Peter Buonaiuto
Data Mining HW 1

1. Data Analysis

PMF and CDF plots

I determine the PMF as $$P(X = x) = \frac{0.85^{x-1}}{3.186625}$$

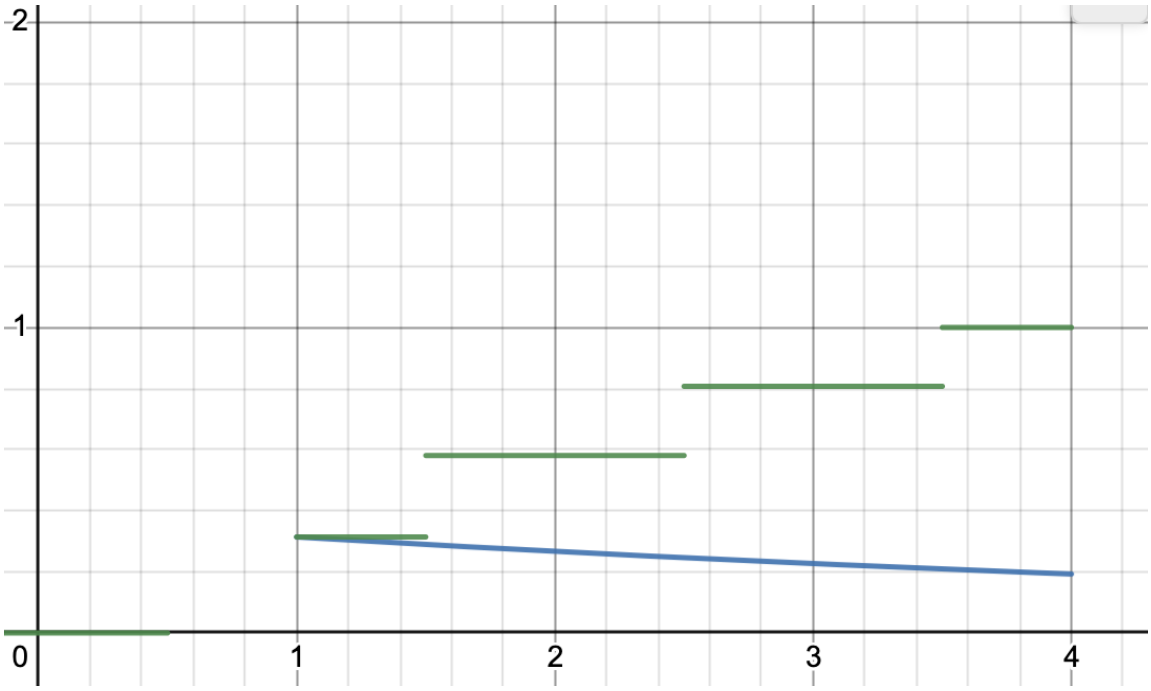
Which is the number of students in year X divided by the sum of all expected students in Y1->Y4:

$$\frac{1000 \cdot 0.85^{x-1}}{1000 \cdot \sum_{x=1}^{4} 0.85^{x-1}} == \frac{0.85^{x-1}}{\sum_{x=1}^{4} 0.85^{x-1}} == \frac{0.85^{x-1}}{3.186625}$$

Now the CDF will be P (X ≤ x), so I can treat it as $$F(x) = \sum_{X=1}^{x} \frac{0.85^{X-1}}{3.186625}$$.

I calculated the PMF over my domain to verify that they sum to 1 to check correctness.
P(X = 1) = 1 / 3.186625 ≈ 0.313989
P(X = 2) = 0.85 / 3.186625 ≈ 0.266603
P(X = 3) = 0.7225 / 3.186625 ≈ 0.226764
P(X = 4) = 0.614125 / 3.186625 ≈ 0.192444

And now, plotting both:



Expected value and variance

I calculate the expected value as a weighted average of the PMF data. I will sum over each data point for the PMF and multiply the coordinate point as follows:

$$E(X)\sum_{x\in X} x \cdot P(X=x)$$

using the data points from part (a):

$E(X) = (1 \cdot 0.314..) + (2 \cdot 0.267..) + (3 \cdot 0.227..) + (4 \cdot 0.192..) \; = \; 2.297263$

For the **variance**, I find the sum of the squares of the centered data times its probability:

$$V(x) = \sum_{x\in X} \left[(x-\mu)^2 \cdot P(X=x) \right]$$

$V(x) = \left(-1.297^2 \cdot 0.313\right) + \left(-0.297^2 \cdot 0.267\right) + \left(0.703^2 \cdot 0.227\right) + \left(1.703^2 \cdot 0.192\right)$

V(x) = 1.220

Expected value with respect to α is the sum of each x times its PMF:

$$E(X,\alpha) = \sum_{x\in X} x \cdot \frac{\left(1-\frac{\alpha}{100}\right)^{x-1}}{\sum_{y\in X}\left(1-\frac{\alpha}{100}\right)^{y-1}}$$

Merged university and robust degree of data

To model the expected value of the merged university, I first define a new PMF as a piecewise function since the number of students between the two universities have no relation:

$$\begin{cases} \frac{0.85^{x-1}}{3.376625} & 0 < x < 5 \\ \frac{2x^2-25x+82}{337.6625} & 5 < x < 9 \end{cases}$$ **0, otherwise**

Note the second PMF was obtained through a system of linear equations given the provided data points of the graduate school.

We can now calculate the new expected value as P(X=1) + 2*P(X=2) … + 8*P(X=8)
P(X=1) = 0.2962
P(X=2) = 0.2517
P(X=3) = 0.2140
P(X=4) = 0.1819
P(X=5) = 0.0000
P(X=6) = 0.0119
P(X=7) = 0.0148
P(X=8) = 0.0296

The sum of the new PMF is 1, as expected.

E(X) = .2962 + 2(.2517) + 3(.214) + 4(.1819) + 6(.0119) + 7(.0148) + 8(.0296)
E(X) = 2.581

When rounding, the mean is **sensitive** to these new data points as the expected year of a student for **University A was 2** whereas the expected year for **University C is 3**. However, if we only consider the integer part (floor) of the data, the mean would be **2** in both cases, being **stable**. The data only changed by about 0.20, so the result is very stable until we round.

Alternative types of statistical data:
Variance:
For **College A,** this is **1.22** as calculated above.
For **College C,** this is **2.53** calculated the same way as below:

$V(X) = -1.58^2(0.296) + -0.581^2(0.252) + 0.419^2(0.214) + 1.419^2(0.182) + 2.419^2(0) + 3.419^2(0.012) + 4.419^2(0.015) + 5.419^2(0.030)$

With the introduction of the outliers provided in the merge, ***variance is unstable*** *as it more than doubled. This is due to the average spread between data points increasing through the influence of high outliers.*

Median:
The median is the smallest m such that **P(X <= m) >= 0.5** *and* **P(X >= m) >= 0.5**

I calculated it by summing the PDF values together for each x, and stopping when this sum meets or exceeds 0.5.
For **College A**, this is **2**. P(X<=2) = 0.581, and P(X>=2) = 0.686
For **College C**, this is **2**. P(X<=2) = 0.548, and P(X>=2) = 0.704

The ***median is stable*** *through the introduction of the outliers. Since a very large majority of the data lies in the original sample, adding a small amount of data won't skew the midpoint, regardless of the magnitude of such data.*

Mode:
For both colleges, the mode is **1** as they yield the highest value of *f(x)* (their PMF) at 1.

*The **mode is stable** (insensitive) to the new data. This depends on the data added. In this case, since the new data didn't compete with regard to quantity, it was unable to dominate the most occuring data point of x = 1. In a case where a large graduate school where there were more students in a particular year than in a particular year of College A were to have merged instead, the mode would be sensitive in that case. However, since a small school merged, the data was not enough to offset the original.*

Quartiles: = F inverse at Qx (or summing pmf until the sum exceeds (or equals) p)

For **College A, Q3** falls between 2 and 3. $F^{-1}(3) = 0.76$
For **College C, Q3** falls between 2 and 3 (closer to 2 than college A's Q3) $F^{-1}(3) = 0.81$

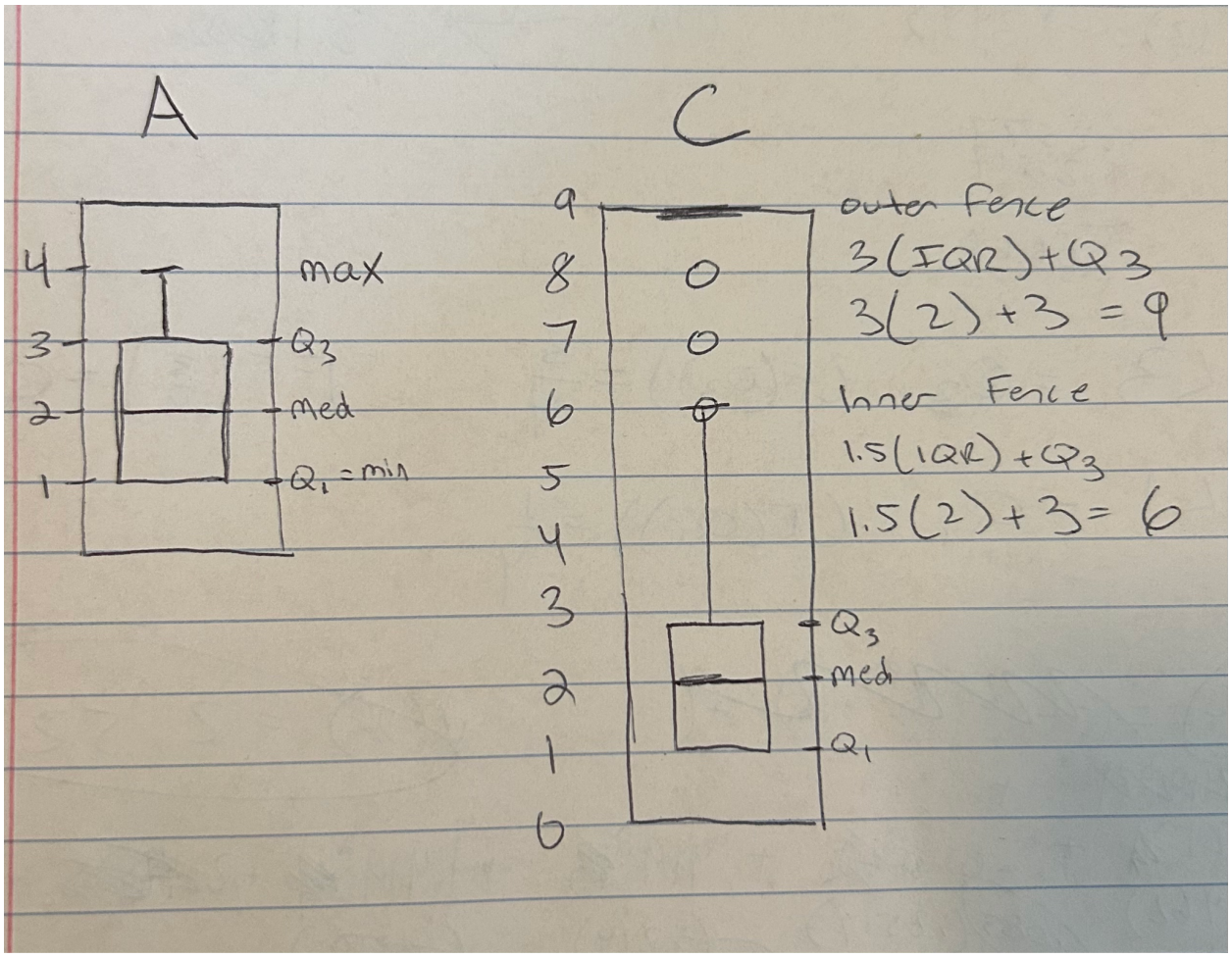
For **College A, Q1** is 1. $F^{-1}(1) = 0.31$
For **College C, Q1** is 1 $F^{-1}(1) = 0.30$

Both college's **IQR** falls between 1 and 3. However, **A's IQR will be greater**. because we guarantee to take a higher number - a lower number when compared to C.

*If quartile values must be in the data set, **Q3 = 3**, **IQR = 2**. Must round up because only then can we say 100p% of data is below Qx. If rounding down, 100p% of data will NOT be below this value, since we iterated to the next x in X when summing pdf while attempting to reach p means that the lower value will certainly not be valid, or else we would've stopped and this would've been the upper bound.*

*Although imposing a slight change in the quartile values, the introduction of College B does not significantly alter the data in these regards. **Quartiles are stable** to the new data in this case. However, as with other data types, this is a case by case basis as other large universities could merge with College A, thus dwarfing the original data. In this case, however, the data remains relatively similar to the original.*

Box plots for two universities



Note for A, there are no outliers and the whisker represents the maximum value.
For C, the graduate students are outliers, visualized at 6,7,8. The whisker here represents the inner fence, depicting all values of 6,7 and 8 as suspected outliers. There are no certain outliers (shaded dots), which would be at or above 9 if present.

Quartile 1 becomes raised to 1 in both cases since 1 is the minimum value, and 30% and 31% of data in colleges C and A lie at this point, respectively.

The median is 2 in both cases, and Quartile 3 is slightly less than 3 in both colleges, however in college C, Q3 is slightly lower than the value in College A.

2. **Irreducible Data**

Discuss when PCA would fail

PCA fails when projecting data onto a dimension which significantly decreases the projected variance. Although maximization techniques locate the best possible vector u_1 that follows the direction of general data flow, there are times when too much variance is lost and data points lose their integrity. Specifically, if the data does not correlate well between each other, PCA will usually not help. Finally, if data fails to follow a linear trend, PCA wouldn't work well either.

How do we quantify that it fails?

By dividing the projected variance by the original variance, we see the percentage of variance preserved. Alternatively, this also represents the amount of content that was lost. A low ratio here determines a less successful PCA.

Provide an example that fails:

<0, -10>, <0,10>, <10,0>, <-10,0>: seemingly uncorrelated, non linear data: In this data, when reduced to one dimension, values and patterns will be lost because we project to a 1d space. Any data points on the axis orthogonal to the direction of the PCA vector will be combined into one data point, and their deviation from this data point will be removed. In cases like this with data tending toward two varies axes, one axis will be eliminated which removes sensitive information regarding data and pattern.

3. **Dimensionality Reduction**

b. Differences in covariance calculations

cov(1) reports an execution time of 0.40ms on my machine.
cov(2) reports an execution time of 0.04ms on my machine.
cov(3) reports an execution time of 4.00ms on my machine.

This suggests that the time complexities are ordered as follows: T(c2) < T(c1) < T(c3).
This means that cov(2) runs the fastest, having the simplest complexity, and c3 runs the slowest. More specifically, the functions of higher time complexity slow down proportionally to an increase in input size (Higher dimensional data).

Since data dimension is 10, and times increase by multiples of 10, it is clear that the size of the data directly affects the execution time of the functions, based on their complexity.

d. Determining r based on alpha

2 principle components are the minimum to ensure 90% retention of variance. Of course this means r > 1, any dimensionality higher than 1 will retain at least 90% of variance.
This was calculated in my **find_r(Y,a)** function. It requires **Y** (the eigen values of the coviariance matrix), as well as **a**, the desired ratio of variance retention.

The function will attempt to reduce the dimensionality of the matrix to the lowest possible dimension, r, which encapsulates a ratio of alpha variance or more. If it is not possible to lower

the dimension of the data, d, by any amount then a message will be displayed that no reduction is possible for the given parameters.

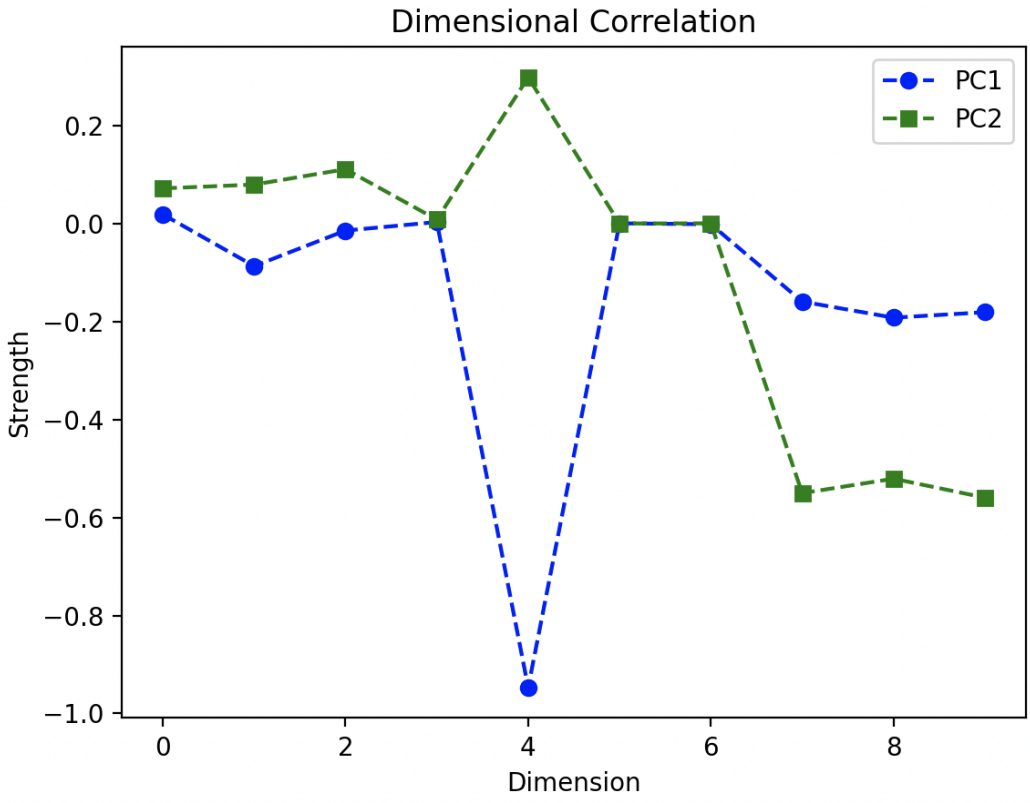
1. The function works by iterating from 1 to d-1 (because d is the trivial case where 100% of variance will remain, but no reduction has occurred so the dataset is unchanged)
2. For each of these values, r, I then calculate the variance retention ratio.
3. I sum over each lamda in Y to get the total variance of the original data.
4. I then sum the first r variances to get the total variance of the adjusted sample.
5. I divide the adjusted variance by the original
6. Still in the for loop, I test if this ratio is at least alpha.
7. If so, I return out of the function with the current value of r and it's retention ratio

The function calculates from the lowest dimension upward, ensuring that we halt at the first acceptable value of r to avoid unnecessary calculations and so that we return the least number of dimensions as possible.

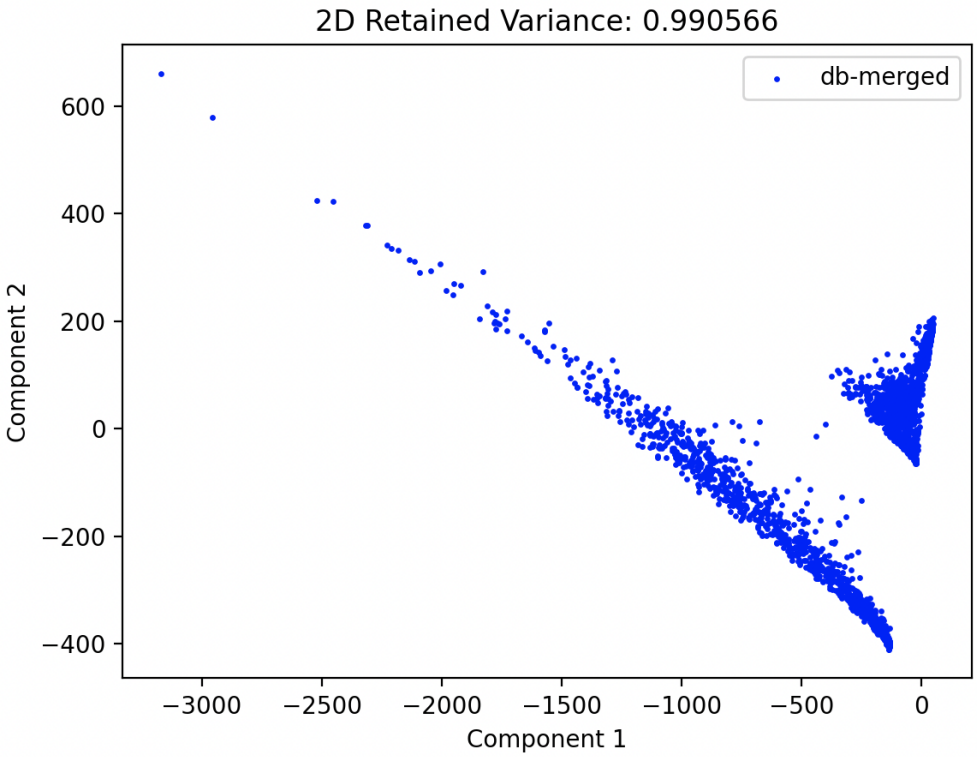
If all d-1 dimensions are attempted and none have an acceptable retention, the function returns -1 as no r < d exists given the dataset that will respect the provided value of alpha.

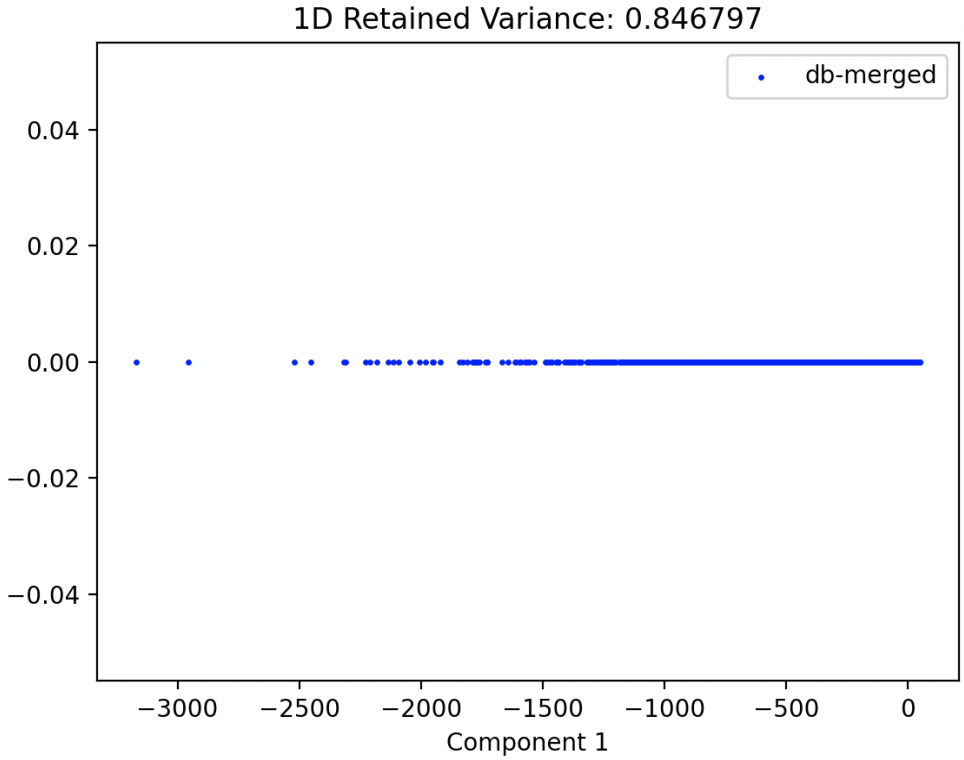
e. Component magnitude diagram

The following represents the alignment strength of individual components to each dimension. It can be interpreted as sources of error, where 0 represents perfect projection onto the component, and alternative values represent a magnitude and direction of displacement.



f. Reduced Matrix, A, depicted with r = 2 and r = 1





As noted in the upper diagram, the retained variance in 2 dimensions is 0.990566. This is an outstanding value, especially considering the drop from 10 dimensions to only 2. In fact, we can go lower to one dimension and still retain about 85% of the variance. By visualizing the 1d reduction in the lower diagram, we have an excellent visualization of data clusters. As we move along the x axis in the positive direction, the density of our data increases and we find clusters. This is a crystal clear trend, especially when looking at the 1d data. However, I would argue that the two dimensional view is the best overall (depicted above). While the 1d data clearly shows density, we can still observe this data easily by looking at the 2 dimensional version. Moreover, the 2D diagram presents other clear trends in the data, such as the apparent linear nature as well as its directional tendencies, as well two intriguing clusters of data that each have their own patterns. Those particular properties are lost when dropping down to 1 dimension, which is why I'd state that for this dataset, two dimensional reduction is excellent; boasting a 0.991 retention.