EMERGENCE OF SIMPLE-CELL RECEPTIVE PROPERTIES BY LEARNING A SPARSE CODE FOR NATURAL IMAGES

Bruno A. Olshausen & David J. Field

Presenter: Ozgur Yigit Balkan

Outline

• Linear data representations

• Sparse Vector Recovery
  • Existing Methods

• Dictionary Learning (Sparse Coding)

• Why is sparseness important?

• How to learn the dictionary?

• Experiments

• Conclusion
Linear Data Representation

- PCA – ICA – Sparse Coding

- Underlying model is linear.

- Each data point is assumed to be created with the following linear model

\[ y = A x \]

- \( y \in \mathbb{R}^M \), is the data vector (observed)

- \( A \in \mathbb{R}^{M \times N} \), is the set of basis vectors (columns of \( A \) are basis vectors \( a_i \)).

- \( x \in \mathbb{R}^N \) is the weights vector. (unknown)

What is Sparse Coding?

\[ Y = AX \]

- **PCA**: Given \( Y = \{y_1, y_2, \ldots, y_L\} \). Creates an orthogonal basis set \( A \), such that the underlying sources (weights) are uncorrelated.

- **ICA**: Given \( Y = \{y_1, y_2, \ldots, y_L\} \). Creates a basis set of vectors as columns of \( A \), such that the underlying sources are independent.

- **Sparse Coding**: Given \( Y = \{y_1, y_2, \ldots, y_L\} \). Find set of basis vectors \( A \) such that the associated vectors \( x_i \) are sparse.
  - Usually called “dictionary learning”.

\[ [y_1 \ y_2 \ \ldots \ y_L] = A [x_1 \ x_2 \ \ldots \ x_L] \]
Sparse Vector Recovery

- Measure of sparseness?
- $\|x\|_p^p = \left(\sum_{j=1}^{N} (x^p)^j\right) .$
- Diversity: Number of nonzero elements in a vector.
  - $D(x) = \|x\|_p^p$ as $p \to 0.$
- If $x \in R^N$, the measure of sparseness is: $N - \|x\|_p^p$ as $p \to 0.$

Sparse Inverse Problem:

\[
\min ||x||_0 \quad \text{s.t} \quad y = Ax
\]

- Used columns in $A$ are called the support set. Finding support set is equivalent to finding $x$. Global solution is NP-hard.

Some of the existing methods

- Convex relaxation of objective function $\|x\|_0$
  \[
  \min ||x||_1 \quad \text{s.t} \quad y = Ax
  \]
- In noisy cases,
  \[
  \min ||x||_1 \quad \text{s.t} \quad ||y - Ax||_2 \leq \varepsilon
  \]
- Other possible formulations
  \[
  \min ||y - Ax||_2 \quad \text{s.t} \quad ||x||_1 \leq k
  \]
  - or,
  \[
  \min ||y - Ax||_2 + \lambda ||x||_1
  \]
Greedy Pursuit Methods

- These methods, in general, start with an empty support set and adds one of the basis vectors from $\mathcal{A}$ at each iteration.

- $A_S \triangleq$ current support set (initially empty)

- Orthogonal Matching Pursuit
  \[ \arg\max_i \{ r_k^T a_i \} \]
  \[ A_S = A_S \cup \{ a_i \} \]

- Order Recursive Matching Pursuit
  \[ r_S = y - P_S y = P_S^+ y. \]
  \[ \arg\min_i \| r_{SU(i)} \|_2 \]

How well do they work?

- **Theoretical**: These algorithms guarantee to find the global solution if $A$ satisfy certain conditions, which are very conservative conditions and doesn’t apply to real world applications.

  \[ \| x \|_0 < \frac{1}{2} \left( 1 + \frac{1}{\mu} \right) \]

- **Practical**: They work really well, especially convex relaxation methods.

- **Note**: In practice, one doesn’t necessarily need the sparsest solution.
Sparse Coding (Dictionary Learning)

- Problem definition: Given data \(\{y_1, y_2, ..., y_n\}\), find dictionary \(A\), such that associated \(x_i\) could be sparse. (diversity < \(M\))

Why sparseness?

- Each cell has a low probability of activation given a single image (subimage block).
- This simplifies feature detection, e.g. edge detection
- Storage and retrieval with associative memory. (For higher levels of the network)
Why sparseness?

- Sparseness also results in independence.
- \( H(x_1, x_2, \ldots, x_N) = \sum_i H(x_{ij}) \) - mutual info.
- Under the assumption that joint entropy (information in the image) is preserved, if we minimize individual entropies, mutual info gets smaller. Thus, we would get more independent cells.
- \( H(x_{ij}) \downarrow \rightarrow \) sparsity providing prior distributions.
- Minimum entropy code

\[
y_i = Ax_i
\]

Not easy, sparse inverse solution.

How to learn \( A \) or \( a_{ij} \)?

- Assume we are given \( x_i \)'s. So we know \( X, Y \).
- We know the model \( Y = AX \).
- \( A_{sol} = \arg\min_A \| Y - AX \|_F^2 = YY^T(XX^T)^{-1} = YY^+ \).
- However we don’t know \( x_i \)'s upfront.
How to learn $A$?

**Task:** Train a dictionary $A$ to sparsely represent the data $(y_i)_i$, by approximating the solution to the problem posed in Equation (12.2).

- **Initialization:** Initialize $i = 0$, and
- **Initialize Dictionary:** Build $A_{0i} \in \mathbb{R}^{m \times n}$, either by using random entries, or using $n$ randomly chosen example.
- **Normalization:** Normalize the columns of $A_{0i}$.

**Main Iteration:** Increment $i$ by 1, and apply:

- **Sparse Coding Stage:** Use a pursuit algorithm to approximate the solution of
  \[ \hat{x}_i = \arg \min_{x} \|y_i - Ax_i\|^2 \text{ subject to } |x| \leq k. \]
  obtaining sparse representations $\hat{x}_i$ for $1 \leq i \leq M$. These form the matrix $X_{A_0}$.
- **MOD Dictionary Update Stage:** Update the dictionary by the formula
  \[ A_{0i+1} = \arg \min_{A} \|Y - AX_{0i}\|^2 + \lambda \sum_{ij} \log(1 + x_{ij}^2). \]
- **Stopping Rule:** If the change in $\| Y - A_{0i}X_{0i} \|^2$ is small enough, stop. Otherwise, apply another iteration.

**Output:** The desired result is $A_{0n}$.

---

**How to learn $A$ or $a_{ij}$? (Olshausen & Field paper)**

- $E = -(\text{preserve information}) - \lambda (\text{sparseness of } x_i)$
  
  (preserve information) = $-\|Y - AX\|^2$.

  (sparseness of $x_i$) = $-\sum_j S \left( \frac{v_j}{\sigma} \right)$.

- Possible $S(x)$ could be $-e^{-x^2}$, $\log(1 + x^2)$, $|x|$.
- They are all unimodal and peaked around 0.
- Minimize $E$.

- $\min_{AX} \| Y - AX \|^2 + \lambda \sum_{ij} \log(1 + x_{ij}^2)$
How to learn $A$ or $a_{ij}$? (Olshausen & Field paper)

- $E = -(\text{preserve information}) - \lambda (\text{sparseness of } x_i)$
  
  (preserve information) = $-\|Y - AX\|^2_F$.
  
  (sparseness of $x_i$) = $-\sum_j S\left(\frac{x_{ij}}{\sigma}\right)$.

Algorithm

\[ Y = AX \quad \rightarrow \quad I = \phi A \]

\[ \dot{a}_i = b_i - \sum_j C_{ij} a_j - \frac{\dot{\lambda}}{\sigma} S\left(\frac{a_i}{\sigma}\right) \]

where $b_i = \Sigma_x \phi_i(x,y)I(x,y)$ and $C_{ij} = \Sigma_{x,y} \phi_i(x,y)\phi_j(x,y)$.

\[ \Delta \phi_i(x_m,y_m) = \eta \left( a_i \left[ I(x_m,y_m) - \hat{I}(x_m,y_m) \right] \right) \quad (6) \]

where $\hat{I}$ is the reconstructed image, $\hat{I}(x_m,y_m) = \Sigma_i a_i \phi_i(x_m,y_m)$, and $\eta$ is the learning rate.
Experiments (Artificial data)

<table>
<thead>
<tr>
<th>Training set</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sparse pixels</td>
<td>![Result Image]</td>
</tr>
<tr>
<td>Sparse gratings</td>
<td>![Result Image]</td>
</tr>
<tr>
<td>Sparse gabor</td>
<td>![Result Image]</td>
</tr>
</tbody>
</table>

Dictionary learning algorithm worked for artificial data.

Experiments (Natural Images)

Ten 512x512 natural images in the American northwest

16x16 image patches

$S(x) = \log(1 + x^2)$.

192 basis functions

Spatially localized, oriented, band-pass (selective to structure at different spatial scales)
Experiments (Natural Images)

Spatially localized, oriented, band-pass (selective to structure at different spatial scales)

Experiments (Natural Images)

Ten 512x512 natural images in the American northwest

16x16 image patches

\[ S(x) = \log(1 + x^2) \]

192 basis functions

Decreased entropy, increased independence.
Conclusion

- Dictionary learning problem although it is hard, can be solved iteratively by first learning the sparse activations and updating the dictionary and so on.

- Sparsification as a preprocessing step is desirable since it has benefits of feature detection, memory storage, and independence etc.

- Sparsification for natural images results in spatially localized, oriented, and bandpass filters (basis functions), just like V1.

Note: Discovery of sensory filters by dictionary learning are not limited to the visual inputs but also auditory.

Sparse Coding for audio?

Cat auditory nerve fibres

Evan Smith & Lewicki, Nature 2006
Sparse Coding for audio?

Learned basis functions

Evan Smith & Lewicki, Nature 2006

Questions?