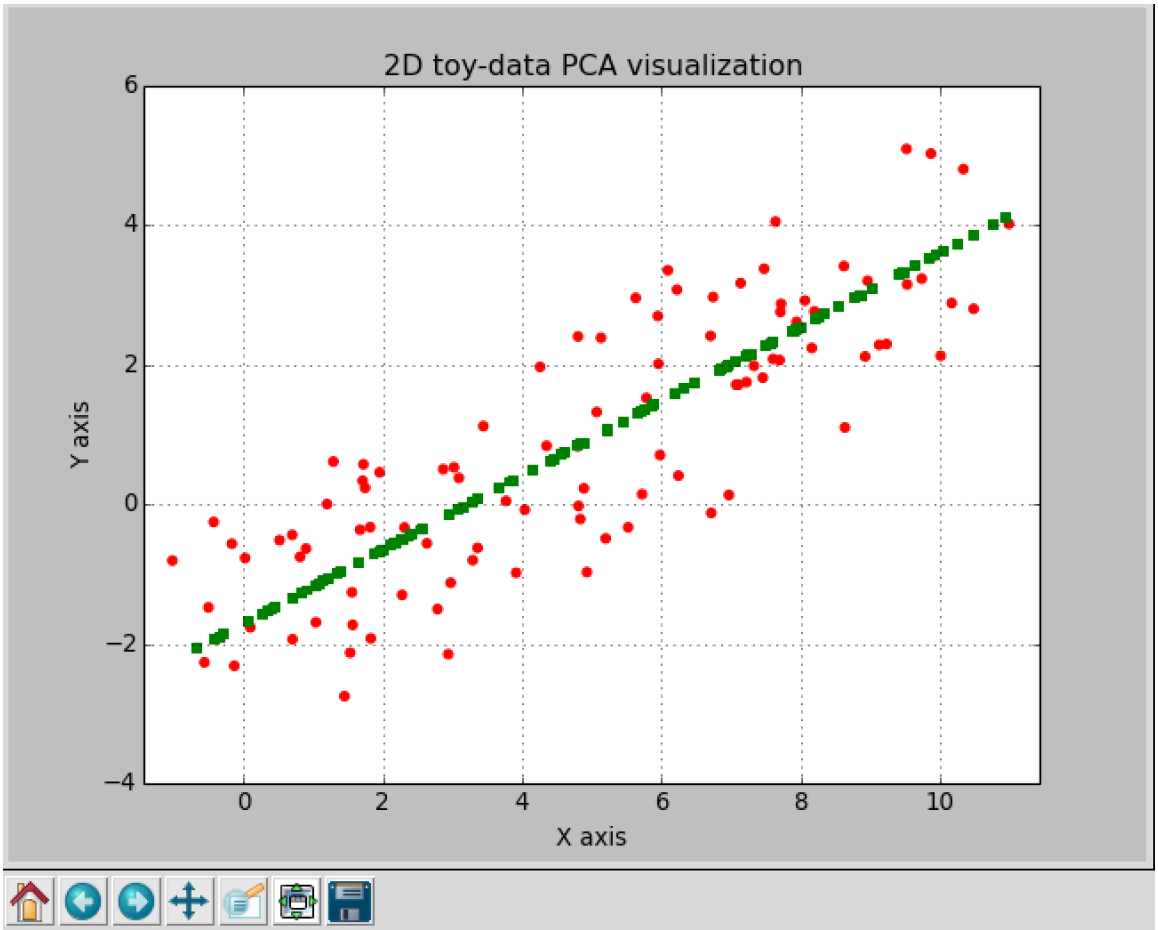
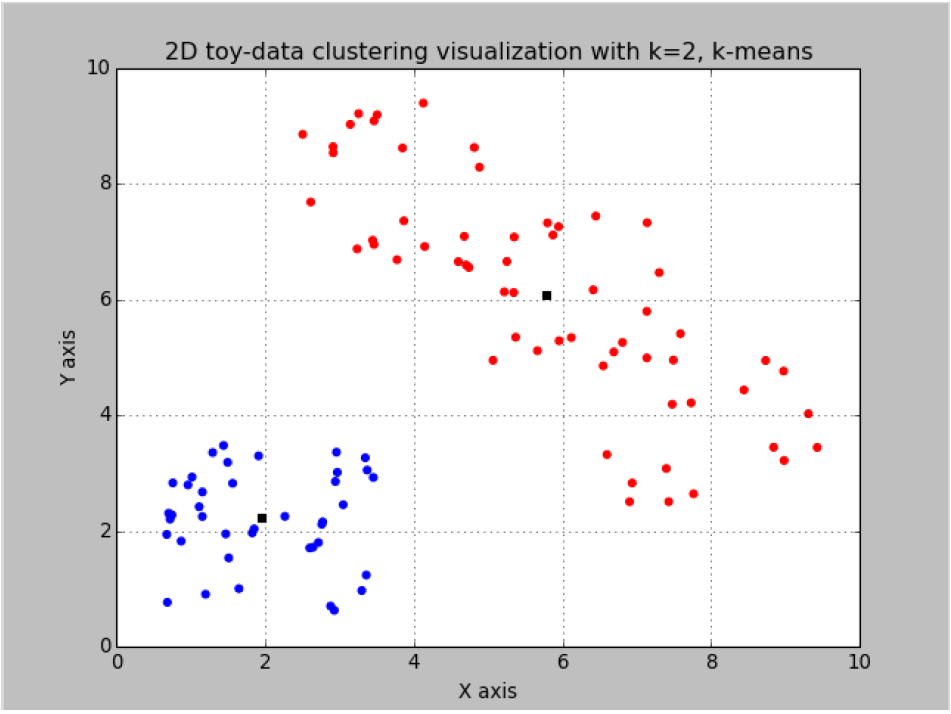
Aaron Thomas
Project 2
6.036

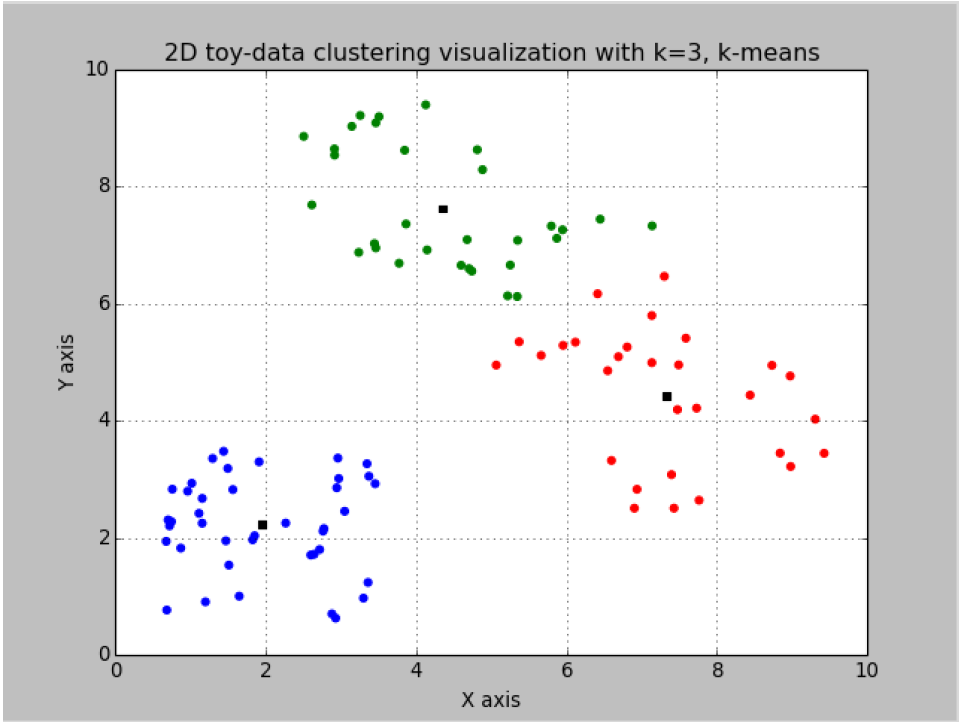
1)Plot of reconstructed points using PCA with m = 1.

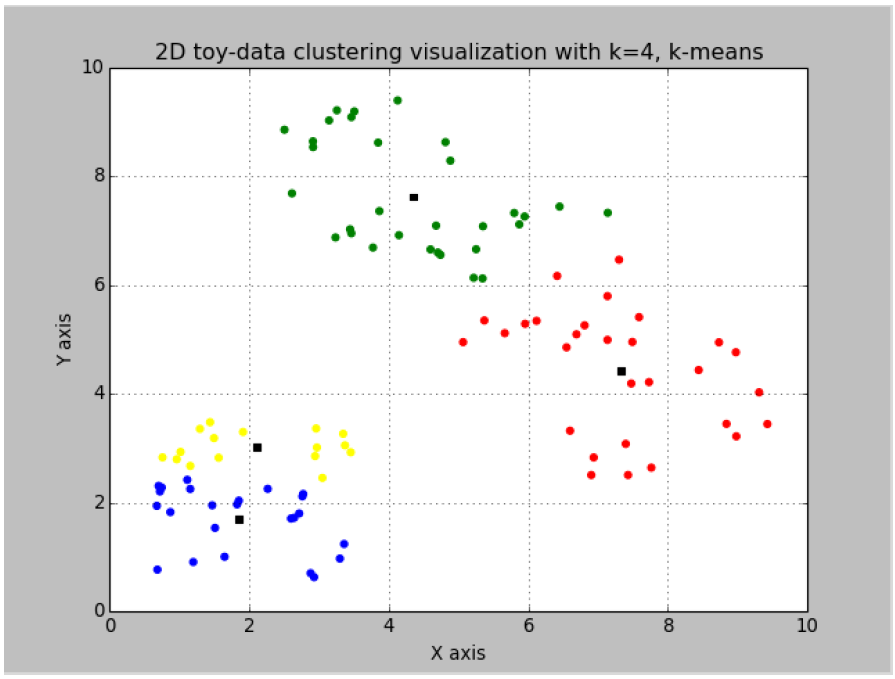


The reconstructed points makes sense because they are along the eigenvector, which runs through the center of the data, and they are capture the range of distances seen in the data, so the eigenvalue is correct. Additionally the single dimension captures the variance of the scatter because it has more points in regions of higher density and less in regions of lower density. Thus we are able to move from 2 dimensions using PCA to 1 dimension, as shown by the plot, and thus the reconstruction worked.

2) K-Means plots

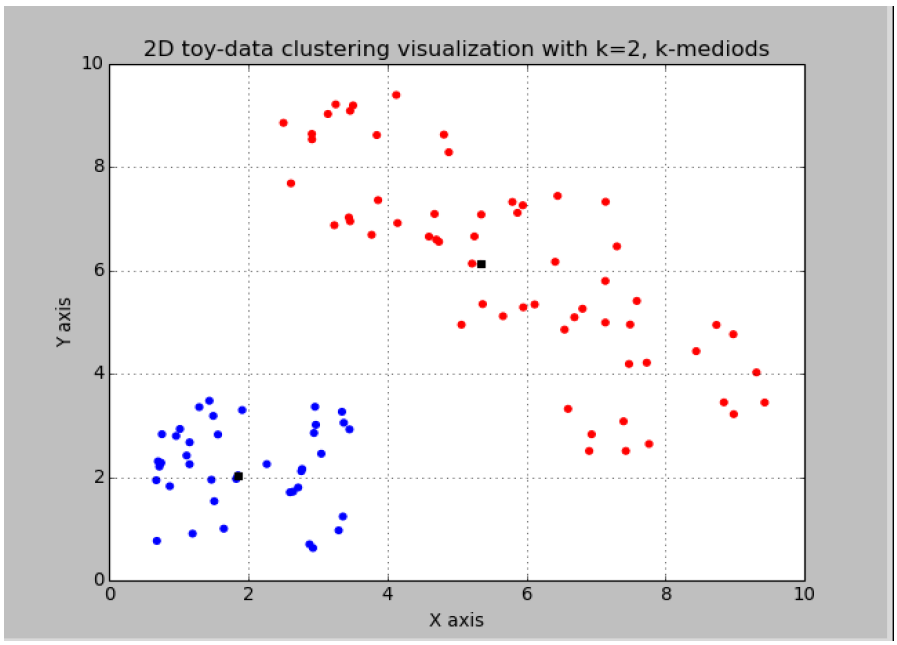


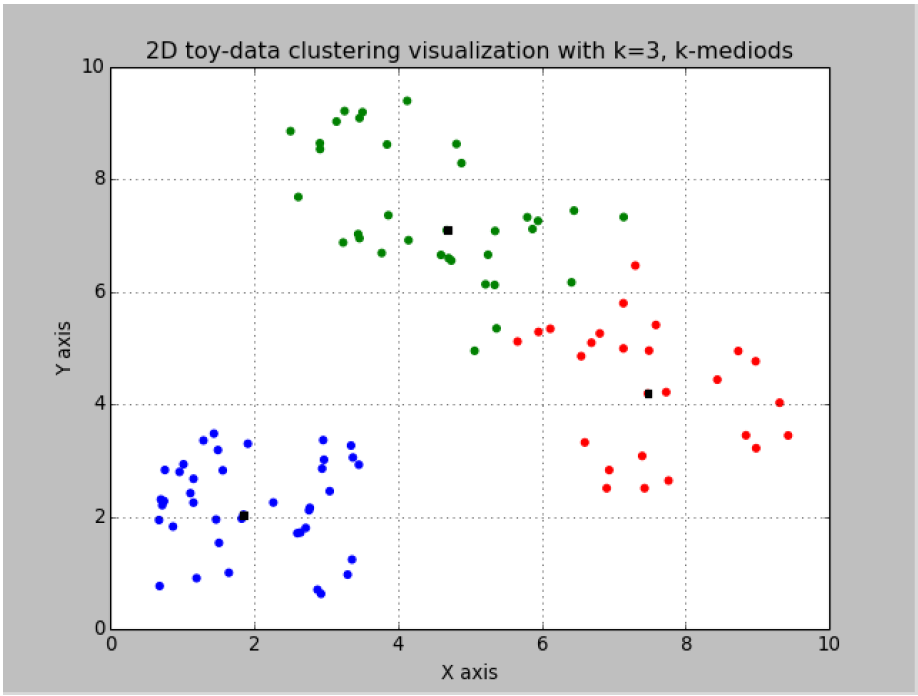


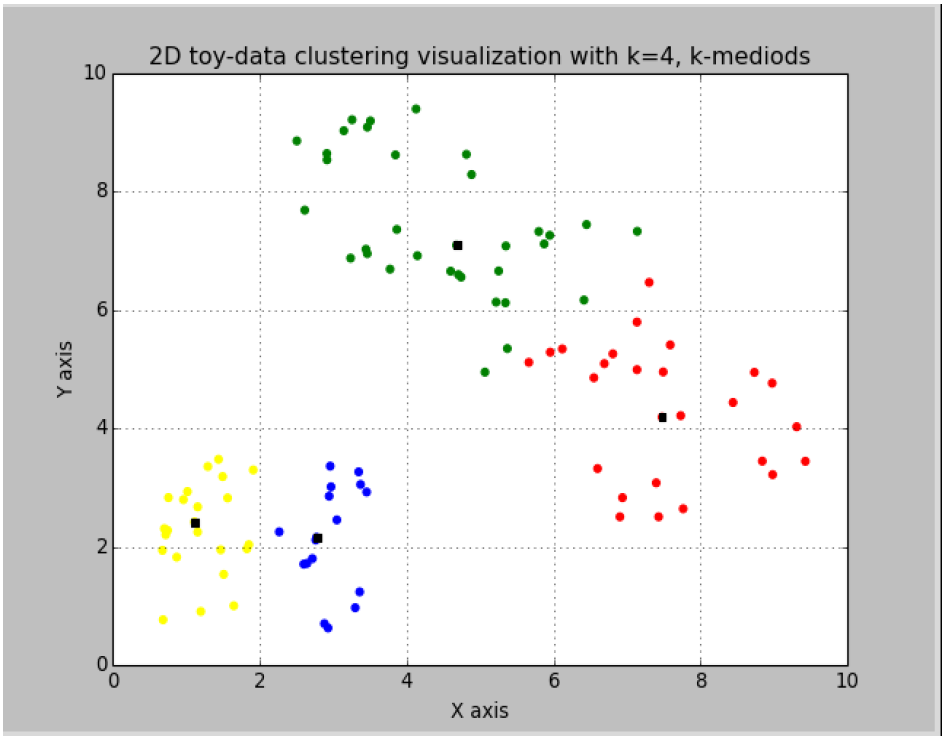


The initialization that we suggested, using the first k points, is generally a poor choice because k-means may converge to a local minimum and thus we won't be attaining the optimal clustering due to an ineffective initialization. However, if we perform random initialization we can compare our convergence each time, and hope to do this until we converge to the global minimum, which thus represents the optimal clustering.

K-medoids Plots







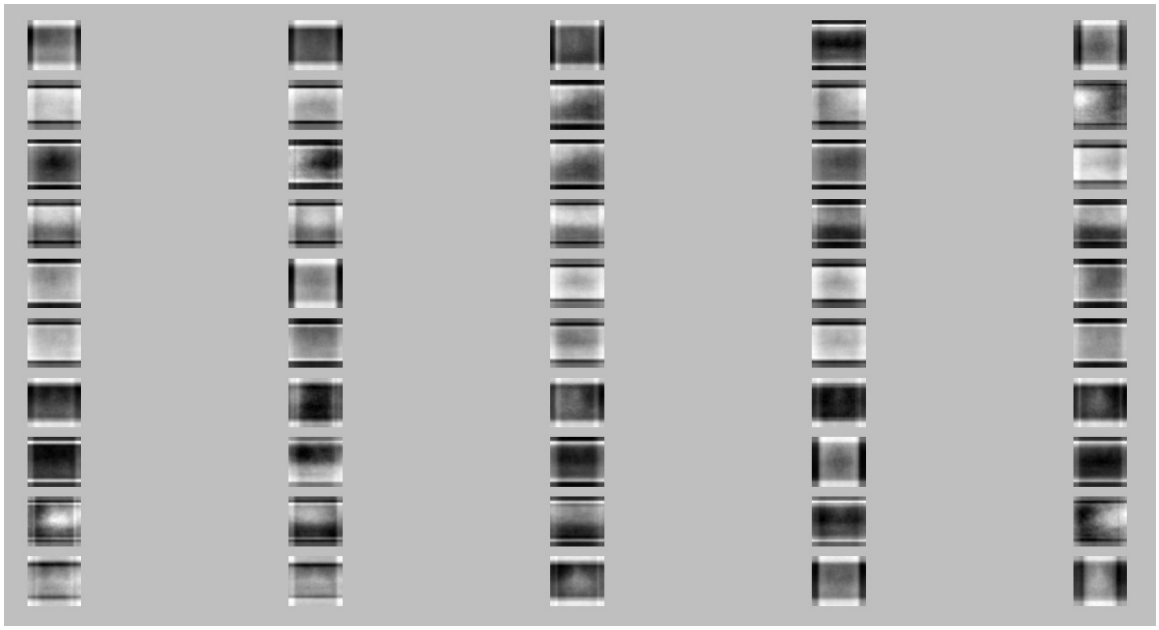
Using medoids, which correspond to actual data points can be useful because we can minimize any type of distance function that we would like to when choosing the best point. This allows us to not be constrained to the sum of Euclidean distances, as in k-means, thus allowing us more flexibility in determining of the optimality of our clustering schema. An example of the usefulness would be to minimize the effect of outliers and noise in our clustering.

3) Each artist works within a vague undefined range, whether it be range of colors, variety of objects, or lack there of, use of paint vs pencil. The separation is not perfectly clear between all these artists, but for the most part I think I could successfully cluster these images by painter if I was using the correct features. I think most of the confusions would become from paintings that I thought were Monet,Gaugin, or Cezanne because all of them use similar color ranges, don't focus solely on landscapes, buildings, or people and each of these artists paintings is very different from their others. I also think I would confuse the separation between J.M.W. Turner, Richard Wilson, and Paul Sandby because they paint landscapes with very similar color schemes and all their paintings are pretty similar to each others. Canaletto is distinct because he solely paints buildings and I think with feature detection that takes surrounding pixels into consideration we could accurately characterize buildings and city-scapes as different from landscapes and portraits. George Romney and John Robert Cozens are different from the rest because they are pencil drawings and each are both distinct from each other with regards to the value of pencil used. Peter Paul Rubens is different because the number of people in the picture is often very high so their will be many determined features because of many different colors in the picture. Rembrandt paints in a darker shade than the rest, and mostly portraits so I think that can be clustered together. Lastly, I said that Gaugin was similar to others, but he is also different, his colors seem more vibrant and saturated than that of the others, so we can use color saturation as a feature for classification.

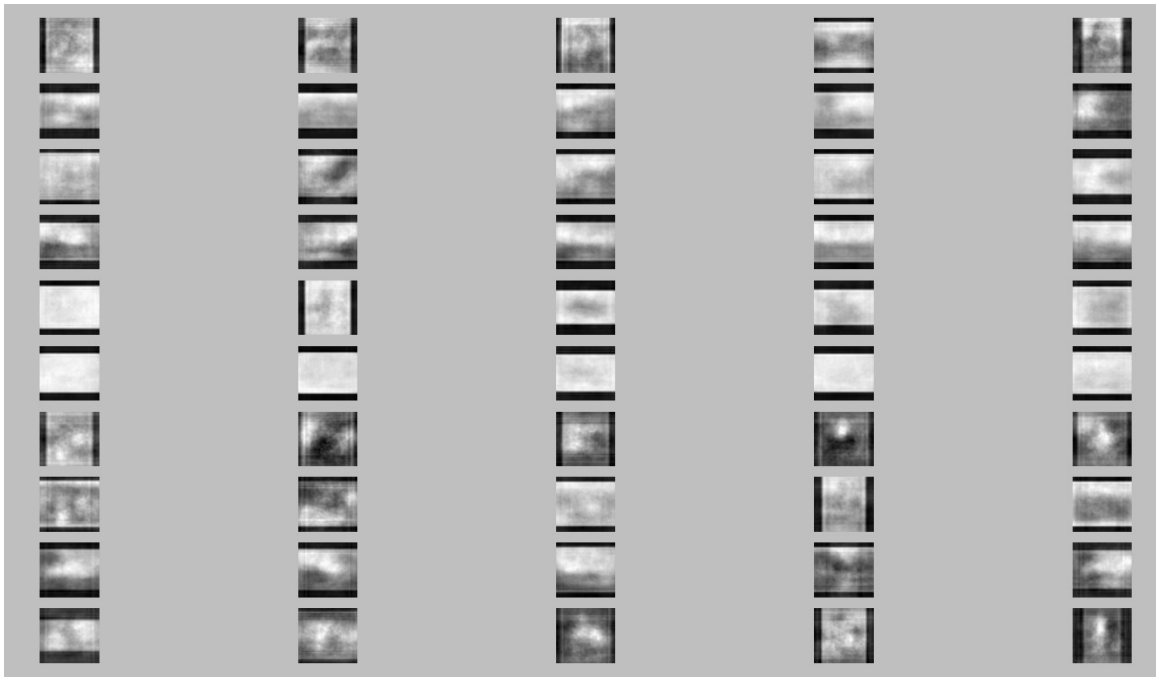
5) At m=10, most of the pictures are just partially shaded boxes, but the outlines of the box and the background colors of the art piece are well defined as would be expected due to the fitting and painting vs drawing nature. The rectangles are blurred as would be expected because the dimensionality of the pictures has been dramatically reduced. At m=10 I would be able to separate the George Romney and John Robert Cozens drawings because they are both set on white paper backgrounds whereas the others are colored, so this distinction is notable. Additionally if there are horizontal shade lines I assume that landscapes are being painted rather than portraits and a few painters only painted landscape scenes.

At m =30 I can start to see varying landscapes and they are discernable but I have uncertainty, and by m=100 I can definitely differentiate between painted landscapes like mountains, valleys, and streams. Also important is that I tested at m=415, the original dimension of the pictures and with m=600 and m=4000, so using more dimensions to represent the images and I don't see any difference between the images in the two, so with these pictures additional dimensions do not seem to make the images any sharper. This implies that if I project onto a higher number of dimensions I am not able to extract additional information from the matrix, but instead have the same data as if I had m=415, but the reconstructed matrix is a lot sparser.

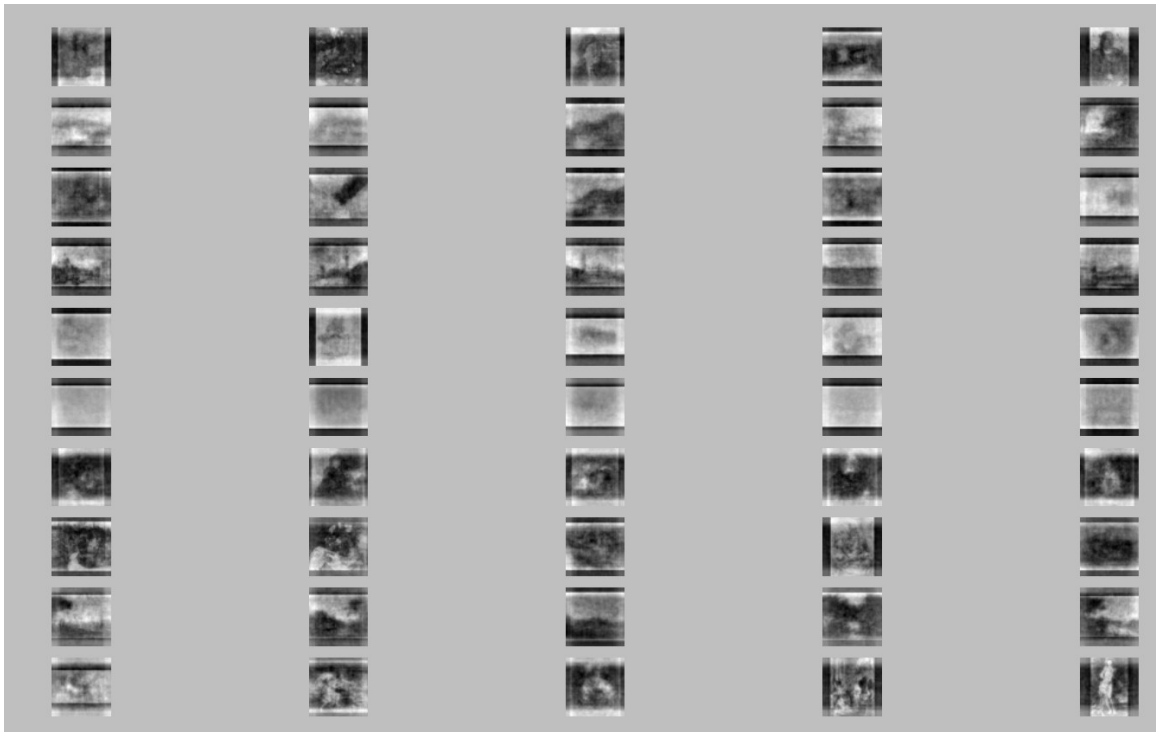
Reconstruction for m = 10



Reconstruction for m = 30



Reconstruction for m = 100



6) Results:

m is: 10 and k is: 2
Dunn Index: 0.459386968724
Cluster Purity: 0.248192771084

m is: 50 and k is: 2
Dunn Index: 0.401249285196
Cluster Purity: 0.24578313253

m is: 400 and k is: 2
Dunn Index: 0.398750951779
Cluster Purity: 0.24578313253

m is: 400 and k is: 5
Dunn Index: 0.288731951793
Cluster Purity: 0.368674698795

m is: 50 and k is: 5
Dunn Index: 0.353838586038
Cluster Purity: 0.412048192771

m is: 10 and k is: 5
Dunn Index: 0.309838306003
Cluster Purity: 0.378313253012

m is: 10 and k is: 10
Dunn Index: 0.269411899486
Cluster Purity: 0.486746987952

m is: 50 and k is: 10
Dunn Index: 0.254878146667
Cluster Purity: 0.493975903614

m is: 400 and k is: 10
Dunn Index: 0.271872489906
Cluster Purity: 0.489156626506

Dunn Index	k		
m	2	5	10
10	0.459	0.31	0.269
50	0.401	0.354	0.255
400	0.399	0.289	0.272

Cluster Purity	k		
m	2	5	10
10	0.248	0.378	0.487
50	0.246	0.412	0.494
400	0.246	0.369	0.489

Questions (in italics)
How do the clusters, and cluster purity vary with k?

As k increases we are able to more specifically cluster the images and when looking at the artworks a few are similar both most of the artists paint in a distinct way so more clusters improves the cluster quality. In k=2 I am unable to discern what the different clusters are, even though they contain similar art pieces, but on the whole I do not see any style that runs throughout all the paintings in the cluster. On the other hand in k=10 I see similarities in all the art pieces in each clustering, so from qualitatively looking clustering improves with increasing until k reaches the maximum usable number of clusters, which in this case is 10 because there are ten artists. As shown by the results cluster purity increases with k which makes sense because as we have more clusters we are better able to separate the pictures by artist and style and thus our clustering will be closer to that of the ground truth. Additionally artists produce pieces where each of their distributions should have a lower variance than the overall variance of all the art pieces and more clusters allows as to capture these local areas with lower variances, which should denote particular artists, thus increasing cluster purity.

How do the clusters, Dunn index, and cluster purity vary with m?

As m changes I do not see any qualitative improvements or reductions in choice of clusters. Instead the clusters seem to change a little bit when k is constant, but not in any way that is better or worse. Interestingly this implies that there is probably a minimum number of features for clustering that we need based on the inputted data, but once we have that number of features additional features are not necessary to improve the clustering. This implies that we can reduce or dimensions a lot using pca and when clustering we will still get pretty similar clusters as if we retained higher dimensionality. As for the dunn index, our results show that when holding k constant and varying m we don't really see a trend downwards or upwards in the Dunn index,

implying that density is independent of the dimensionality of our data.

If m is very small (e.g. 10), what happens when k is small/large? (think back to your experiments with PCA from part 2).

With small m, and small k, we have low dimensions and a small number of clusters, so we can capture the largest variances in the data with the small number of clusters, so the clusters will be spread out, but well defined. However, with large k, we are capturing the same overall variance, but with more clusters, so our clusters will have smaller variances, and thus be more overlapping than when k is small. With this also comes less well defined clusters because they can be next to each other or top of each other as in k=4 for k-means in part 2. The dunn index calculation above does not represent this, but I assume that that because of some normalizing factor as a function of the number of clusters.

7)
Results:

type is: raw
m is: 200 and k is: 10
Dunn Index: 0.297481625167
Cluster Purity: 0.496385542169

type is: raw
m is: 200 and k is: 5
Dunn Index: 0.243584043789
Cluster Purity: 0.431325301205

type is: raw
m is: 200 and k is: 2
Dunn Index: 0.399149784074
Cluster Purity: 0.24578313253

type is: GIST
m is: 200 and k is: 2
Dunn Index: 0.613957079966
Cluster Purity: 0.255421686747

type is: GIST
m is: 200 and k is: 5
Dunn Index: 0.464578532403
Cluster Purity: 0.366265060241

type is: GIST
m is: 200 and k is: 10
Dunn Index: 0.232367584566
Cluster Purity: 0.457831325301

type is: CNN
m is: 200 and k is: 10
Dunn Index: 0.269772370515

Cluster Purity: 0.739759036145

type is: CNN
m is: 200 and k is: 5
Dunn Index: 0.251204769854
Cluster Purity: 0.587951807229

type is: CNN
m is: 200 and k is: 2
Dunn Index: 0.303863007975
Cluster Purity: 0.269879518072

M=200, Dunn Index	k		
Feature Type	2	5	10
raw	0.399	0.244	0.297
GIST	0.614	0.465	0.232
CNN	0.304	0.251	0.27

M=200, Cluster Purity	k		
Feature Type	2	5	10
raw	0.246	0.431	0.496
GIST	0.255	0.366	0.458
CNN	0.27	0.588	0.74

Cluster Quality

Cluster quality is better for both CNN and GIST than the raw pixel values because feature tracking allows us a better opportunity to have each pixel have a relative effect on the pixels around them, which is useful in art because changes in color are directly chosen often implying an edge or change in object. Additionally, there are more than 5 natural clusters, and as k increases, the cluster quality improves as described in question 6.

Dunn Index

For the dunn index I do not see any trends in terms of a feature type producing higher or lower dunn index values, but two of the values from the GIST feature are far higher than any of the others but this high correlation between the dunn index and the gist feature is not highly correlated because the lack of enough data and that the third GIST feature data has a lower value than most of the others. This implies that cluster density is independent of feature type.

Cluster Purity

Cluster purity is definitely increasing in k, and I explained this in terms of variance of each artist compared to variance of all the art pieces and that each artist should produce a natural cluster above. Thus as k increases to the number of artists, the cluster purity should increase as shown here. With regards to feature type there is no strong correlation for one feature type producing greater purity, but for CNN two of the values are far higher than the others and the other is low, so if more data was collected, we could show whether or not the trend is actual. However, with large k, CNN seems to produce the highest cluster purity.

Best Feature Type

Across all k, CNN dominates all other feature types in terms of cluster purity, but at k=2 the differential is very small, so it is not a strong dominance, but for greater k, the dominance is strict and strong. With regards to dunn index and cluster quality I do not see a strictly dominant feature type.

8)
Results:

Semi-Supervised Classification	k			
% of dataset--labeled	10	30	50	70
5	0.4876	0.3631	0.3157	0.2482
15	0.6811	0.6402	0.6241	0.5944
25	0.7116	0.7221	0.6924	0.6675
50	0.7197	0.7261	0.7382	0.7518
75	0.6835	0.7647	0.7815	0.8
100	0.7181	0.7647	0.7888	0.7671

A classifier would have a very hard time learning a mapping from features to artist labels in this setting (why?).

If we had sparsely-labeled data it would be hard to match features to artists because of our lack of data, so with minimal data and many features we could struggle with determining which features are most important for classification and determining which classifier to use because many will be reasonable, but that will have little correlation to how good they will be on a testing set. The problem here is how can classify a system when we know very little about it, and the answer to this problem is to learn more about it, so it would be hard due to the lack of data and the inability to reason between the many plausible classifiers.

Look at the code inside the classifyUnlabeledData function: what classification decisions did we make in these cases?

In the classiffyUnlabeledData function if there are no votes present then a -1 is assigned to the cluster for those votes, and we then search for the mode of those cluster votes. After looking at the stats package that is used when we call .mode() we return a list in order of items with decreasing votes. In the case where we have no labeled data a list is still returned, but it is ordered in a seemingly random way and we set the first value of the returned list to be the classification of the cluster. In the case when two values are tied again one of the those two classifications will be first in the returned mode list and we set the classification to again be that first element. Thus if we have no votes or are unsure due to a high the returned classification is a random element in the set of possibilities, so any element if the votes are all zero and one of the two highest vote elements if there is a tie.

What happens as you increase k and decrease the amount of labeled data at the same time?

As we increase k and decrease the amount of labeled data at the same time, our accuracy decreases and this makes sense because with less labeled data and more clusters we have less

information to discern which cluster an image should be mapped to, and with more clusters we have an ever lesser chance of mapping to the correct cluster. Thus we are not only less sure of which cluster to assign to, but with numerous plausible probabilities and randomness of the k-means we are less likely to choose the correct cluster when the number of clusters increases. This is exemplified by the downward trend shown in the table above.

Should choice of k depend on the amount of labeled data?

As shown by the table there is no definite trend for changing k while holding the amount of labeled data constant, if the labeled data is small and k is increases then accuracy decreases, but instead if the amount of labeled data is large then accuracy generally increases. Thus choice of k should depend on the amount of labeled data available. This makes sense because if more data is labeled then I have more ways to classify and order the data and thus more reasonable clusters, but if I only have a small amount of labeled data then I am uncertain as to what cluster the data belongs to and far less certain of what cluster to map to when there are many clusters.