Subspace Clustering in Julia

Sophie Champ Michael Peralta Sarah Vaknin
Faculty Advisor: Daniel Robinson

Problem

Subspace Clustering: A technique which finds clusters within a set of unmarked data that represent distinct subspaces of possibly different dimensions.

Objectives

- Build a subspace clustering algorithm in the relatively new programming language called Julia.
- Test the subspace clustering algorithm on a dataset of tens of thousands of points with differing subspace dimensions.

Project Motivation

Subspace Clustering of Human Faces: A subspace clustering algorithm could take a dataset of a tens of thousands of photographs of different people, with varying light levels and discern commonalities in the data to cluster the photos by person, or by face.

Motion Segmentation: If many screenshots are taken of moving cars and points are assigned to each car, then the subspace clustering algorithm would be able to take in that dataset of car locations and cluster each vehicle in its own subspace.

Lasso Problem

$$c_j = \underset{c}{argmin}\,\frac{1}{2}||x_j - Xc||_2^2 + \lambda|c|_1$$

- λ a positive weighting parameter
- X is the data matrix of concatenated data from each subspace with the j^{th} column removed
- x_j represents the j^{th} column of the data matrix

There are two terms that form the Lasso problem:

- The first term finds a vector, c, that is a linear combination of the data
- The second term uses λ to penalize the function which gives the vector sparseness.

Synthetic Data Generation

- Creates n (subspace dimension) by d (ambient dimension) matrix of random numbers
- Executes the Gram-Schmidt process in order to orthogonalize the basis
- Creates another random matrix of dimensions N (number of data points) by d
- Takes linear combination of random matrix and orthogonal basis to obtain subspace data
- Normalizes subspace if user-specified to generate normalized data set

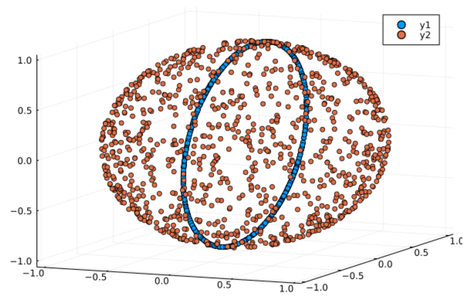
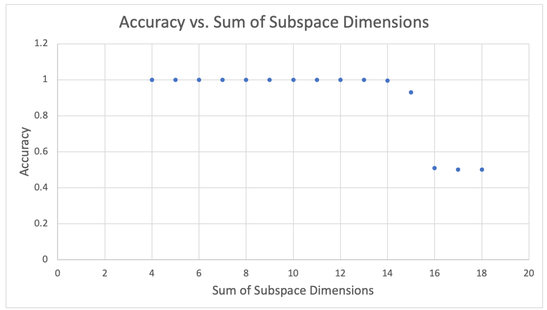


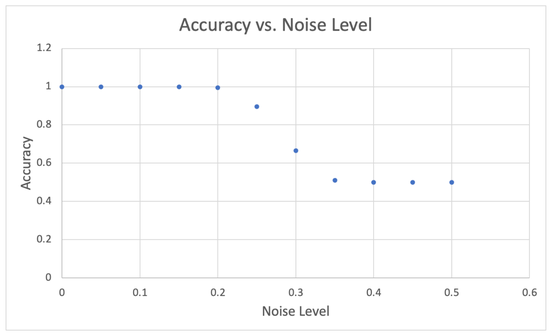
Fig 1. Example of normalized synthetic data for 2 subspaces in ambient dimension 3

- Blue dataset y1: Subspace dimension of 2, 1000 points
- Red dataset y2: Subspace dimension of 3, 1000 points

Testing Performance On Synthetic Data



- Clustering performance for increased sum of subspace dimensions.
- Two subspaces of 100 points each with ambient dimension 15 and subspace dimension 2.
- Increased the subspace dimension of one of the subspaces by one until accuracy leveled off.



- Clustering performance for increased levels of noise in synthetic data.
- Two subspaces of 100 points each with ambient dimension 10 and subspace dimension 2.
- Noise for each data point is drawn from a normal distribution with an increasing standard deviation, which is represented as Noise Level.

Future Steps

- Test the algorithm's performance on real, non-synthetic sets of data
- Apply algorithm to real world problems
- Find ways to optimize and further improve the current algorithm, including researching a lasso solver that can efficiently solve a dataset of more than 10,000 points.

Challenges and Lessons Learned

- Translating from MATLAB and Julia
- How to debug code and troubleshoot
  - Adjusting threshold value for Eigenvalues
  - Forcing symmetry in Affinity matrix and Laplacian
- Varying performance of built-in solvers
- Teamwork and task delegation
- Communication and presentation

Acknowledgements

Thank you to Ana Alexandrescu and the Lehigh University ISE Department