Assignment 1

Tutors:
Group members:
Daniel Martinez Formoso
Philip Grenfell
James Macdonald

Abstract

Non-negative Matrix Factorization (NMF) has found popular use in the field of image compression, where it is well suited to the domain for two reasons. Firstly, images are naturally non-negative in their data. Secondly, NMF works well with convex loss-functions that can provide a stable solution with guarantees around accuracy and required iterations. This report will investigate the base NMF algorithm and the effect of alterations to the algorithm on convergence behaviour and robustness in the face of noise.

1 Introduction

Non-negative matrix factorization (NMF) has found widespread use in fields where an intuitive representation of the base data is desirable. Examples of this include image factorization, text analysis, and product recommendation systems. In these problems, negative values are both unintuitive and physically impossible. While an algorithm like PCA would outperform NMF in these domains, the advantages of having a purely positive solution outweigh the cost of a worse optimization measured against the loss function. For example, in the case of learning facial images, an NMF solution can produce a set of basis faces that can then be additively used to reconstruct the original data. The basis faces that are learned in this process are useful beyond their role in compressing the image.

This paper will examine NMF in the following sections. The first section will contain a brief look at seminal papers and important developments in NMF research. The second section will outline the methodology of the investigation including the cost function, update algorithms, noise generation and evaluation metrics. The third section will review the experiment and results.

2 Related work

An early mention of NMF techniques can be found in Paatero and Tapper (1994) which describes a “Positive Matrix Factorization” paradigm as an alternative to both PCA and Factor Analysis. The paper touts the importance of non-negativity constraints in “environmental applications”, and suggesets unintuitive results is a cause that competing methods “have only found limited applications” in that domain. The paper sets up the fundamentals of the algorithm, describing two matrices of equal rank that are solved by Alternating Regression (AR), whereby one of the matrices is kept constant and the other is solved, then vice versa, repeating until a solution is reached or the number of iterations is exhausted.

Another influential paper, Lee and Seung (1999) explored the potential of NMF in two key areas, text data and image data (in fact, the authors specifically examine NMF performance in facial image data, which will be explored later in this piece). The paper outlines how the algorithm distills the data down to a set of basis faces or *eigenfaces*, using linear combinations of these bases to recreate the original input data. This is given mathematically as $V \approx (WH)$, where V is the original set of images, W is a set of basis images and H encodes the original information as a sparse addition of the basis images. Like the previously discussed paper, Lee and Seung (1999) use an alternating iterative update algorithm to fix H while updating W, and vice versa.

The fundamental idea has been exploited in a number of variations. In Hamza and Brady (2006), the authors employ an adjusted cost function that behaves linear when large, but quadratic when small. Theoretically this should make it more robust to outliers by de-emphasizing the penalty of larger error terms. Another regularization technique is to introduce a penalty term proportional to the L1 Norm of the data. This approach has the added benefit of encouraging sparsity into the solution.

3 Methods

This experiment will analyse the performance of three non-negative matrix factorization (NMF) algorithms on facial image data. Firstly, hypersurface cost-function NMF as described in Hamza and Brady (2006). Secondly, L1-norm regularized NMF which includes a penalty term to reduce overfitting and encourage sparsity. Lastly, these variations will be compared against a vanilla application of NMF.

Two image datasets are used. The first consists of photos from the Yale Face Database which have been cropped to keep the position of key facial features consistent between photos. The second is the ORL dataset, consisting of multiple photos per individual, taken from different angles. The first dataset poses a much simpler problem, as the algorithm does not need to adjust for the shifting relative position of facial features as angles change.

In order to test the algorithms further, block occlusion was introduced into the features. Rectangles of pixels were set to the max pixel value, hiding the information in those pixels from the algorithm. This is shown in figure 1.

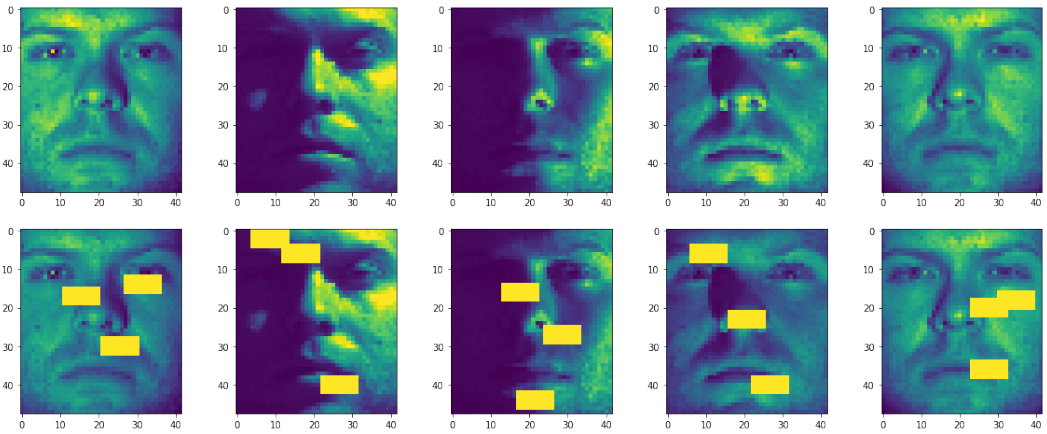

Figure 1: Example cropped faces from the Yale dataset. The top image shows the original, the bottom shows randomly generated occlusion

In order to measure the the performance of these different algorithms, the following metrics will be considered:

3.1 Loss

Loss is the difference between the original matrix and the reconstruction, and can be considered what information is 'lost' in the matrix factorization process. This exercise uses squared loss, given by $(Y - HW)^2$ where H and W are the dictionary and compressed representation respectively, and Y is the original image.

Squared loss can obscure marginal improvements later in the training process because the gains from early iterations are of a much greater magnitude. This can be rectified by using Logloss, which is given by $ln[(Y - HW)^2]$. This downweights the low-hanging fruit of early iterations and shifts the focus onto late-stage optimization.

3.2 Relative Reconstruction Error

Relative Reconstruction Error (RRE) is similar to loss, but the result is normalized by the original image. RRE is constructed by:

$$RRE = \frac{\|V - WH\|_F}{\|V\|_F}$$

where V is the clean dataset, W and H are the dictionary and encoding respectively, and $\hat{V}$ is the noisy dataset from which W and H are learned.

3.3 Normalized Mutual Information

4 Experiment

4.1 Hypersurface Smoothing

The hypersurface smoothing algorithm was designed primarily to be robust against outliers. The occlusion noise in this experiment can be considered a form of outlier, and the adjusted cost function downweights the relative importance of these pixels.

The hyperparameters were set as follows: Step size was 0.005 for both W and H matrices, rank was 10, and maximum iterations was 5000. The algorithm shows rapid initial gains in decreasing loss, as shown by figure 2. However, the algorithm quickly finds a "floor" beyond which it cannot improve further.

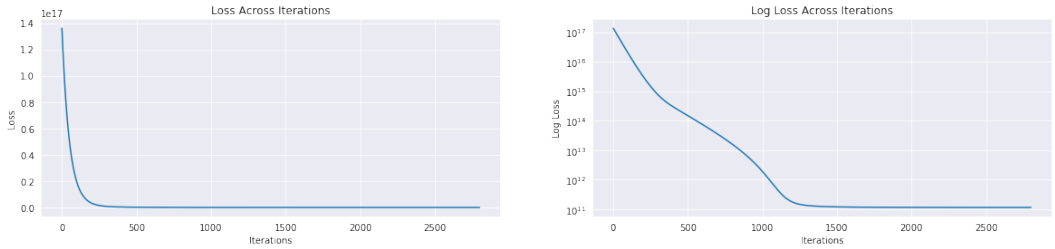

Figure 2: Algorithm performance as a function of training iterations. Loss declines steeply at first but levels off quickly. Log loss shows that performance improves for the initial $\frac{1}{3}$rd of the training process

The algorithm was able to produce a sensible representation of faces from training over the cropped Yale images, as shown in 3. Furthermore, there are noticeable differences between faces. However, there is still substantial differences between the output and the original data.

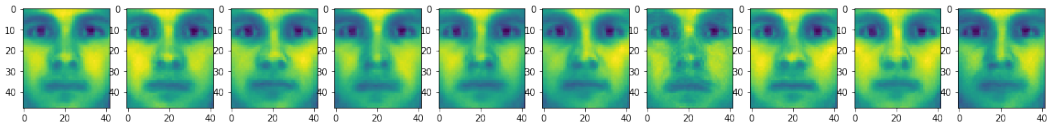

Figure 3: The reconstruction of the data based on the hypersurface algorithm

5 Conclusion

6 Reference

7 References

References

Hamza, A. and Brady, D. (2006). Reconstruction of reflectance spectra using robust nonnegative matrix factorization. *Trans. Sig. Proc.*, 54(9):3637–3642.

Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by nonnegative matrix factorization. *Nature*, 401:788–791.

Paatero, P. and Tapper, U. (1994). Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5:111–126.