Dimensionality Reduction with PCA

Rei Sanchez-Arias, Ph.D.

Principal Component Analysis (PCA)

Pre-requisites

Checklist

☑ Load the `tidyverse` package

```
library(tidyverse)
```

☑ Check materials in our Canvas page

New function(s) you will use today.

To perfom Principal Component Analysis (PCA) you will be using the function `prcomp()` from the `stats` package (you don't need to install any package to use it, since it comes with your R installation)

Example: cereals nutritional information

Cereals

Data on the nutritional information and consumer rating of 77 breakfast cereals is available. The consumer rating is a rating of cereal **"healthiness"** for consumer information (not a rating by consumers). For each cereal, the data include 13 numerical variables, and we are interested in **reducing this dimension**. For each cereal, the information is based on a *bowl of cereal* rather than a serving size, because most people simply fill a cereal bowl (resulting in constant volume, but not weight).

```
cereals <- read_csv("https://github.com/reisanar/datasets/raw/
```

Quick look

	name	rating	mfr	type	calories	protein
1	100%_Bran	68.402973	N	C	70	4
2	100%_Natural_Bran	33.983679	Q	C	120	3
3	All-Bran	59.425505	K	C	70	4
4	All-Bran_with_Extra_Fiber	93.704912	K	C	50	4

Previous 1 2 3 4 5 … 20 Next

PCA in two components

Using R to perform PCA

We use the `prcomp()` function in R. The calculation is done by a **singular value decomposition** of the (*centered* and possibly *scaled*) data matrix. We first use the attributes `calories` and `rating`:

```
# smaller dataset with 2 attributes
cereals_small <- cereals %>%
                  select(calories, rating)
# compute PCs on two dimensions
pcs <- prcomp(cereals_small)
summary(pcs)
```

```
## Importance of components:
##                             PC1    PC2
## Standard deviation     22.3165 8.8844
## Proportion of Variance  0.8632 0.1368
## Cumulative Proportion   0.8632 1.0000
```

Check results (1)

We can check the information from the principal component analysis by checking the different elements in the object `pcs`:

```
# the matrix of variable loadings
pcs$rotation
```

```
##                     PC1       PC2
## calories  0.8470535 0.5315077
## rating   -0.5315077 0.8470535
```

The first column is the projection onto z_1 using the weights (0.847, -0.532). The second column is the projection onto z_2 using the weights (0.532, 0.847).

Check results (2)

For example, the first score for the `100%_Bran` cereal (with 70 calories and a rating of 68.4) is

$$(0.847)(70-106.88)+(-0.532)(68.4-42.67)=-44.92.$$

(here 106.88 and 42.67 are the means of `calories` and `rating` respectively)

```
mean(cereals$calories)
```

```
## [1] 106.8831
```

```
mean(cereals$rating)
```

```
## [1] 42.6657
```

PCA space

The transformed variables are stored in the `x` list element:

```r
# the values of the transformed variables
head(pcs$x)
```

```
##              PC1        PC2
## [1,] -44.921528  2.1971833
## [2,]  15.725265 -0.3824165
## [3,] -40.149935 -5.4072123
## [4,] -75.310772 12.9991256
## [5,]   7.041508 -5.3576857
## [6,]   9.632769 -9.4873273
```

Variances

You can create the covariance matrix of calories and ratings:

```
var(cereals_small)
```

```
##            calories    rating
## calories  379.6309 -188.6816
## rating   -188.6816  197.3263
```

Notice that in this case, using the first principal component (a linear combination of `calories` and `rating`), allows us to capture 86.32 of the variation in the data (compared to the 66% of the variation explained by `calories` only)

```
# variation explained by calories
379.6309/(379.6309 + 197.3263)
```

```
## [1] 0.657988
```

PCA using more features

PCA in higher dimensions

Create a new data frame consisting of all 13 numerical variables, and removing any observation with missing information (those with `NA` values for any of the variables)

```
# new data frame
cereals_clean <- cereals %>%
                 select(-c(1:3)) %>%
                 drop_na()
# PCA in all numerical variables
pcs_all <- prcomp(cereals_clean)
summary(pcs_all)
```

Explained Variation

```
## Importance of components:
##                            PC1     PC2      PC3      PC4     PC5     PC6     PC7     PC8
## Standard deviation     83.7641 70.9143 22.64375 19.18148 8.42323 2.09167 1.69942 0.77963 0.
## Proportion of Variance  0.5395  0.3867  0.03943  0.02829 0.00546 0.00034 0.00022 0.00005 0.
## Cumulative Proportion   0.5395  0.9262  0.96560  0.99389 0.99935 0.99968 0.99991 0.99995 0.
##                           PC10   PC11    PC12      PC13
## Standard deviation     0.37043 0.1864 0.06302 5.334e-08
## Proportion of Variance 0.00001 0.0000 0.00000 0.000e+00
## Cumulative Proportion  1.00000 1.0000 1.00000 1.000e+00
```

The first 3 components account for more than 96% of the total variation associated with all 13 of the original variables. This suggests that we can capture most of the variability in the data with less than 25% of the original dimensions in the data. In fact, the first two principal components alone capture 92.6% of the total variation.

Pre-processing

Scaling

Let us check the loadings for the first 2 principal components:

```
pcs_all$rotation[ , 1:2]
```

```
##                   PC1           PC2
## calories  0.0779841812  0.0093115874
## protein  -0.0007567806 -0.0088010282
## fat      -0.0001017834 -0.0026991522
## sodium    0.9802145422 -0.1408957901
## fiber    -0.0054127550 -0.0306807512
## carbo     0.0172462607  0.0167832981
## sugars    0.0029888631  0.0002534853
## potass   -0.1349000039 -0.9865619808
## vitamins  0.0942933187 -0.0167288404
## shelf    -0.0015414195 -0.0043603994
## weight    0.0005120017 -0.0009992138
## cups      0.0005101111  0.0015910125
## rating   -0.0752962922 -0.0717421528
```

What are we measuring?

In our example, it is clear that the first principal component is dominated by the sodium content of the cereal: it has the highest (in this case, positive) weight.

This means that the first principal component is in fact measuring how much sodium is in the cereal. Similarly, the second principal component seems to be measuring the amount of potassium. Since both these variables are measured in milligrams, whereas the other nutrients are measured in grams, the scale is obviously leading to this result.

The variances of potassium and sodium are much larger than the variances of the other variables, and thus the total variance is dominated by these two variances. A solution is to **standardize** the data before performing the PCA.

Pre-processing?

The `prcomp()` function already *centers* by default (`center = TRUE`), so the only extra option we need to specify is `scale = TRUE`

Standardization means replacing each original variable by a standardized version of the variable that has unit variance. This is easily accomplished by dividing each variable by its standard deviation. *The effect of this is to give all variables equal importance in terms of variability*.

Domain Knowledge

When should we pre-process the data like this?

It depends on the nature of the data.

> If the variables are measured in different units so that it is unclear how to compare the variability of different variables (e.g., dollars for some, parts per million for others) or if for variables measured in the same units, scale does not reflect importance (earnings per share, gross revenues), it is generally advisable to standardize. In this way, the differences in units of measurement do not affect the principal components' weights. In the rare situations where we can give relative weights to variables, we multiply the scaled variables by these weights before doing the principal components analysis.

PCA (with standardized variables)

PCA output using all standardized numeric variables:

```
pcs_std <- prcomp(cereals_clean, scale = T)
summary(pcs_std)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4    PC5     PC6     PC7     PC8     PC9
## Standard deviation     1.9062 1.7743 1.3818 1.00969 0.9947 0.84974 0.81946 0.64515 0.56192
## Proportion of Variance 0.2795 0.2422 0.1469 0.07842 0.0761 0.05554 0.05166 0.03202 0.02429
## Cumulative Proportion  0.2795 0.5217 0.6685 0.74696 0.8231 0.87861 0.93026 0.96228 0.98657
##                           PC11    PC12      PC13
## Standard deviation     0.25194 0.13897 1.499e-08
## Proportion of Variance 0.00488 0.00149 0.000e+00
## Cumulative Proportion  0.99851 1.00000 1.000e+00
```

PCA results

Now we need 7 PCs to account for more than 90% of **total variability**. The first 2 PCs account for only 52% (by considering only 2 variables, we'd lose a lot of information.)

```
pcs_std$rot[ , 1:5]
```

```
##                  PC1        PC2          PC3         PC4         PC5
## calories  0.29954236  0.3931479 -0.114857453  0.20435870  0.20389885
## protein  -0.30735632  0.1653233 -0.277281953  0.30074318  0.31974897
## fat       0.03991542  0.3457243  0.204890102  0.18683311  0.58689327
## sodium    0.18339651  0.1372205 -0.389431009  0.12033726 -0.33836424
## fiber    -0.45349036  0.1798119 -0.069766079  0.03917361 -0.25511906
## carbo     0.19244902 -0.1494483 -0.562452458  0.08783547  0.18274252
## sugars    0.22806849  0.3514345  0.355405174 -0.02270716 -0.31487243
## potass   -0.40196429  0.3005442 -0.067620183  0.09087843 -0.14836048
## vitamins  0.11598020  0.1729092 -0.387858660 -0.60411064 -0.04928672
## shelf    -0.17126336  0.2650503  0.001531036 -0.63887859  0.32910135
## weight    0.05029930  0.4503085 -0.247138314  0.15342874 -0.22128334
## cups      0.29463553 -0.2122479 -0.139999705  0.04748909  0.12081645
## rating   -0.43837841 -0.2515389 -0.181842433  0.03831622  0.05758420
```

Explaining the dimensions

- Examining the weights, we see that the first principal component measures the balance between 2 quantities: (1) calories and cups (large positive weights) vs. (2) protein, fiber, potassium, and consumer rating (large negative weights).
- High scores on principal component 1 mean that the cereal is high in calories and the amount per bowl, and low in protein, and potassium. Unsurprisingly, this type of cereal is associated with a low consumer rating.
- The second principal component is most affected by the weight of a serving, and the third principal component by the carbohydrate content. We can continue labeling the next principal components in a similar fashion to learn about the structure of the data.