Unsupervised Learning and Dimensionality Reduction

Elijah Rockers
CS-7641

Datasets

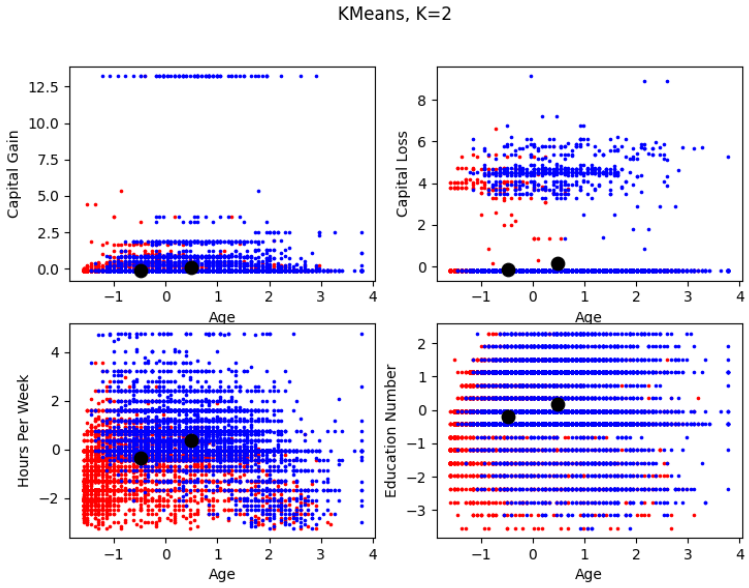
In the spirit of consistency, this submission uses the same 2 real world datasets used in the previous submissions. Briefly, they are

- census.csv – A binary classification problem (>$50k/year, or <=$50k/year) consisting of demographic data, where the attributes are mixed between real number-valued observations (capital gain, hour-per-week, etc), as well as categorical variables (nationality, occuptation, etc). In previous submissions I had used onehot-encoded representations of the categorical variables, however have found that it may be beneficial to do so manual reduction. For instance, instead of onehot encoding for every nationality, I have aggregated the observations into "United-States", and "Other-Country" which I propose would give the attributes more predictive power. In additional the "fnlwgt" continuous attribute is a feature aimed to allocate similar weights to people with similar demographic characteristic, and is thus removed. Similar to the "education" categorical variable, which is simply substituted with the already existing "education-num" continuous variable. Standardization is applied to normalize continuous values, so as to reduce skew in clustering methods.
- beans.csv – Similarly, the beans dataset is a multiclass classification problem, using physical characteristics of a dry bean measured from a camera to determine classification into 1 of 7 possible bean species. Standardization was applied to the continuous attributes, and a label encoder is used to transform the species named classes into integer values from 0 – 6.

Methods

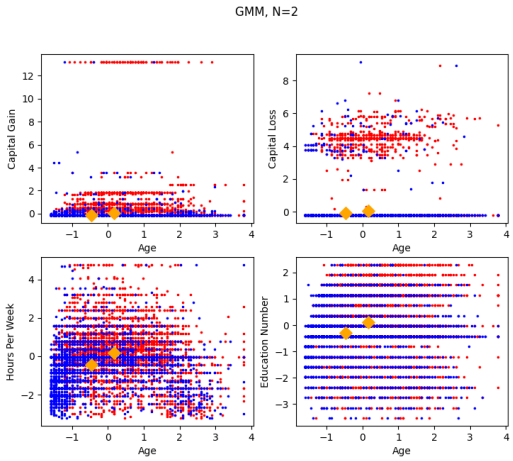
For both K-means and Expectation Maximization (GMM), the number of clusters/components was determined by the number of expected classes. The justification is that, in a problem like the census dataset, we might expect to be able to naturally cluster data into 1 of the 2 classes, and that these clusters might roughly correspond with the ground truth classes. Similarly for the beans dataset, a 'k' of 7 was chosen for similar reasons. One caveat is that in particular for the census dataset, it might have been interesting to try greater numbers of k, as there may be clustering modes of k=3 or k=4 that correspond to "hidden" a classification with the demographic data that is not necessarily reflected by the binary classes given. However, this was not explored for this submission.

Because the datasets I chose have a non-trivial number of features, it's not feasible to plot every correlation relationship between each pair of features in 2 dimensions. However, it may be reasonable to show a few plots of several possibly related variables.

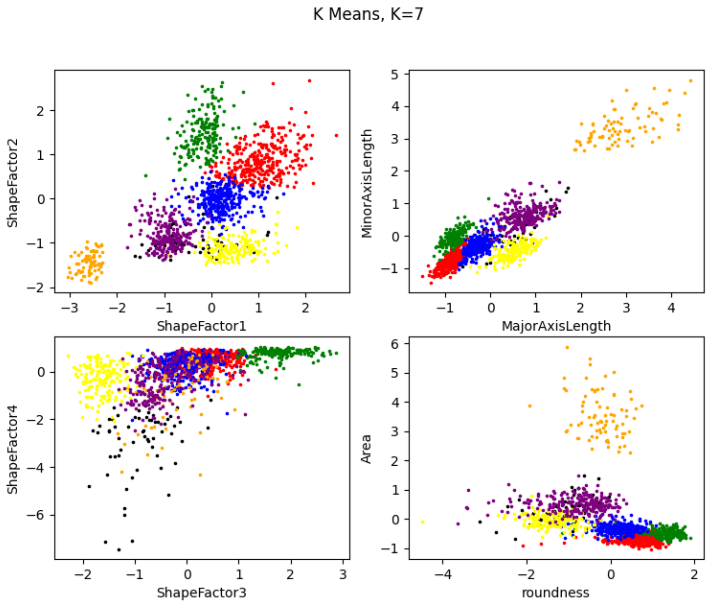


In the above plot, showing the census data, we might notice that even after standardization, “Capital Gain” and “Capital Loss” scales tended to overpower most other attributes (it’s possible of course that I miffed the standardization). The binary classes are represented by blue and red data points. It may come as no surprise that observations with very large capital gain tended to be clustered together, and similarly, those with very small capital loss were grouped into that same cluster. It’s probably a reasonable assumption that the “blue” cluster in this case might correspond to the >$50K/year class. Similarly, if we look at hours per week, younger people who worked fewer hours were less likely to be grouped into the “blue” cluster, and more likely to have been assigned the “red” cluster. This might well indicate the “red” cluster is approximating the group that makes <=$50k/year. A similar trend can be seen looking at Education Number, though it is not quite as pronounced. The expectation maximization clustering showed very similar results, though interestingly the cluster colors were reversed. This likely reflects the initial conditions of each cluster center being different from that of k-means. This trend is something we see throughout. And while the clustering labels may change between K-Means and GMM, the overall resulting clusters do not vary much, with some minor exceptions.

For reference, here are plots of the GMM clustering against the same data.



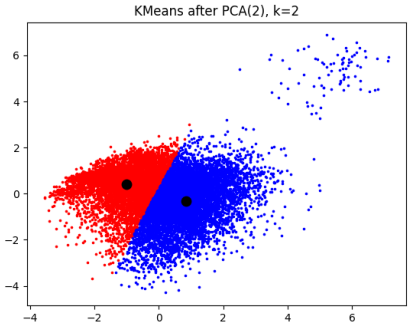
We will now look at the beans dataset initial clusterings.

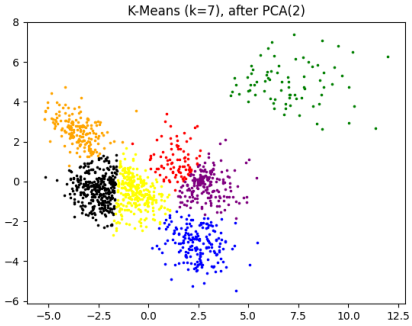


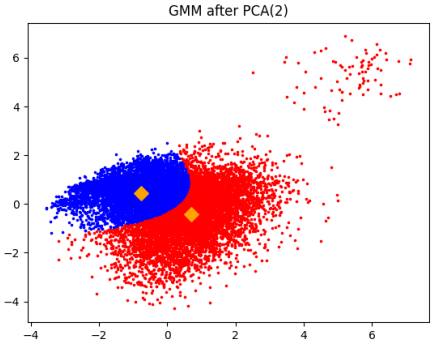
For the beans dataset shown above, we see several physical attributes mapped against each other, and the results of a k-means clustering of k=7. While the physical attributes (Axis length, Area, etc) do a remarkable job of segmenting the orange cluster away from the rest of the group, (likely, this species of bean is just generally larger and rounder than the other species), the ShapeFactor1 and ShapeFactor2 metrics seem to have an easier time separating out the other beans into more manageable clusters. While the dataset description does not explicitly outline what the ShapeFactor metrics are, it's likely that they are some derived factor of the more directly physical attributes measured by the camera, and may be dimensionally reduced features in their own right. Similarly, features such as "Area" and "Convex Area" were nearly linearly correlated, and could probably be reduced by simply dropping one or the other from the dataset.

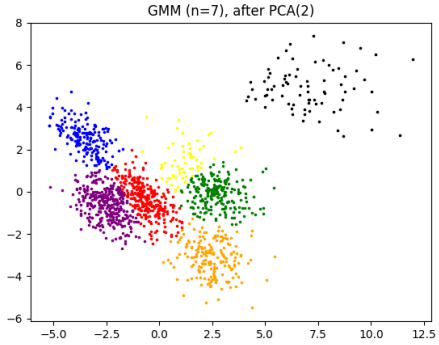
Principal Component Analysis

For principal component analysis, we chose to keep the first two principal components, thus making it easier to visualize for both the census (below, left) and beans (below, right) datasets.









Notice above, for both datasets, the clustering colors have switched between k-means and GMM, and the GMM clusters tend to be more feathered, with less "hard" linear boundaries.

Census PCA

For the census dataset, observing the original capital gain and capital loss plots, we see the plots are highly polarized, with the vast majority of observations appearing in a tight cluster on the low end of the capital gain plot, and many fewer samples appearing on the higher end. There is a similar reversed trend with capital loss. After PCA reduction, we can see a somewhat similar trend, with a very large blob of samples in the lower left end of the PCA plot, and a smaller smattering of samples in the upper right hand. This suggests that the PCA component reduction of the census data was relying in some part due to the capital gain or capital loss attribute.

Though, in fact, an analysis of the PCA eigenvectors shows that while capital-gain and capital-loss do play a role, it's age and hours-per-week, followed by education, (in that order) that seem to be the top three attributes contributing to the first principal component.

Similarly, for the second principal component, age, education, and capital-gain took the top three contributive spots.

Beans PCA

For the beans dataset, it's harder to make guesses from the original four data plots as to which feature space attributes we might expect to contribute more to the principal component decompositions. It's possible we just do not have plots of more appropriate feature pairs, as the features I decided to plot were a best estimate at which features I thought might be interesting to compare.

The first principal component seemed to take its top three contributions (among the most highly positive and negative contributions) from (in absolute value order from greatest to least) perimeter, major-axis-length, and shape-factor-2.

The second principal component similarly saw contributions from minor-axis-length, shape-factor-1, and aspect-ratio (which is defined as the relationship between the minor and major axis).

PCA: Beans vs Census

Interestingly, the contributions from the original feature space to the principal eigenvectors were much closer in the beans dataset. While I did only list the top the feature contributions for each component, there were many physical attributes that seemed to share a similar contribution to the top three. I also noticed that one particular feature "Extent" (The ratio of the pixels in the bounding box to the bean area), had nearly zero contribution to the 1st principal component, but had a notably larger involvement with the 2nd component. It may be that the direction of highest variance for Extent is orthogonal to those components that most contributed to the 1st principal component, and could actually make this attribute more valuable than we might initially think, in terms of predictive power.

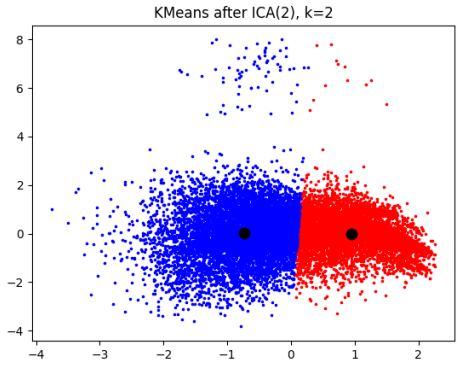
Additionally, a large number of census features had nearly zero contribution to the 1st and 2nd principal components, perhaps implying that these features might be dropped entirely. Among such features were an occupation of "never-worked", "armed-forces", and marital status of "married-af-spouse".

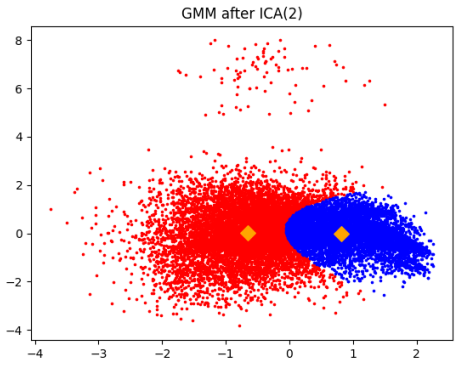
Independent Component Analysis

For ICA, I ran FastICA with the number of components equal to the number of features in the original feature space. In both cases, I used the most kurtotic ICA components as inputs for the clustering algorithms, associated with that problems known number of classifications. The thinking here is that if the data is generated by some noisy process with some number of unknown sources (e.g. 7 blind sources for the beans problem, 2 for census), then those new features might be able to facilitate an accurate clustering.

For census data (binary classification) this resulted in the two most highly kurtotic signals (kurtosis of 3408, and 2155) and for beans, the 7 most kurtotic signals (kurtosis of 155, 119, 68, 42, 37, 24 and 22). Interestingly, for the census dataset, those two highly kurtotic signals were by far the most kurtotic by a wide margin. The next most kurtotic being 800, 500, and so on.

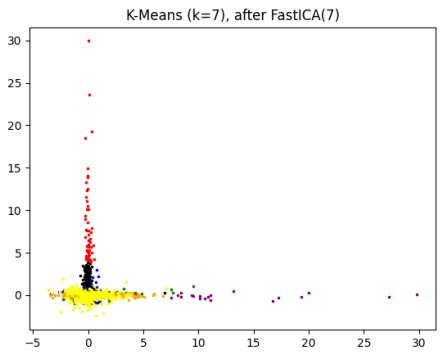
Census

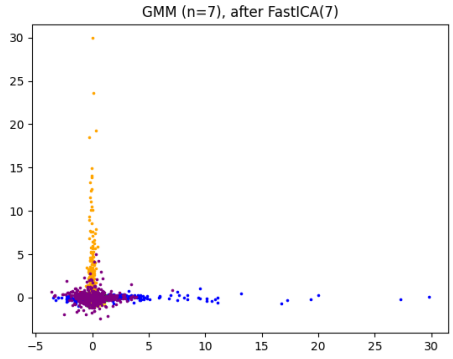




In this case, the only two remaining components are graphed orthogonally. While originally, I had speculated that K-Means and GMM would yield negligibly different clusters in most cases, in this case, one might reasonably expect that smaller cluster above the larger one to likely be part of the same cluster, in which case GMM might be considered slightly advantageous. However it's difficult to know if that larger cluster should be part of a whole cluster itself. Part of this could come down to the construction of the dataset itself, which has a very large number of categorical onehot encoded variables, such as occupation, marital status, etc. However despite my attemps to perform manual dimensionality reduction based on common sense (as opposed to an algorithm) there might still be room for improvement.

Beans



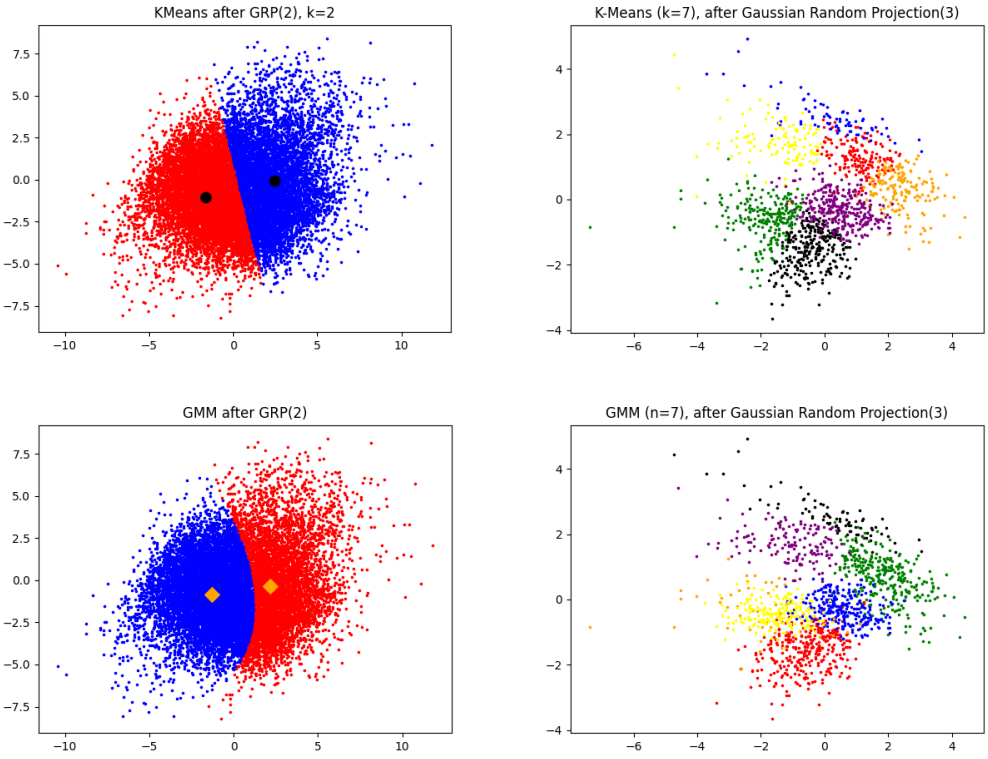


Perhaps here, it might have been more constructive to look at some of the other resulting ICA components. Here we are looking at the two most kurtotic components. While we can seem to see a bit of separate in orthogonal directions, most of the samples appear highly concentrated around the origin, likely making them more difficult to segment meaningfully. This could reflect that the feature space of the beans dataset, being highly related in terms of physical structure (axes, area, perimeter, diameter, etc), meaning it might be very difficult to separate these "signals" out into independent components.

Alternative Dimensionality Reduction

I won't go into great detail about Gaussian random projection, or Kernel-based PCA, which are the two remaining methods, save to look at some plots of their projections and clustering (census data on the left, beans data on the right).

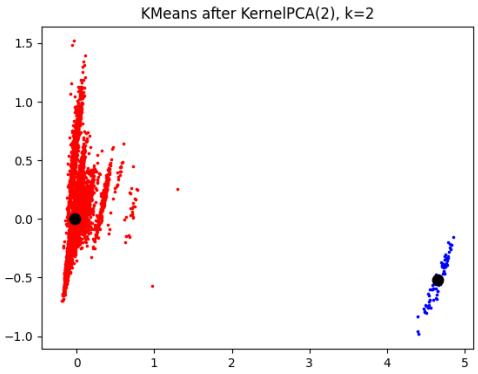
Gaussian Random Projections

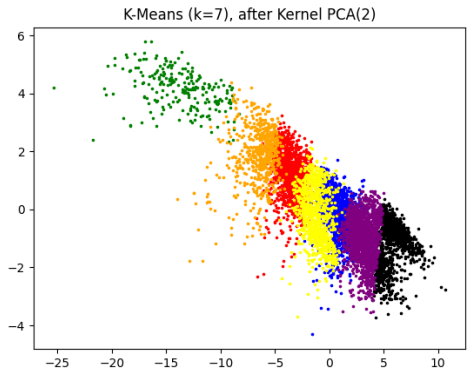


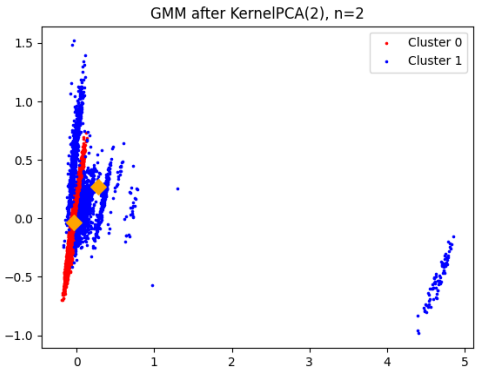
The Gaussian random projections (GRP) seem remarkably similar to the original PCA projections. I will note that I had at one point generated GRP projections for the census data that resulted in nearly a linear straight line, with two clusters almost on top of each other. I failed to save the image, but my intuition tells me that was probably not a valuable projection anyway.

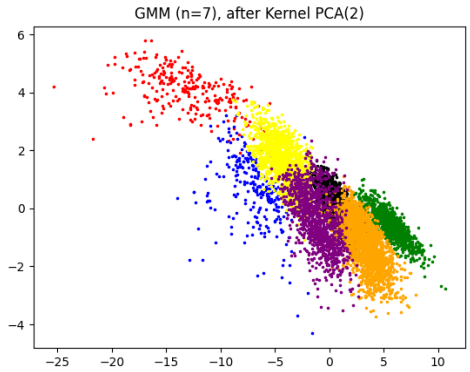
Kernel PCA

Kernel PCA is supposed to be a method for dealing with data that is not linearly separable, similar to the "kernel trick" explored in the supervised methods section of this course. In this case a polynomial kernel was used of degree 2.









Conclusion

Clearly the clusters generated will be determined by not only the clustering algorithms used, but also by any type of dimensionality reduction performed beforehand. In fact in some cases, the original feature space seemed to make more sense. One issue with dimensionality reduction is that it while it can make clustering easier in some cases, it becomes less clear what is represented by your data, as the reduced feature spaces essentially become arbitrary mixtures of the original feature spaces.

Perhaps one strategy to mitigate this could be to include those original features you believe have strong predictive power on their own, in addition to performing a PCA-like feature space reduction on other attributes.

One issue I ran into was the stochasticity of the clustering algorithms, in the sense that depending on your initial conditions, you could have different cluster labelings for different presumed classifications or ground truth labeling. Especially with a multiclass classification problem, untangling which clusters are “presumed” to represent which original labels proved to be tricky, and thus made it difficult for me to quantify the performance of the clustering methods with respect to the ground truth labels. Doubtless there is some indexing trick I am missing, but especially when you would suspect some error between ground truth and cluster results, it’s not immediately clear how to untangle which clusters might best represent which classes.