New Algorithms for Learning Incoherent and Overcomplete Dictionaries

ICERM Workshop, May 7

Sanjeev Arora
Princeton

Rong Ge
Microsoft Research

Tengyu Ma
Princeton

Ankur Moitra
MIT
Dictionary Learning

- Simple “dictionary elements” build complicated objects

- Given the objects, can we learn the dictionary?
Why dictionary learning? [Olshausen Field ’96]

natural image patches

dictionary learning

Gabor-like Filters
Example: Image Completion [Mairal, Elad & Sapiro ’08]
Outline

• Dictionary Learning problem

• Getting a crude estimate

• Refining the solution
Dictionary Learning Problem

• Given samples of the form $Y = AX$
• $X$ is a sparse matrix
• Goal: Learn $A$ (dictionary).
• Interesting case: $m > n$ (overcomplete)
Previous Approach

Dictionary A

LASSO
Basis Pursuit
Matching Pursuit

Least Squares
K-SVD

Sparse Code X

Alternating Minimization
Problem with Alternating Minimization

- Monotone objective function?
  - Local Minimum Issues
Empirical Behavior

- **Synthetic Experiment**, “qualitative” plot
- **K-SVD** converges with prob. $1/3$, random samples as initial dict.
Provable Algorithms

• Run in poly time, uses poly samples, learn the ground truth

• Separate modeling and optimization error

• Design new algorithms/Tweak old algorithms

• Work only on “reasonable instances”
When is the solution “reasonable”?

• Consider Image Completion

- Dictionary

- Image

• The representation should be unique and robust!
Sparse Recovery

- Given $A, y$, find $x$.

- **Incoherence** [Donoho Huo ’99]

- Dictionary elements have inner-product $\mu/\sqrt{n}$

- Solution is **unique** and **robust**

- Long line of work [Logan, Donoho Stark, Elad, .......]

- Sparsity up to $\sqrt{n}$
Our Results

• Thm: If dictionary $A$ is incoherent, $X$ is randomly $k$-sparse from “nice distribution”, learn dictionary $A$ with accuracy $\varepsilon$ when sparsity

$$k \leq \min\left\{ \sqrt{n}/\mu \log m, m \uparrow 0.4 \right\}$$

• Handles sparsity up to $\sqrt{n}$
• Sample complexity $O^*(m/\varepsilon^2)$.

Independently [Agarwal et al.] obtain similar result with slightly different assumptions and weaker sparsity

Later [Barak et al.] get stronger result using SOS
Our Results

• Thm: Given an estimated dictionary $\epsilon$-close to true dictionary, one iteration of K-SVD outputs a $\epsilon/2$-close dictionary

• Works whenever $\epsilon<1/\log m$ (before require $1/poly$).

• Sample complexity $O(m\log m)$

• Combine: Can learn an incoherent dictionary with $O^*(m)$ samples in poly time.
Outline

• Dictionary Learning problem

• Getting a crude estimate

• Refining the solution
Ideas

- Find the **support** of \( X \), **without** knowing \( A \).

- Given **support** of \( X \), find approximate \( A \).
Finding the Support

- Tool: Test whether two columns of $X$ intersect

Disjoint $\approx$ Small Inner-product
Intersect $\approx$ Large Inner-product
Finding the Support: Overlapping Clustering

• Connect pairs of samples with large inner-product
• Vertex = Sample
• Cluster = Rows of X!

[Diagram showing overlapping circles representing clusters and nodes connected by edges]
Overlapping Clustering

• Main problem

• Idea: Count the number of common neighbors
• pair of points share unique cluster ➔ cluster
Estimate Dictionary Elements

- Focus on a row of $X$/column of $A$
  - Can use **SVD** to find **maximum variance** direction
  - Or take samples with **same sign** and average
Outline

• Dictionary Learning problem

• Getting a crude estimate

• Alternating Minimization Works!
K-SVD [Aharon, Elad, Bruckstein 06]

- Given: a good guess \((A)\)
- Goal: find a even better dictionary
- Update one dict. element:
  - Take all samples with the element
  - Decode: \(y \approx Ax\)
  - Residual: \(r = y - \sum_{j \neq i} A_j x_j = \pm A_i + \sum_{j \neq i} (A_j x_j - A_j x_j)\)
  - Use top singular vec of residuals
K-SVD illustrated

Blue: True dictionary
Dashed: Estimated Dictionary

Take all samples with same element
Compute Residual

Hope: In residuals, noise is small and random, top singular vector robust for random noise
K-SVD: Intuition

• When error \((A \downarrow i - \hat{A} \downarrow i)\) is random
  • Still incoherent
  • Can “decode”
  • Noise looks random

• When error is adversarial
  • May not be incoherent
  • Noise can be correlated

• Bad case: error is highly correlated, pointing to same direction
make the noise “random”

• Observation: Can detect the bad case!

• To handle bad case, need to
  • Perturb the estimated dictionary
  • Keep perturbation small
  • The result has low spectral norm

\[ \min \| B \| \text{ s.t. } \| B \downarrow_i - A \downarrow_i \| \leq \epsilon \]

Large Singular value!

Convex!

\[ \text{OPT} \leq \| A \| \]
Low spectral norm is enough

• Key Lemma: When $B$ has small spectral norm, $|<B_i,B_j>| \leq \frac{1}{\log m}$, random $k$ columns of $B$ are “almost orthogonal”

• Decoding is accurate for a random sample

• Proof sketch: For $B^TB$
  • Diagonals are large
  • Off-diagonals are small in $\text{Exp}$.
  • Concentration $\Rightarrow$
    random submatrix is diag. dominant
Conclusion & Open Problems

• **K-SVD** works provably with **good initialization**

• Does the proof give any insight in practice?
  • Whitening

• “Error behaves random” useful in other settings?

• Handle larger sparsity?
  • work with **RIP** assumption?

• Lowerbounds?

**Thank you!**
Thank you!

Questions?
**K-SVD** [Aharon, Elad, Bruckstein 06]

- **Given:** a *good* guess
- **Goal:** find a even *better* dictionary

- **Update one dict. element:**
  - Take all samples with the element
  - Remove *other* elements
  - Use *top* singular vec of residuals

- **Hope:** In *residuals*, error is small and *random*, **SVD** robust for random noise
Other Applications

Image Denoising [Mairal et al. ’09]

Digital Zooming [Couzinie-Devy ’10]
Applications

Image Completion [Mairal, Elad & Sapiro ’08]

Image Denoising [Mairal et al. ’09]

Digital Zooming [Couzinie-Devy ’10]
Refining the solution

- Use other columns to reduce the variance!
- Get $\varepsilon$ accuracy with $\text{poly}(m,n) \log \frac{1}{\varepsilon}$ samples