SPARSE SUBSPACE CLUSTERING (SSC) AND DIMENSIONALITY REDUCTION (DR)

Mehmet Furkan Demirel

Dr. Enrico Au-Yeung

INTRODUCTION I - SSC

Many real-world problems deal with collections of high-dimensional data—such as images, videos, sound recordings, web documents, and DNA microarray data. When taken into account, these high-dimensional data lie close to low-dimensional structures corresponding to several classes or categories to which the data belong. In the first part of this project, we implement an algorithm—called Sparse Subspace Clustering—that can cluster a set of multi-space data using sparse representation techniques. We, then, experiment with the algorithm using a large collection of video frames from an open-source movie (Elephants Dream) and observe its capability to identify the subspace clusters based on each image's ability to be expressed as a linear combination of the others. The key idea is that, among the infinitely many possible representations of a data-point in terms of other points, a sparse representation corresponds to selecting a few points from the same subspace. In other words, we make an attempt to express a vector (whether it is an image or a text) as a linear combination of other vectors—in a sufficiently sparse manner.

INTRODUCTION II - DR

In machine learning and statistics, dimensionality reduction is the process of reducing the number of random variables under consideration, via obtaining a set of principal variables. Given a manifold in a high-dimensional space, the purpose is to come up with a representation of that manifold in a lower-dimensional space while preserving its overall structure. In the second part of this project, we implement an algorithm that can generate increasingly-fine covers of a high-dimensional manifold in a low-dimensional space. The algorithm works similar to a greedy algorithm-–making the locally optimal choice at each step. Nonetheless, it is worth to mention that there is no obvious reason to believe that the greedy approach will always give us the global optimum. As we continue working on this project, we hope to explore a better method to design these covers by performing global optimization.

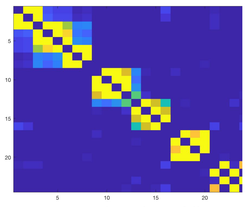
SSC METHOD AND EXPERIMENT

For the SSC algorithm, we express the following sparse optimization program.

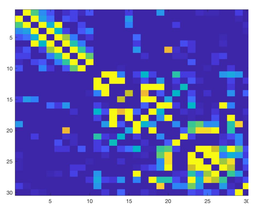
$$\min \|\boldsymbol{C}\|_1 \quad \text{s. t.} \quad \boldsymbol{Y} = \boldsymbol{Y}\boldsymbol{C}, \quad \operatorname{diag}(\boldsymbol{C}) = \boldsymbol{0},$$

where Y is a self-expressive dictionary in which each point can be written as a linear combination of other points. C is the matrix whose i-th column corresponds to the sparse representation of y_i, c_i, and *diag(C)* is the vector of diagonal elements of C.

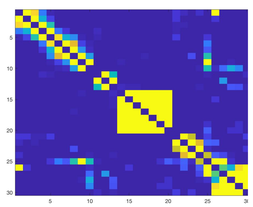
Experiment: We obtain a number of film frames from an open-source movie called Elephants Dream and run them in our algorithm to see how similar they are to each other. The following are the results:



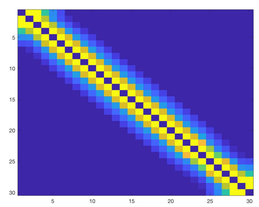
24 images chosen from 3 different sections of the movie (≤ 5 frames apart)



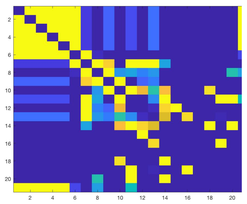
30 images chosen from 3 different sections of the movie (exactly 25 frames apart)



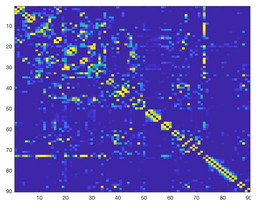
30 images chosen from 3 different sections of the movie (≤ 25 frames apart)



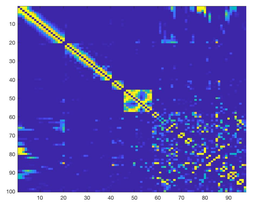
30 images chosen from one specific part of the movie (exactly 1 frame apart)



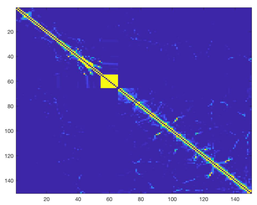
21 images chosen from 3 different sections of the movie (Fibonacci - exponentially increasing frames apart)



90 images chosen from 3 different sections of the movie (exactly 24 frames apart)



100 images chosen from 5 different sections of the movie (the frame difference increases as we go from the first cluster to the fifth)



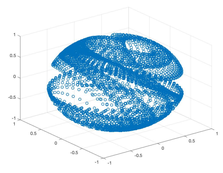
150 images chosen from one specific section of the movie (exactly 1 frame apart)

DIMENSIONALITY REDUCTION METHOD AND EXPERIMENT

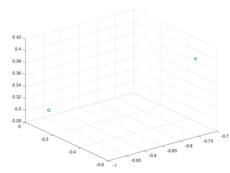
As for the Dimensionality Reduction algorithm, we design our covers according to the following chaining argument:

$$\begin{aligned}
&\mathrm{P}\left\{ \sup_{x_1,x_2\in\mathcal{M}} \frac{\|\Phi x_1 - \Phi x_2\|}{\|x_1 - x_2\|} > 1+\epsilon \right\} \\
&\quad = \mathrm{P}\left\{ \sup_{y\in U(\mathcal{M})} \|\Phi y\| > 1+\epsilon \right\} \\
&\le \mathrm{P}\left\{ \sup_{y\in U(\mathcal{M})} \|\Phi\pi_0(y)\| + \sum_{j\ge 1} \sup_{y\in U(\mathcal{M})} \|\Phi(\pi_j(y) - \pi_{j-1}(y))\| > 1 + \sum_{j\ge 0}\epsilon_j \right\} \\
&\le \mathrm{P}\left\{ \max_{p\in C_0} \|\Phi p\| + \sum_{j\ge 1} \max_{(p,q)\in C_j\times C_{j-1}} \|\Phi(p-q)\| > 1 + \sum_{j\ge 0}\epsilon_j \right\} \\
&\le \#C_0 \cdot \max_{p\in C_0} \mathrm{P}\{\|\Phi p\| > 1+\epsilon_0\} + \sum_{j\ge 1} \#C_j \cdot \#C_{j-1} \max_{(p,q)\in C_{j+1}\times C_j} \mathrm{P}\{\|\Phi(p-q)\| > \epsilon_j\},
\end{aligned}$$

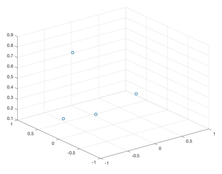
where ϕ is a measurement operator, M is a manifold in the high dimensional space, U(M) is the set of all normalized secants of M, and $\pi_j(y)$ represents the nearest point to y on the j-th cover.



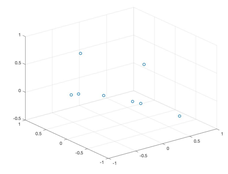
U (the set of all normalized secants of M in three-dimensional space)



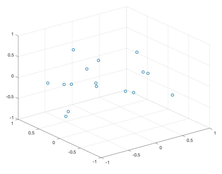
C_1 with 2 vectors



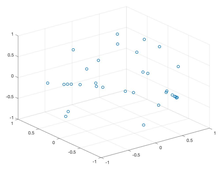
C_2 with 4 vectors



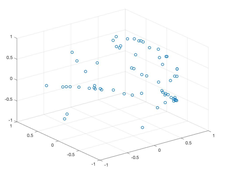
C_3 with 8 vectors



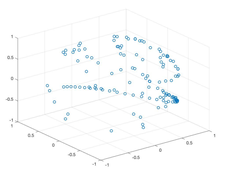
C_4 with 16 vectors



C_5 with 32 vectors



C_6 with 64 vectors



C_7 with 128 vectors

CONCLUSION I

In the SSC experiment that we performed, we can observe that our algorithm was able to identify the given clusters in the matrix representations of linear combinations. We see that as long as the "frame-difference" between images in the same cluster is around 25, the algorithm does a really good job of separating different sets of similar images. We believe that this observation is heavily linked to the fact that every second in a movie corresponds to 24 frames.

In particular, in the experiment with 30 images from a specific part of the movie, we can see that the resulting matrix tells us every image can be written as linear combination of a few images around it, which is very sparse.

CONCLUSION II

In the Dimensionality Reduction experiment, we demonstrated that our algorithm was able to design increasingly-fine covers of the original manifold in a lower dimension. As can be seen, as j increase, the j-th cover looks more and more like the original U, which is the set of all normalized secants of M. It is worth to note that, for the purpose of providing visual material, the experiment mentioned in this poster was performed on a manifold in 3-D, which is not very high-dimensional. Nonetheless, the algorithm works with manifolds in higher dimensions as well.

Again, we believe that the greedy approach does not necessarily give the best covers all the time. Considering that the project is ongoing, we hope to be able to design these covers by performing global optimization—rather than finding local optimum at each step of the process.

REFERENCES

Eftekhari, Armin and Michael B. Wakin. "New analysis of manifold embeddings and signal recovery from compressive measurements." Applied and Computational Harmonic Analysis 39.1 (2015): 67-109.

Elhamifar, Ehsan and Rene Vidal. "Sparse Subspace Clustering: Algorithm, Theory, and Applications." IEEE Transactions on Pattern Analysis and Machine Intelligence 35.11 (2013): 2765-2781.