Unsupervised Learning:
Deep Auto-encoder
Auto-encoder

28 x 28 = 784

Usually <784

Compact representation of the input object

Learn together

Can reconstruct the original object
Starting from PCA

Minimize $(x - \hat{x})^2$

As close as possible

Input layer $x$

$W$

$\text{encode}$

Hidden layer (linear) $c$

$W^T$

$\text{decode}$

Output layer $\hat{x}$

Bottleneck later

Output of the hidden layer is the code
Deep Auto-encoder

- Of course, the auto-encoder can be deep

As close as possible

Initialize by RBM layer-by-layer

Symmetric is not necessary.

Deep Auto-encoder

Original Image

PCA

Deep Auto-encoder
Pokémon

- [http://140.112.21.35:2880/~tlkagk/pokemon/pca.html](http://140.112.21.35:2880/~tlkagk/pokemon/pca.html)
- [http://140.112.21.35:2880/~tlkagk/pokemon/auto.html](http://140.112.21.35:2880/~tlkagk/pokemon/auto.html)
- The code is modified from
Auto-encoder – Text Retrieval

**Vector Space Model**

- Query
- Document

**Bag-of-Word**

- Word string: “This is an apple”
- Tokens:
  - this: 1
  - is: 1
  - a: 0
  - an: 1
  - apple: 1
  - pen: 0

Semantics are not considered.
Auto-encoder – Text Retrieval

The documents talking about the same thing will have close code.

LSA: project documents to 2 latent topics

Bag-of-word
(document or query)
Auto-encoder – Similar Image Search

Retrieved using Euclidean distance in pixel intensity space

(Images from Hinton’s slides on Coursera)

Auto-encoder – Similar Image Search

(crawl millions of images from the Internet)
Retrieved using Euclidean distance in pixel intensity space

retrieved using 256 codes
Auto-encoder – Pre-training DNN

- Greedy Layer-wise Pre-training again
Auto-encoder – Pre-training DNN

• Greedy Layer-wise Pre-training again

Input 784

Output

Target

1000

1000

1000

500

10

\( x \)

\( \hat{a}^1 \)

\( a^1 \)

\( W^1 \)

\( W^2 \)

\( W^{2'} \)
Auto-encoder – Pre-training DNN

• Greedy Layer-wise Pre-training *again*
Auto-encoder – Pre-training DNN

• Greedy Layer-wise Pre-training *again*

Target

<table>
<thead>
<tr>
<th></th>
<th>Output</th>
<th>Input</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10</td>
<td>784</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td></td>
</tr>
</tbody>
</table>

Find-tune by backpropagation

Random init

<table>
<thead>
<tr>
<th></th>
<th>Output</th>
<th>Input</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10</td>
<td>784</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td></td>
</tr>
</tbody>
</table>

\[ \begin{align*}
   x &= W^1 \cdot \begin{bmatrix} 1000 \\ 1000 \end{bmatrix} \\
   &= W^2 \cdot \begin{bmatrix} 500 \\ 500 \end{bmatrix} \\
   &= W^3 \cdot \begin{bmatrix} 10 \end{bmatrix} \\
   &= W^4 \cdot \begin{bmatrix} 784 \end{bmatrix}
\end{align*} \]
Auto-encoder

• De-noising auto-encoder

As close as possible


Learning More
- Restricted Boltzmann Machine

- Neural networks [5.1] : Restricted Boltzmann machine – definition
  - https://www.youtube.com/watch?v=p4Vh_zMw-HQ&index=36&list=PL6Xpj9I5qXYEcOhn7TqghAJ6NAPrNmUBH

  - https://www.youtube.com/watch?v=lekCh_i32iE&list=PL6Xpj9I5qXYEcOhn7TqghAJ6NAPrNmUBH&index=37

- Neural networks [5.3] : Restricted Boltzmann machine - free energy
  - https://www.youtube.com/watch?v=e0Ts_7Y6hZU&list=PL6Xpj9I5qXYEcOhn7TqghAJ6NAPrNmUBH&index=38
Learning More
- Deep Belief Network

- Neural networks [7.7] : Deep learning - deep belief network
  - https://www.youtube.com/watch?v=vkb6AWYXZ5I&list=PL6Xpj9I5qXYEcOhn7TqghAJ6NAPrNmUBH&index=57

- Neural networks [7.8] : Deep learning - variational bound
  - https://www.youtube.com/watch?v=pStDscJh2Wo&list=PL6Xpj9I5qXYEcOhn7TqghAJ6NAPrNmUBH&index=58

- Neural networks [7.9] : Deep learning - DBN pre-training
  - https://www.youtube.com/watch?v=35MUIYCColk&list=PL6Xpj9I5qXYEcOhn7TqghAJ6NAPrNmUBH&index=59
Auto-encoder for CNN

As close as possible
CNN - Unpooling

Alternative: simply repeat the values

Source of image:
https://leonardoaraujosantos.gitbooks.io/artificial-intelligence/content/image_segmentation.html
Actually, deconvolution is convolution.
Next ..... 

Can we use decoder to generate something?
Can we use decoder to generate something?