}_k}^{AE} = \mathbf{XWW}^T$

Problème d'optimisation :

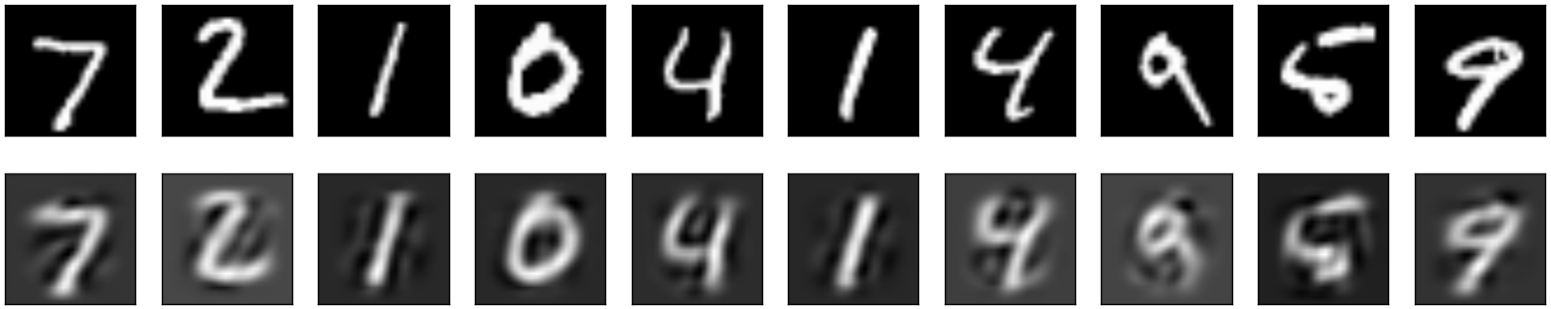
$$argmin_{w_{ij}}||\hat{X}^{AE} - X||^2 = argmin_{w_{ij}}||X(W'W - I)||^2$$

Comparaison AE linéaire et PCA

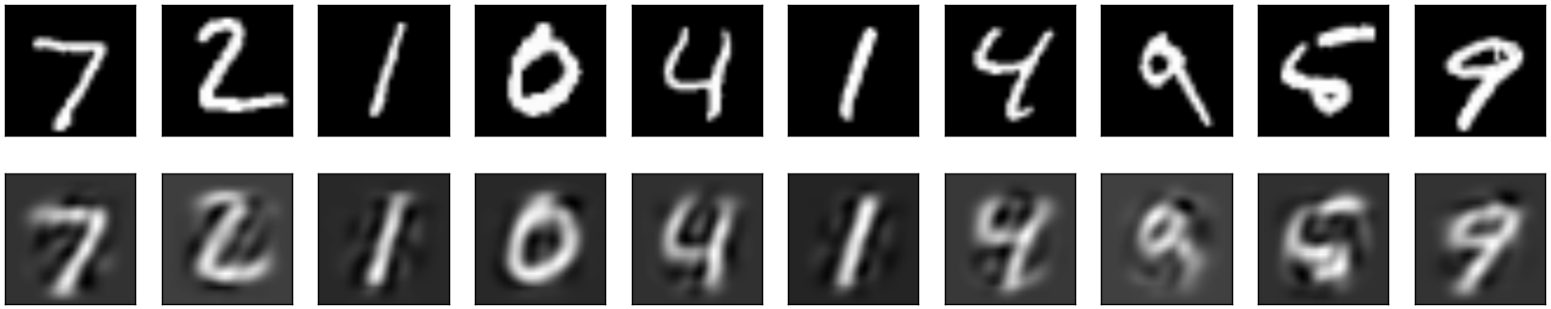
	AE linéaire	**PCA**
Optimisation	$argmin_{w_{ij}} \|\|X(W'W - I)\|\|^2$	$argmin_{w_{ij}} \|\|X(W'W - I)\|\|^2$
Calcul	par apprentissage	direct via SVD
Encodage	Base de l'EV de $\widehat{\mathbf{X}_k}^{AE}$	Base orthonormée de l'EV de $\widehat{\mathbf{X}_k}^{PCA}$

Convergence expérimentale entre AE linéaire et PCA

Reconstruction via AE linéaire :



Reconstruction via PCA :

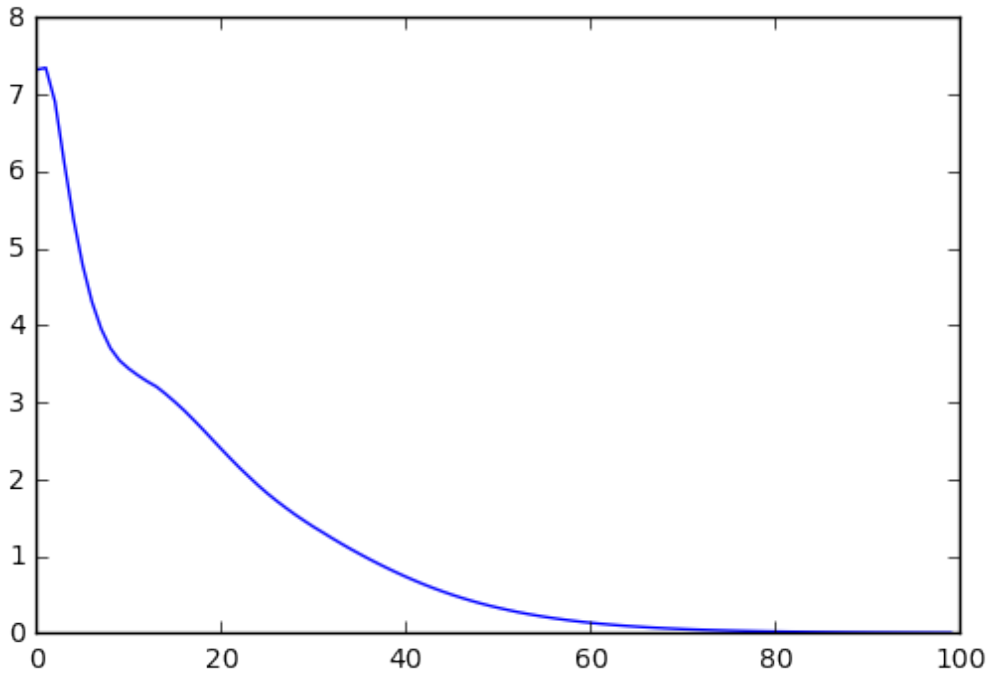


99% de labels prédits de façon identique avec ces deux méthodes

Convergence entre AE linéaire et PCA

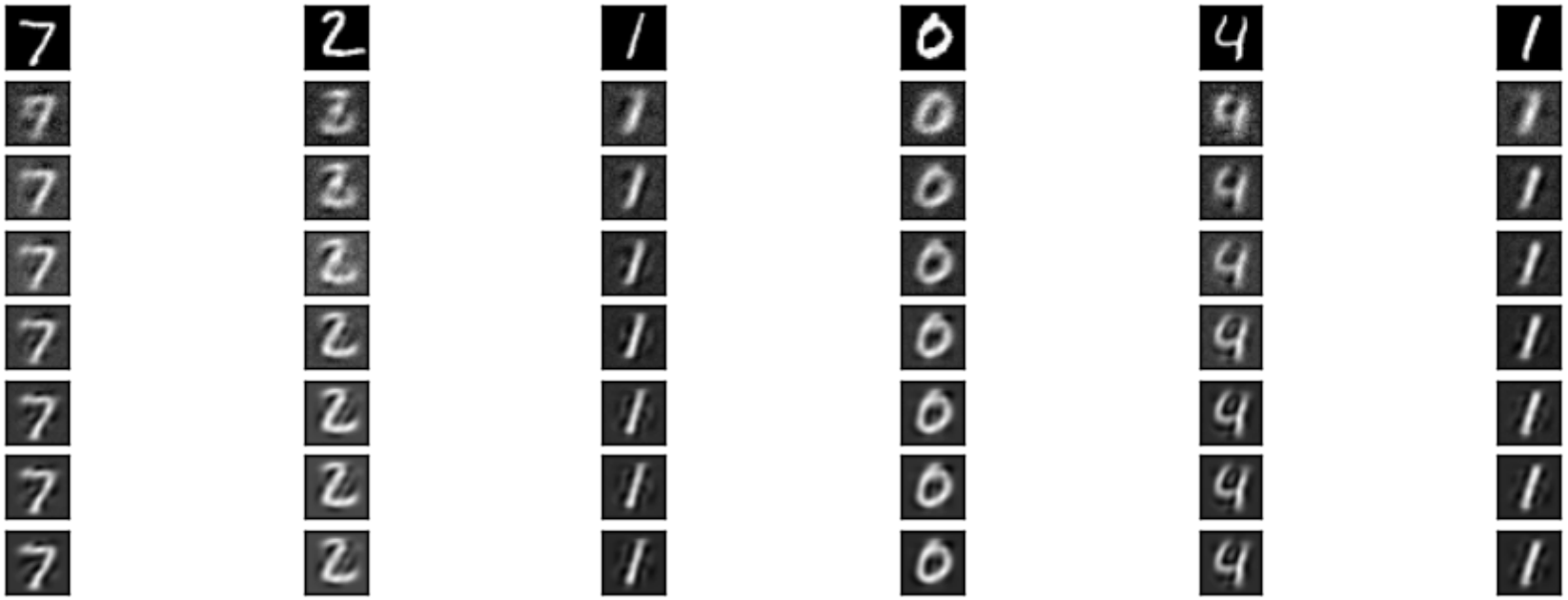
Différence (norme de Frobenius) entre $\widehat{\mathbf{X}}_k^{AE}$ et $\widehat{\mathbf{X}}_k^{PCA}$ en fonction du nombre d'époques de l'entraînement

Données : MNIST



Quelle amélioration avec un AE non linéaire ?

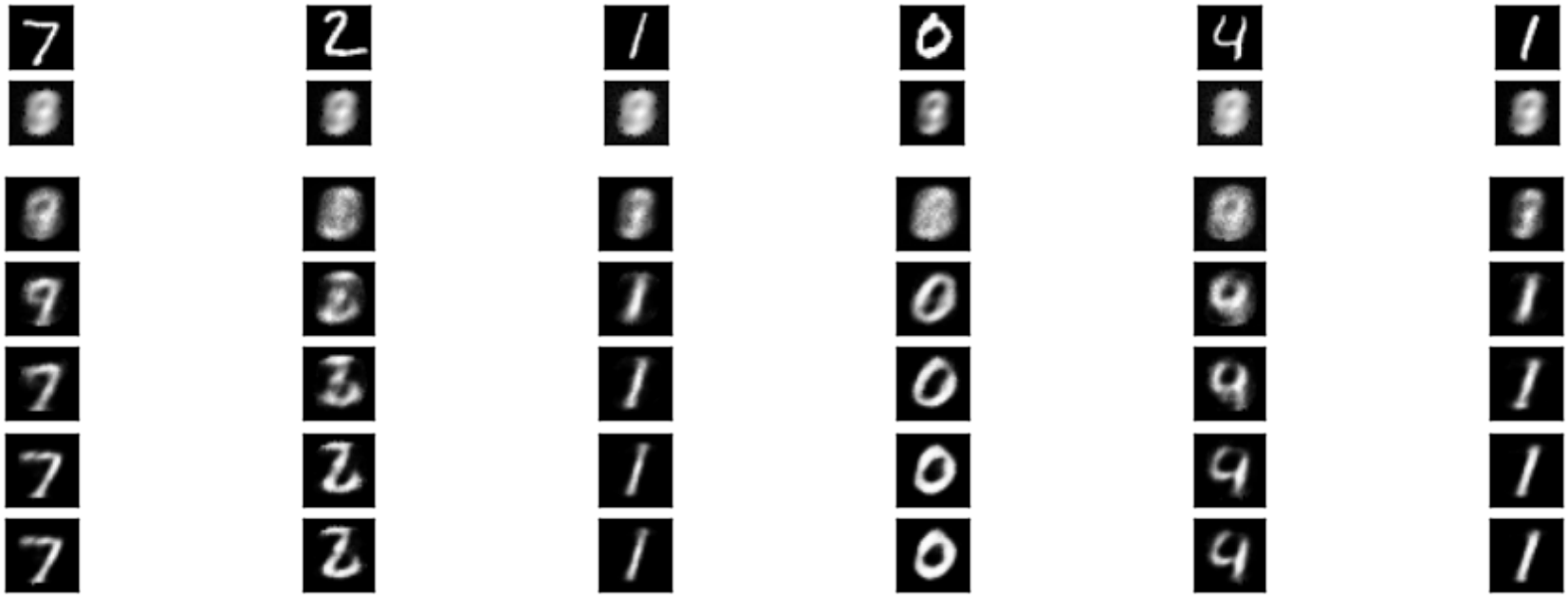
Progrès de la reconstruction avec l'AE linéaire (original, époques 1, 5, 10, 20, 40, 50 et 100)



Pourcentage d'erreur de classification avec AE : **9 %**

Quel amélioration avec un AE non linéaire ?

Progrès de la reconstruction avec l'AE non-linéaire (5 couches relu, sortie sigmoïde) (original, époques 1, 5, 10, 20, 40, et 50)



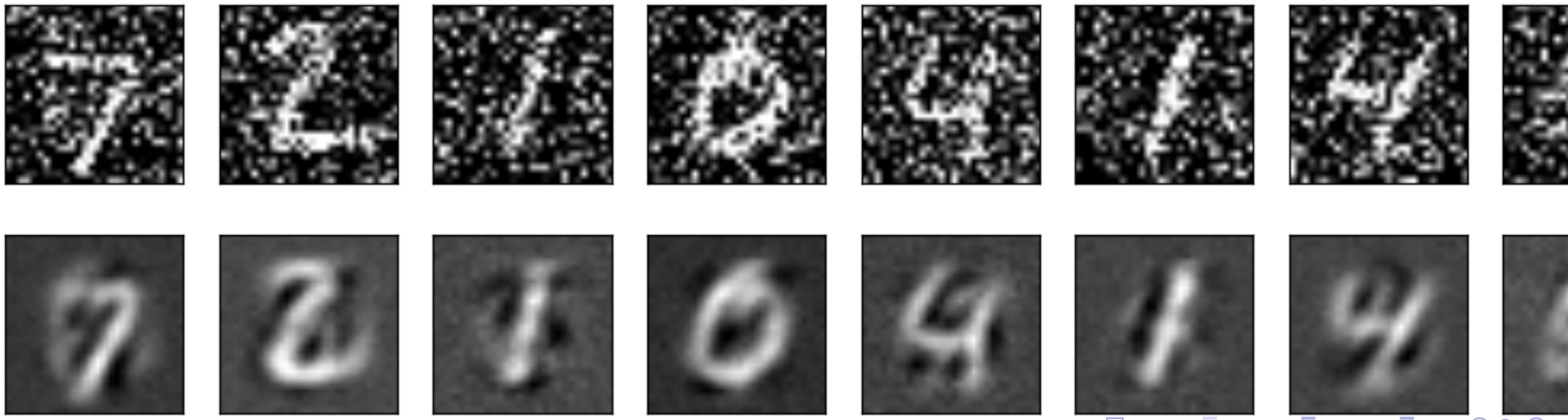
Pourcentage d'erreur de classification avec deep AE : **12 %**

Plan

Principe du débruitage

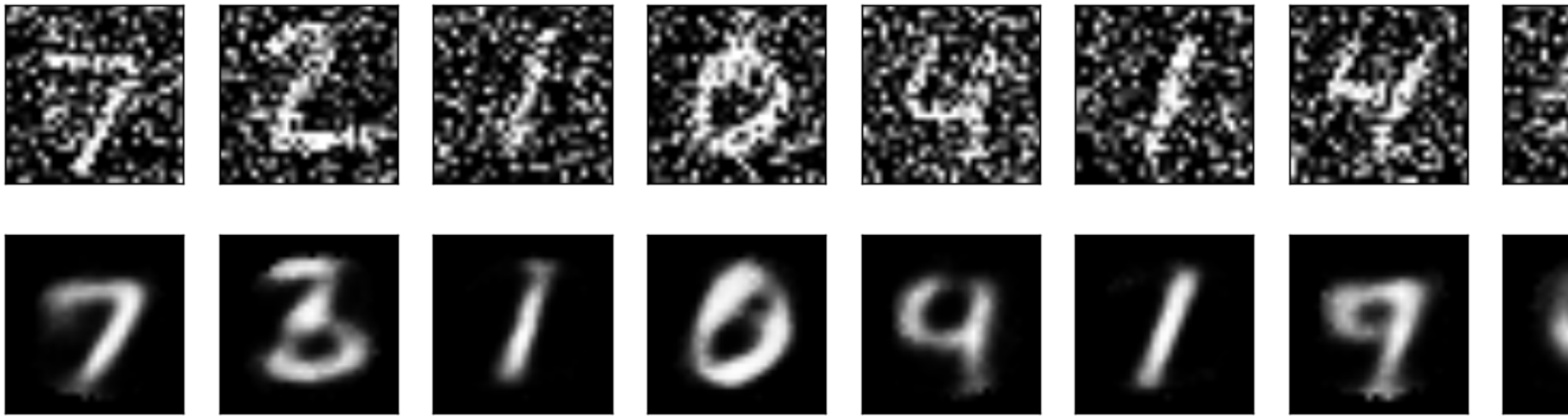
Un *denoising* AE reçoit en input une version artificiellement bruitée des données, et est entraîné à reproduire les données originales, non bruitées, en sortie. Il peut ensuite débruiter des données hors de cette échantillon d'entraînement.

Débruitage via un AE linéaire :



Un Deep AE est plus performant s'il est bien entraîné

Un AE non linéaire (5 couches) opère manifestement un débruitage très différent. Les gris dûs au caractère linéaire ont disparu. L'impression visuelle est bien meilleure. Mais certains chiffres sont transformés !



Qu'est-ce qui est gardé dans la couche compressée ?

Plan

De l'article à l'expérimentation

Usage d'un AE avec 2 unités d'encodage pour projeter les données sur un plan (Hinton et Salakhutdinov, Science, 2003) : supériorité visuelle du deep AE sur la PCA

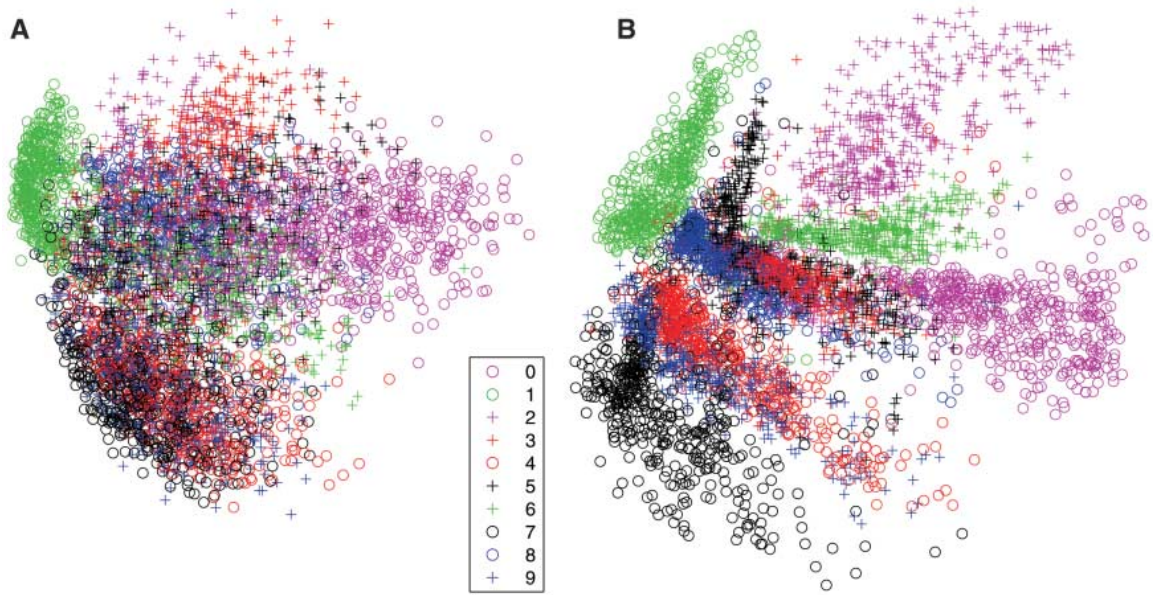
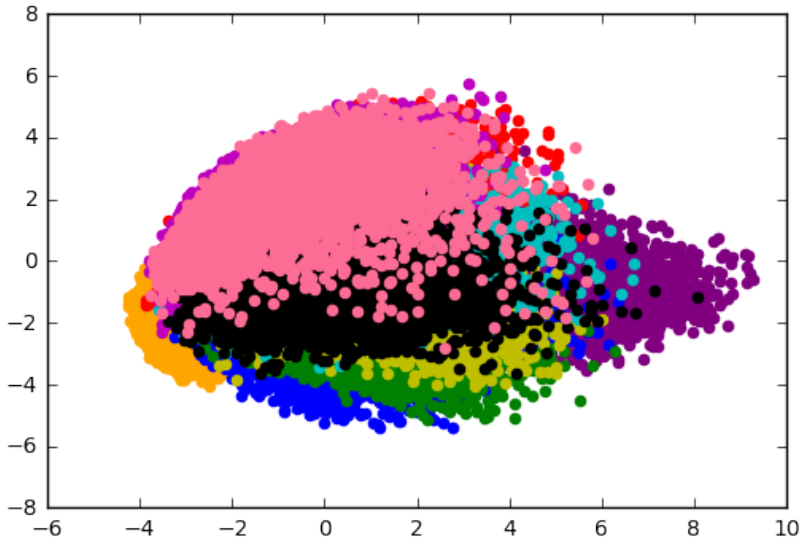


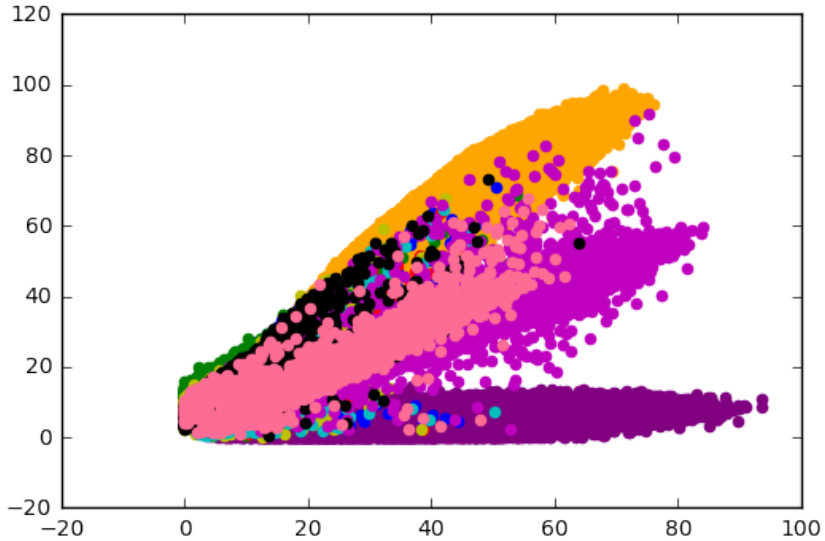
Fig. 3. (**A**) The two-dimensional codes for 500 digits of each class produced by taking the first two principal components of all 60,000 training images. (**B**) The two-dimensional codes found by a 784-1000-500-250-2 autoencoder. For an alternative visualization, see (*8*).

Reproduction de Hinton et Salakhutdinov, Science

PCA :

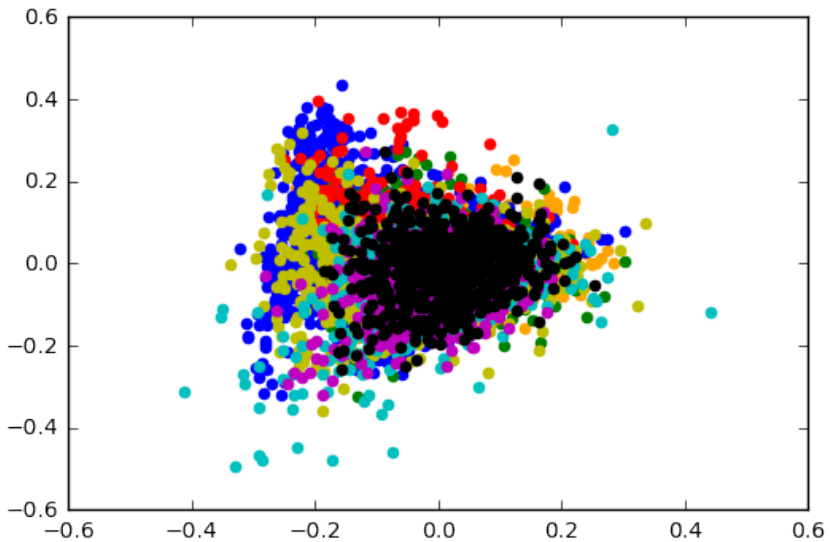


Deep AE non linéaire (5 couches), sans préentraînement

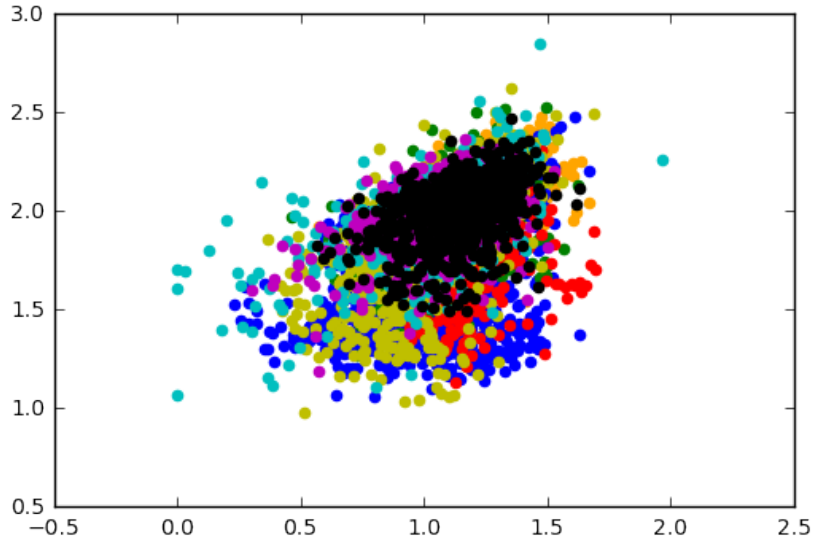


Exercice de visualisation sur données électorales

11 scores par bureau de vote (en % des inscrits). Visualiser les départements dans l'espace politique (Ain, Rhône, Savoie, Haute-Savoie, Isère, Drôme, Ardèche, Loire)



PCA



Deep AE (3 couches)

Les faiblesses de l'AE non-linéaire

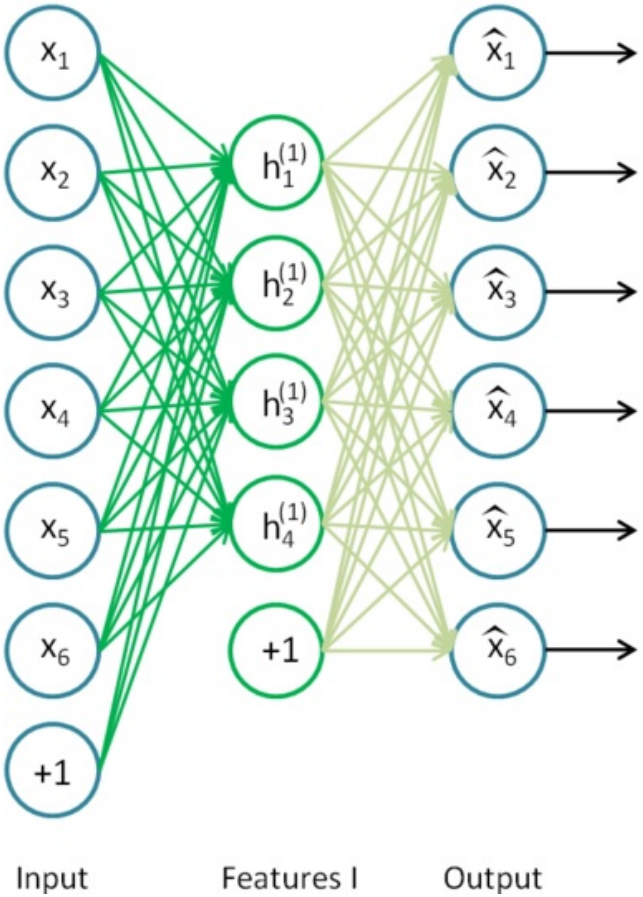
- Conclusions de nos implémentations : Plus d'erreurs de classification sur MNIST, erreurs de débruitage, visualisation peu convaincante...
- Problème principal (Cf. article de Hinton et Salakhutdinov) :
  - Des gros poids initiaux : optimaux locaux
  - Des petits poids : l'entraînement du réseau de neurones quasiment impossible.
  - Si les poids sont proches de la valeur effective, la descente de gradient est alors performante.
- Solution pour rendre l'entraînement efficace : bien initialiser les poids cad phase de préentraînement

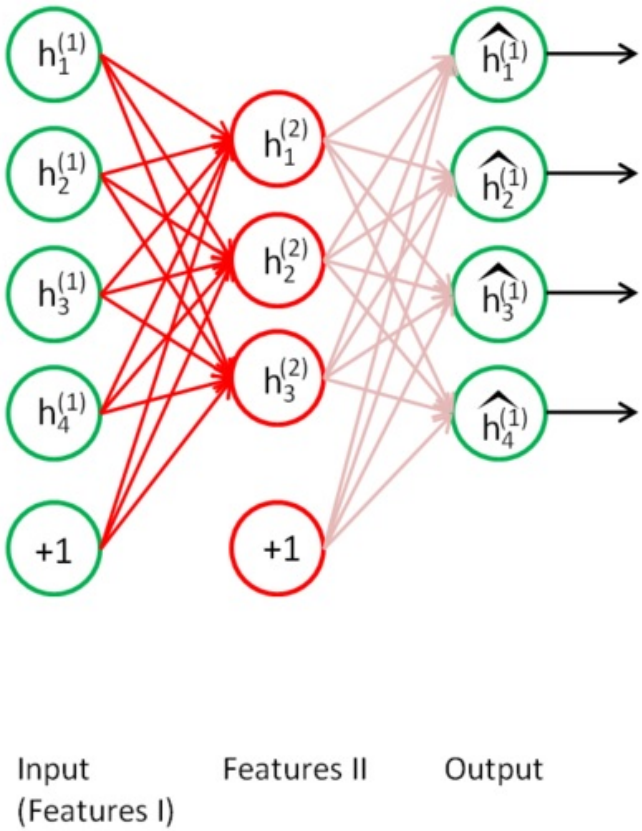
Pistes pour améliorer les performances

- Préentraîner chaque couche à l'aide d'une *Restricted Bolztmann Machine* (RBM) :
  - Poids mis à jour successivement pour chaque couche de compression
  - Processus réitéré autant que possible
- Préentraîner chaque couche à l'aide d'un AE (stacked AE) : Entraînement successif de chaque couche cachée selon le mécanisme ci-dessous.

Pistes pour améliorer les performances

Mécanisme du stacked AE :





Plan

Différentes conclusions de notre étude :

- Equivalence entre AE linéaire et PCA
- PCA globalement plus performante que deep AE non préentraîné
- Nécessité de préentraîner un AE via RBM ou stacking par exemple

Nous vous remercions pour votre attention.