Temple University

Art Works:

A Synthesis of Art and Machine Learning

Angela Minster and Eric Stone

3rd May, 2012

Abstract

A visit to the Philadelphia Museum of Art got us thinking: How can we classify, categorize or otherwise extract interesting information or patterns from art when art is viewed as data? A literature review on gaze tracking reveals that art consumers, artists, photographers, and other users and creators of media treat different areas of the canvass differently. It would be interesting if this manifests itself in the representation of popular works of art throughout time. This paper describes methods for observing the distribution of colors in a given painting, artist, or period/genre. We also use several cutting-edge machine learning tools to investigate how the pixels from different quadrants of the canvass relate to one another.

# Introduction

While observing an array of art from different periods and by different artists, it is difficult to imagine there is a uniting characteristic. It seems each has its own particular style, unique both to person and era. We see this when we observe the distributions of colors used, and we see it just from looking at different paintings.

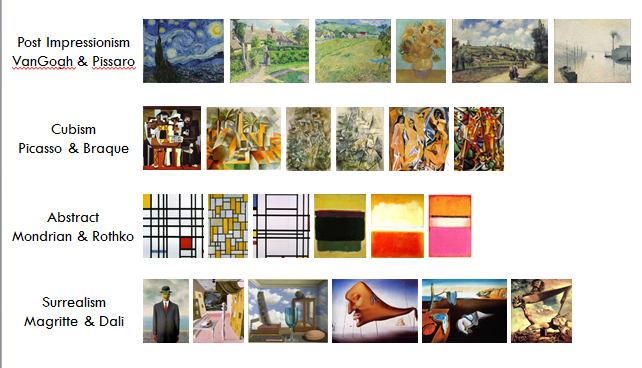
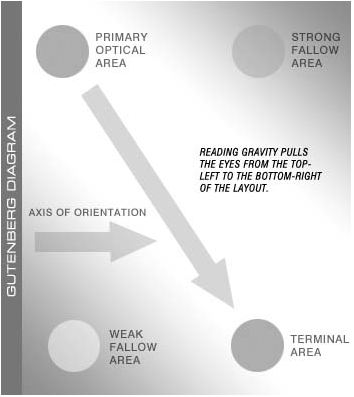


Figure 1. Paintings we considered

It may be clear to even an untrained eye (like ours) that these paintings are quite different, but that each genre and artist shares a similar look and feel. The present investigation will delve into these similarities and differences visually, and also provide some evidence that there is at least some consistency among all of these pieces that may aid in developing a system that can classify art more adeptly than present approaches.

In order to keep track of whether pixels are different by canvass location, we assigned an index value to each quadrant, using a labelling scheme like the Cartesian plane. It is a common design principle that people tend to view art in a quadrant based progression, starting in quadrant 2, and ending in 4, before turning their focus to interesting features and colors. [1]

Note, throughout the remainder of the paper, quadrants will be identified according to their Cartesian location.

# Data Processing & Methods

In order to import each painting, we authored a program in the *R* computer language that loads each jpg file, scales it to a standardized dimension, and then compresses it to a level that doesn’t overwhelm the processor. We then translate the RGB values the computer uses into a single value representing hue (our program is capable of translating to brightness, luma, and saturation, but hue was the focus for this investigation). Once this is done, we assign an index to each pixel and painting, so that we are aware throughout our investigation which quadrant each pixel corresponds to, and which genre each painting.

## Methods

We utilize several methods to extract the desired information from each painting and our collection as a whole.

* Density estimation: histograms of hues, Guassian Kernel Density estimation, Independent Component Analysis for projection pursuit density estimation (using the popular fastICA algorithm).
* Principal Components: computation of the principal components by painting and for all pixels
* Clustering: several methods attempted, but spectral clustering was the only one to give any reasonable results
* Manifold learning: kernel PCA contrasts interestingly with PCA, and locally linear embedding was used to see where the pixels map on higher dimensional planes

# Results

## Density Estimation

Results from density estimation are straightforward. You can see the results of several paintings in Figure 2, and for the genres we considered in Figure 3. Of note is the heavier-than-expected use of reds/oranges, and underuse of greens. Some genres are more evenly spread out (abstract) in terms of color palette than others (cubism).



Figure 2. Histograms of hue for 3 works from different artists

It is interesting to see in Figure 2 that the mountain in Starry Night is clearly visible in the histogram.



Figure 3. Histograms of hue for all 4 genres we considered

## Clustering

Since there were four genres of art considered, we hoped to categorize the works into four clusters. Several clustering techniques were used, including PAM, CLARA and k-means, but most methods failed to produce distinct clusters. Instead the majority of the works would be placed into one category with only one or two works filling out the other three clusters.

Spectral clustering with simple local PCA-based neighbors was the most effective clustering technique, and the results are described below.



Figure 4. Spectral clustering results

## Projection Pursuit Density Estimation

For PPDE, as it is commonly referred, we used fastICA() to run projection pursuit with deflation (essentially Independent Component Analysis (ICA) without a generative model for the data). The idea behind ICA is to find the least Gaussian directions in the given dataset, assuming it can be represented by an orthogonal matrix. In this method, the function

is maximized using a function for . We extract 4 components to demonstrate an interactive 3D plot, however that cannot be included here, so the corresponding 2-dimension component interactions are plotted in Figure 5.

We notice that there is definite grouping among quadrants 1 & 2 and 3 & 4. We also note that there are usually between 2 and 4 distinct bands or groups in each plot. This could correspond to different sections of canvass, though it seems there is overlap in the banding, particularly in comparing the first to Independent Components.

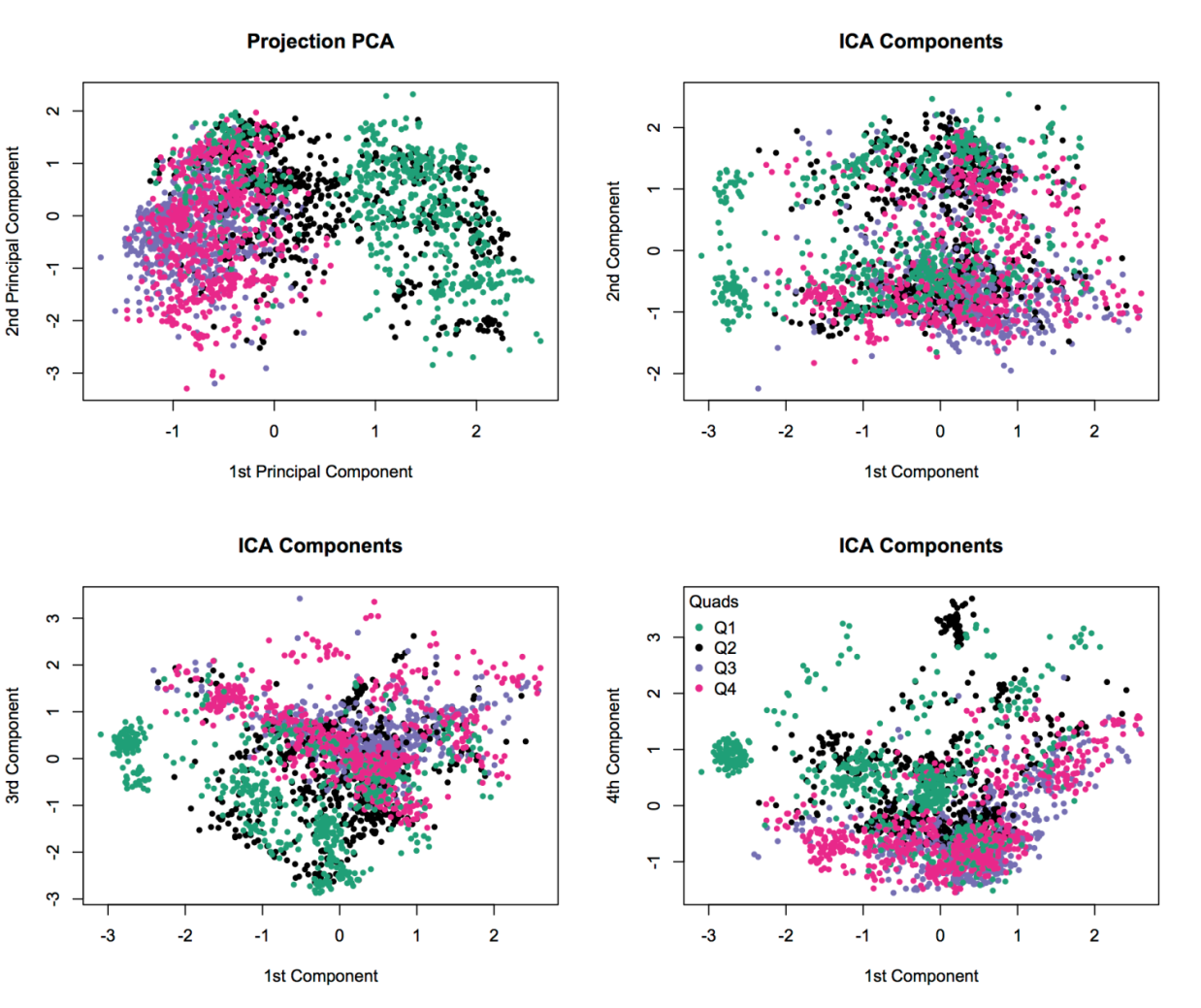


Figure 5. Plots of Projection PCA, along with plots of the projected ICA components. Note the characteristic banding, as well as grouping by quadrant.

## Principal Component Analysis

We ran a standard principal component analysis on the data, using both paintings and pixels as variables, respectively. This lets us compare the dual space of the two aspects about which we are curious. While it is immediately clear that there is a relationship of pixels having to do with whether they are in the upper or lower half of the canvass, it is less clear that there is significant grouping by genre. In an attempt to make this clearer, we have added Loess fit lines between paintings of the same genre.

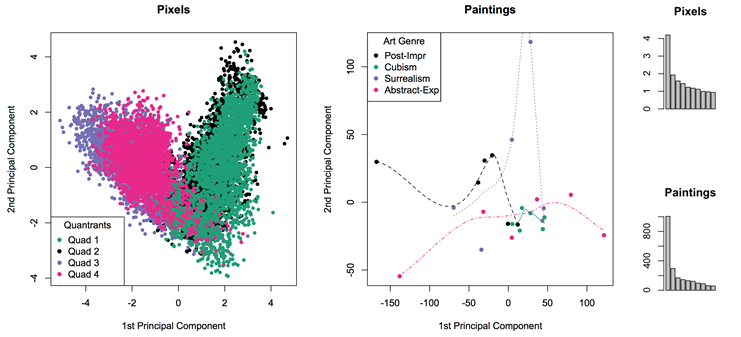


Figure 6. Plot of 1st two principal components for pixels, and for paintings, along with variance explained by each. Note the elbows in both at 2 components.

We note that while only a plurality of variance is explained by the first two components for pixels (~30%), and about 55% is explained by the first two for paintings.

## Locally Linear Embedding

We thought it would be an interesting exploration to observe what the pixels from the entire collection look like when passed through a locally linear embedding algorithm. To determine how many neighbors to include, we applied the method proposed by Kayo, 2006 [2], and arrived at an optimal value of 12 for . In calculating the inverse of the Gram matrix has to be calculated. The rank of , which is less than here, so regularization is performed using the trace of Gram matrix divided by. Now we run the algorithm and plot the first two dimensions of the resulting embedded data. We also note that the algorithm calculates the intrinsic dimensions of the data, what it believes the true dimensionality to be, and it is consistently around 8.

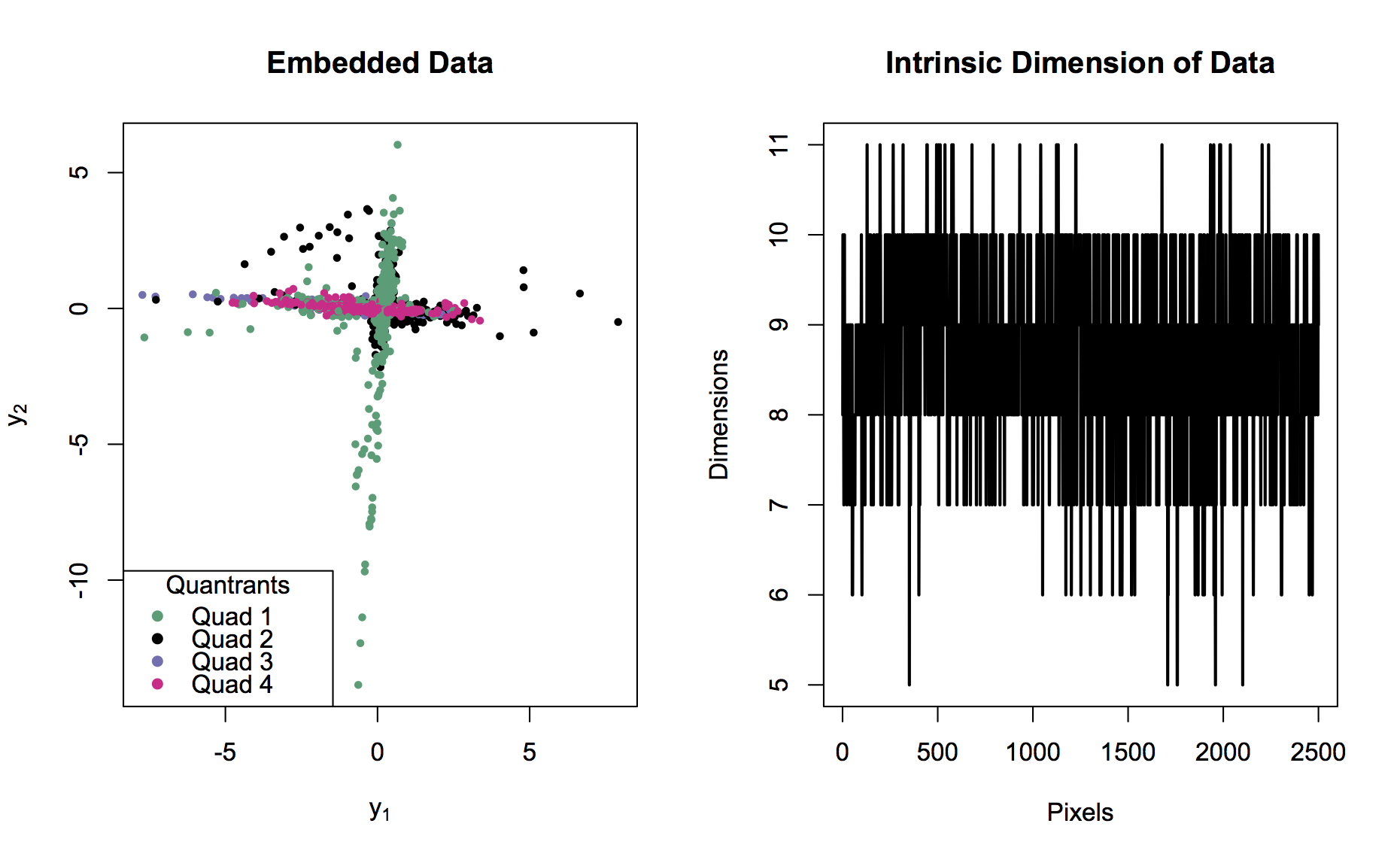


Figure 7. Embedded data resulting from LLE

## Kernel Principal Component Analysis

We also ran Kernel PCA, which is a fairly straightforward endeavor. We selected a radial basis kernel, and observed the resulting first two components from different values for sigma. The results are visually compelling, and further demonstrate the evidence of pixel groupings by quadrant. Figure 8 contains those results.

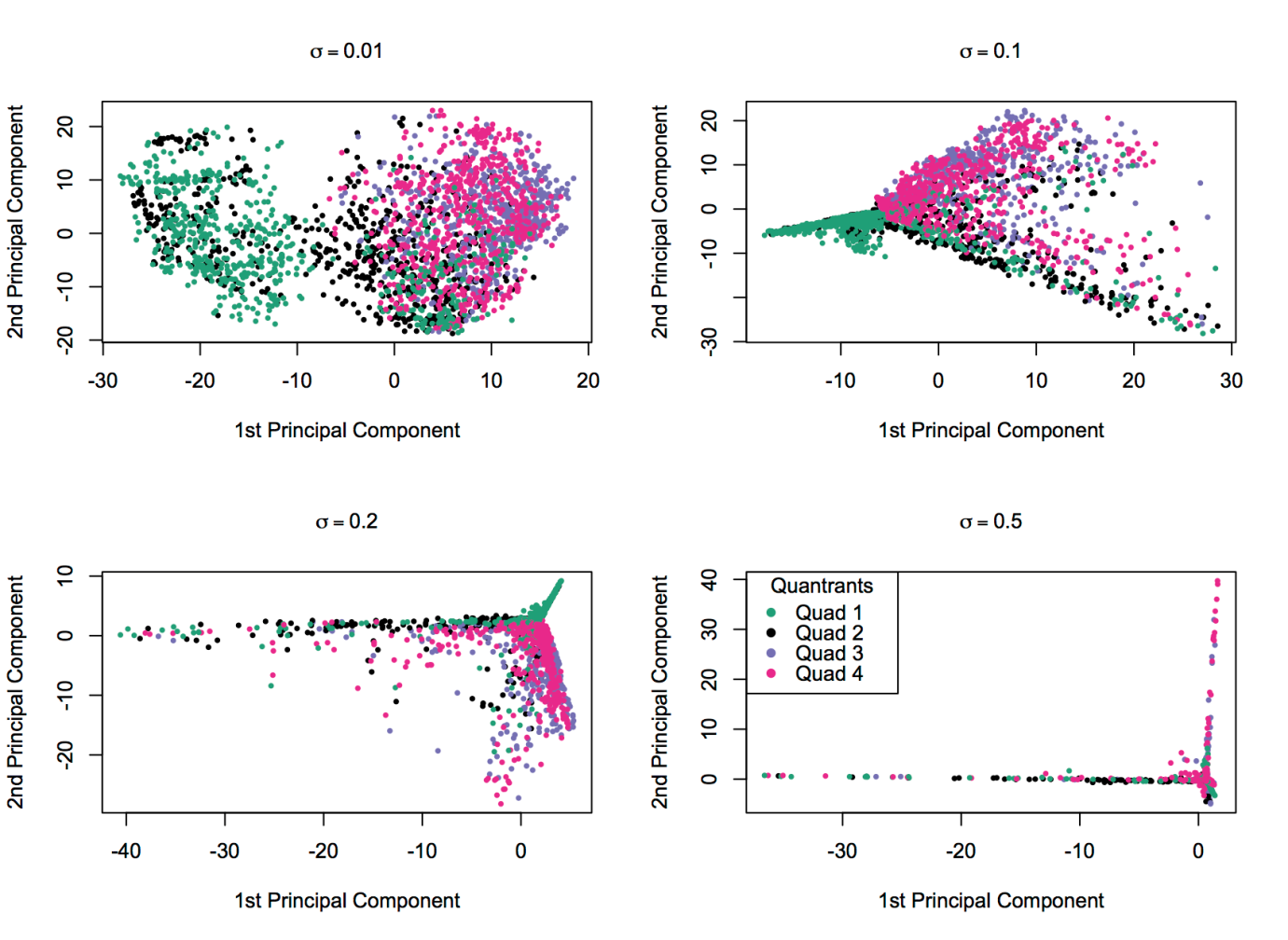


Figure 8. Plots of KPCA results for different values of sigma. Note the groupings by canvass region.

# References

|  |  |
| --- | --- |
| [1] | W. Lidwell, et. al., *Universial Principles of Design*. Rockport, 2003. |
| [2] | O. Kayo, *"Locally linear embedding algorithm - extensions and applications"*, Universitatis Ouluensis, Oulu, Finland, 2006 |
|  |  |