|  |  |  |  |
| --- | --- | --- | --- |
| **Learning objectives**   1. **Principal components analysis**   ***Note: You may want to do some supplementary reading on PCA, particularly if it interests you. Here is an optional resource to read before doing the assignment:***  ***https://cran.r-project.org/web/packages/LearnPCA/vignettes/Vig\_02\_Conceptual\_Intro\_PCA.pdf*** | | | |
| Principal components analysis (PCA) involves converting multi-dimensional data into another set of dimensions (the *principal components*) of equal or lesser number to help to better understand correlated structure of the original data.  Principal components are synthetic variables that contain/represent maximum variance in data, such that each component is orthogonal (uncorrelated) with all other components | | | |
| When it works, PCA can help reduce the complexity of large data sets without losing much information. The basic idea is that synthetic variables capture variation in the original data in a way that reduces the number of variables without losing information. The amount of information lost will depend on the amount of correlation in the data. Data that are largely uncorrelated cannot be reduced without losing information. Data that are highly correlated can be reduced to synthetic variables without losing much information.  Why is PCA used?  PCA can help organise data into groups with common attributes. Sometimes PCA is even useful for grouping observations together based on similarities of attributes. PCA can also be used to create ‘indices’ that help us understand complex concepts in simpler ways. | | | |
| Below is a plot of two variables measuring something. There are 21 points, each representing an observation. Each point has two measured values—measure 1 and measure 2. | | | |
|  | | | |
| Assume that Measures 1 & 2 are two measurements of work performance in some job. Looking at the plot, we would conclude that they are fairly correlated. We might ask if both measures are even necessary. We might ask if there is some way to represent these two measures as a single synthetic variable or ‘*index’* of work performance—which might make comparisons between employees easier. PCA might help us pick the best single measurement and simplify the interpretation of work performance. For example, when converted into one metric, the employee with the highest score on that metric might be the one who gets the promotion.  This is the rationale behind IQ tests. For many in psychometrics, the belief is that a test of many intellectual traits can be reduced to a single index—IQ—that captures the most important features of intelligence. This is why IQ is used to judge potential in some professions and identify students who would benefit from different learning methods. There are many issues with IQ tests, but we’ll not get into this here. | | | |
| A better rationale for PCA may come from the idea that ‘lower dimensionality’ is easier for humans to understand. Consider a dataset with 6 variables that describe a set of observations. Six variables is a lot to deal with. It might be that through PCA, we can reduce this dataset to one index that captures 80% of the variation in the data. When we do this successfully, we have an easier way of visualising, classifying and understanding data that could be representative of the information that is important to us. | | | |
| Now for an example. Download the following file to your computer:  <https://drive.google.com/file/d/1MUtOp60maADyYGktThHJ5SZLB5CqxF74/view?usp=sharing>  Import this file and name the dataframe ‘df’  These are monthly (January to December) precipitation data from 75 weather stations in Greece. The data set also includes latitude and longitude for the weather station locations. We’re going to use the data to understand precipitation in Greece. Specifically, we’ll try to understand the seasons of Greece based on these precipitation data. | | | |
| The first question to ask is if the monthly precipitation data are correlated. Correlation coefficients can vary between -1 and 1. The closer the correlation to 1, the stronger the positive correlation. The closer the correlation to -1 the stronger the negative correlation. The closer to 0, the weaker the correlation | |  | |
| No correlation | Moderate positive correlation | Moderate negative correlation | Strong positive correlation |
| Above is a correlation matrix. You can look at the correlations between months of precipitations in rows and columns. Along the diagonal, we see perfect correlation between a variable and itself, which is neither unexpected nor interesting. Also notice the above and below the diagonal are the same correlation coefficients—analogous to a mirror image. | | | |
| **Q1. Do some simple descriptive analysis, and provide a few brief sentences that summarise the information in this table.** | | | |
| We will now explore whether or not these  weather data can be usefully reduced using PCA.  Sometimes the variables we analyse are measured on very different scales. For example, income is in the tens of thousands but age is in the tens. They are measured at such different scales, it would make PCA difficult to interpret. By setting the scale and centre parameters to true, we are reducing the impact of absolute differences between observations. The components are more or less standardised for easier comparison. | |  | |
| The result is 12 principal components since we had 12 variables. The bottom row (cumulative proportion) explains the amount of variation in the dataset that is contained within components. The first principal component contains 77.17% of the variation in the data. By adding the second component, 87.68% of variation in the data is captured by the first and second components. This suggests two components alone (PC1 and PC2) might be able to represent much of the variation in the total data set (87.68% together). | |  | |
| It can be useful to think about PCA as a way of ‘projecting’ data onto a new multidimensional space. Before PCA, the data are measured onto a twelve-dimensional monthly precipitation space. Every station has a monthly precipitation value, so all stations exist in a twelve-dimensional space defined by these precipitation measurements.  PCA involves projecting observations onto synthetic variables in a new 12-dimensional space, and each station can be plotted in this new twelve-dimensional space. However, these synthetic variables are interesting because unlike the original data, each synthetic variable is **uncorrelated** with all other synthetic variables (unlike the real data). The same information exists in the 12 PCAs as the 12 original variables, but the synthetic variables (principal components—PCs) contain a mixture of information from the different original variables (called ‘loadings’).  In this example, the first synthetic variable (principal component—PC 1) captures more of the variability in the real variables than any other. The second PC, PC 2, captures more of the remaining variability in the real variables once the variability in PC 1 is accounted for, and so on. Since much of the total variability in these data is in PC1 and PC2, we don’t have to plot the data on all 12 dimensions—which would make no sense to our simple brains (which understand 2 and 3 dimensions at most). Instead we can plot on the first few two principal components, and understand much of the structure of these data. | | | |
| Use the code to the right to plot out the stations in pca1 and pca 2 coordinate space (sometimes called a ‘score plot’). PC1 and PC2 are the first two principal components | |  | |
|  | | | |
| Each dot above is a station location in the **synthetic** PC 1 and PC 2 space. Much of the variation between stations is on PC 1; note that the data spread out more along PC 1 (horizontal) than PC 2 (vertical). This isn’t a surprise since PC 1 is the component that contains over 75% of the variation in the data.  We see that there are a couple of stations that stand out from the clump in the middle. These two stations differ along PC 2, but are similar along PC 1 (both around 0 on the x-axis). PC2 is an important synthetic variable for explaining how these two stations are different from one another. | | | |
| Here is a handy R function that helps plot this all more elegantly | |  | |
|  | | | |
| The numbers are the stations. The left and bottom axes tell us the PCA scores for the weather stations, and are measured in standard deviations from the mean of the transformed data. They are the ‘locations’ of the data in PC space. They give us an idea of how "spread out" data are along each synthetic axis.  The red lines tell us about the contributions of different **variables** (monthly precipitation) to the two principal components (PC 1 and PC 2). **Parallel red lines in the same direction suggest positive correlation between variables. Parallel lines in the opposite direction suggest negative correlation between variables. Lines at right angles are uncorrelated.** The length of the line tells us how much the variable loads on the principal component. The top and right axes measure loadings—how much each original vector loads onto the PCs.  Many of the lines run horizontally and roughly parallel to PC 1, and are similar in length. This suggests a correlation between these variables, and that they all ‘load’ on the first principal component. In other words, PC 1 explains much of the variation of these variables.  The variables that stand out most are August and July (and maybe June and May). They run a little more in the vertical direction. This indicates that variation in precipitation in these months ‘load’ a bit more on PC 2. In fact a careful look will show that there is a seasonal pattern here. From May to August, there is a mix of loadings on PC 1 and PC 2, and for the other parts of the year, the loading is largely on PC 1. This analysis has revealed to us that precipitation varies seasonally in Greece. It does not tell us what the precipitation levels are (high or low, for example) but that could be found by calculating the range, mean, median and other statistics. PCA tells us about the structure of *relationships* between variables.  As noted before, we also see that stations #37 and #1 stand out. Plot out a ‘map’ of station locations (latitude and longitude) to see what stations are spatially proximate.    If you look at the output, you’ll see station #1 is close to #56 and station #37 is close to #32. I suspect there may be a data collection problem at #1 and #37, since they differ so much from their neighbours. Either that or a micro-climate or extreme weather event.  **Q2. Remove the observations from stations #1 and #37 and re-run the PCA analysis. Comment briefly on how this has changed the results.** | | | |
| Use this code to calculate the total precipitation for all stations:    If you look at the results from this, you’ll note that annual precipitation varies quite a bit from station to station (and ultimately, from region to region). However, the PCA suggests seasonal similarity; in other words, while some places are drier and some places are wetter, their rainy and dry seasons happen at about the same time of year.  Now use this code to calculate the total precipitation by month:    The final bit of information tells us which months are dry and wet. | | | |
| Now we’re going to try this all with fake/synthetic data. Copy and paste the code to the right. This is a data set with 6 variables that vary in how correlated they are with each other. Run the code to generate the fake data. | | set.seed(8675309)  x1 <- c(0)  x2 <- c(0)  x3 <- c(0)  x4 <- c(0)  x5 <- c(0)  x6 <- c(0)  for(i in 1:300){  x1[i] <- rnorm(1)  x2[i] <- rnorm(1) + x1[i]  x3[i] <- 2\*rnorm(1) + x1[i] + x2[i]/2  x4[i] <- rnorm(1) + x2[i] + x3[i]/2  x5[i] <- rnorm(1) - x1[i]  x6[i] <- rnorm(1) + x1[i]/3  }  prcomp(m, center=TRUE, scale=TRUE) | |
| **Q3. Create a correlation matrix using all the variables created in the previous step. Based on the code above and the results from the correlation matrix, discuss briefly what you expect to see from PCA. How many principal components might you expect to see explaining the bulk of variability in these data? Explain your answer in two to three sentences.** | | | |
| Write and run the code to the right. | |  | |
| **Q4. Create a biplot of these data, and then using the biplot and the results of the table of principal components, provide an interpretation of the structure of these data. Consider the angles of the red lines of the biplot. Are they pointing in the same direction? Are they at right angles? Are they pointing in opposite directions? Look back at how the data are synthesised and discuss what might explain the angle that these arrows are pointing. Explain your answer in no more than 3 to 4 sentences.** | | | |
| There are helpful visualisation tools in R for understanding the principal components. Write and run the code to the right. | |  | |
| The value of sdev (standard deviation) is associated with the principal components in order ( The first sdev is the standard deviation of the principal component 1, the second sdev is the standard deviation of the second principal component, etc.). By taking the ratio of the standard deviation to the sum of all standard deviation, we are seeing the variation ‘explained’ by that component. | | | |
| To plot this out, write and run the code to the right. This is sometimes called a ‘scree’ plot | |  | |
| Note that the first principal component contains about 60% of the variation. A lot of the variation in the data set is contained in PC 1, but not all of it. A combination of PC 1 and PC 2 might together summarise the data very well.  **Q5. Modify the code for this plot in some way to make it look nice and stuff!** | | | |
| To understand the nuts and bolts of PCA a little better, let’s create some more synthetic data. | |  | |
| **Q6. Write code that measures the correlations between all these variables.** | | | |
| Use the prcomp() function to create principal components | |  | |
|  | | | |
| The cumulative proportion gives a sense of the loading of the variables on each principal component. The first two components have about 70% of the variation in the four variables. | | | |
| Capy, paste and run the code to plot the first three principal components:  par(mfrow=c(1,3))  biplot(pca\_out,choices=c(1,2),main="PC1 and PC2",  cex = c(0.5,1.5),col=c("grey","red"))  biplot(pca\_out,choices=c(1,3),main="PC1 and PC3",  cex = c(0.5,1.5),col=c("grey","red"))  biplot(pca\_out,choices=c(2,3),main="PC2 and PC3",  cex = c(0.5,1.5),col=c("grey","red")) | | | |
|  | | | |
| Look carefully at the biplots mindful of the original data (V1 , V2, V3 and V4). (click the zoom button in the plots panel in RStudio to see things more clearly  Panel 1: Note that V1, V3 and V4 are roughly horizontal along PC 1. Variable V4 is pointing in the opposite direction because it is negatively correlated with V1 and V3. V2 is uncorrelated with the other variables, and loads more on PC 2. V4 is a little shorter, suggesting that while it loads on PC1, there is some other variation loading on the principal components.  Panel 2: The V2 line is very short—meaning that variation in this variable is not loaded onto PC1 and PC3. V4 loads more on PC3 than PC2, and V1 and V3 more on PC1 (same as in previous panel).  Panel 3: Note that V2 loads on PC2 (same as on panel 1) and v4 loads more on PC3 (same as on panel 2). In all panels V2 is perpendicular to the other variables. This confirms that it is uncorrelated with the other variables. | | | |
| You can use the code to the right to create a score plot for the first two principal components. Remember that each dot is an observation, but the locations are determined from the principal components. | |  | |
| **Q7. Reflect on why you see this pattern on the plot, and alter the process for creating synthetic data to generate a pattern with some sort of clustering or clumping of two or more groups in the PC1 and PC2 space. This may require some experimentation/trial and error. *Hint: think about changing the means and/or variances of the random number generating function for two of the variables. If subsets of the data have different means and/or variances, this can produce ‘clustering’.* Don’t rush this step of the experimentation process*,* as this exploration process will give you a deeper understanding of what the underlying data are saying.**  **Once you are done, re-run the prcomp() step and plot the components to show the ‘clumping’ with the synthetic data you create. Plot the biplot as well.** | | | |