PCA using R: Calculation and Visualization

Rei Sanchez-Arias, Ph.D.

Principal Component Analysis (PCA)

Pre-requisites

Checklist

☑ Load the `tidyverse` package

```r
library(tidyverse)
```

PCA calculation and visualization

To perform 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)

☑ For data visualization you will be using the `factoextra` package

```r
library(factoextra)
```

Example: Boston Housing

House prices in Boston

For each neighborhood, a number of variables are given, such as the crime rate, the student/teacher ratio, and the median value of a housing unit in the neighborhood.

The file `BostonHousing.csv` contains information collected by the US Bureau of the Census concerning housing in the area of Boston, Massachusetts. The dataset includes information on 506 census housing tracts in the Boston area.

```
housing <- read_csv("https://raw.githubusercontent.com/reisanar/datasets/master/BostonHousing
```

Boston dataset

Variables	Description
`CRIM`	Crime rate
`ZN`	Percentage of residential land zoned for lots over 25,000 ft2
`INDUS`	Percentage of land occupied by non-retail business
`CHAS`	Does tract bound Charles River (`= 1` if tract bounds river, `= 0` otherwise)
`NOX`	Nitric oxide concentration (parts per 10 million)
`RM`	Average number of rooms per dwelling
`AGE`	Percentage of owner-occupied units built prior to 1940
`DIS`	Weighted distances to five Boston employment centers
`RAD`	Index of accessibility to radial highways
`TAX`	Full-value property tax rate per $10,000
`PTRATIO`	Pupil-to-teacher ratio by town
`LSTAT`	Percentage of lower status of the population
`MEDV`	Median value of owner-occupied homes in $1000s

Explore the dataset

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	LSTAT	MEDV	CAT_MED
1	0.00632	18	2.31	0	0.538	6.575	65.2	4.09	1	296	15.3	4.98	24	
2	0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	9.14	21.6	
3	0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	4.03	34.7	
4	0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	2.94	33.4	

Previous 1 2 3 4 5 … 127 Next

The first row represents the first neighborhood, which had an average per capita crime rate of 0.006, 18% of the residential land zoned for lots over 25,000 ft2, 2.31% of the land devoted to non-retail business, no border on the Charles River.

Continuous variables subset

Let us consider a smaller data set of continuous variables only:

```
red_boston <- housing %>%
                    select(-c(ZN, RAD, TAX, CHAS, CAT_MEDV))
```

Are there any missing values?

```
red_boston %>%
  is.na() %>%
  sum()
```

```
## [1] 0
```

Other summaries

Summaries   R Code

```
## # A tibble: 14 x 5
##    var_name var_mean  var_sd var_median var_miss
##    <chr>       <dbl>   <dbl>      <dbl>    <dbl>
##  1 CRIM       3.61     8.60       0.257        0
##  2 ZN        11.4     23.3        0            0
##  3 INDUS     11.1      6.86       9.69         0
##  4 CHAS       0.0692   0.254      0            0
##  5 NOX        0.555    0.116      0.538        0
##  6 RM         6.28     0.703      6.21         0
##  7 AGE       68.6     28.1       77.5          0
##  8 DIS        3.80     2.11       3.21         0
##  9 RAD        9.55     8.71       5            0
## 10 TAX      408.     169.       330            0
## 11 PTRATIO   18.5      2.16      19.0          0
## 12 LSTAT     12.7      7.14      11.4          0
## 13 MEDV      22.5      9.20      21.2          0
## 14 CAT_MEDV   0.166    0.372      0            0
```

Matrix of correlations

	CRIM	INDUS	NOX	RM	AGE	
CRIM	1	0.406583411406259	0.420971711392456	-0.219246702862514	0.352734250901364	-0.3796700869
INDUS	0.406583411406259	1	0.763651446920915	-0.391675852656844	0.644778511355256	-0.7080269887
NOX	0.420971711392456	0.763651446920915	1	-0.302188187849594	0.731470103785959	-0.7692301132
RM	-0.219246702862514	-0.391675852656844	-0.302188187849594	1	-0.240264931047751	0.2052462129
AGE	0.352734250901364	0.644778511355256	0.731470103785959	-0.240264931047751	1	-0.7478805408
DIS	-0.379670086951024	-0.708026988742768	-0.769230113225828	0.205246212930055	-0.747880540868632	
PTRATIO	0.28994557927952	0.383247556428888	0.188932677112767	-0.355501494559085	0.261515011671958	-0.2324705424
LSTAT	0.455621479447946	0.603799716476621	0.590878920880846	-0.613808271866396	0.60233852872624	-0.4969958308
MEDV	-0.388304608586812	-0.483725160028373	-0.427320772373283	0.695359947071539	-0.376954565004596	0.2499287340

PCA Calculation

Using R to perform PCA

We use the `prcomp()` function to perform principal component analysis (PCA) on the Boston housing dataset:

```
pca_boston <- prcomp(red_boston, scale = T)
summary(pca_boston)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7     PC8     PC9
## Standard deviation     2.1897 1.2194 0.88062 0.82777 0.62255 0.55048 0.46566 0.44013 0.3958
## Proportion of Variance 0.5327 0.1652 0.08617 0.07613 0.04306 0.03367 0.02409 0.02152 0.0174
## Cumulative Proportion  0.5327 0.6979 0.78411 0.86024 0.90331 0.93698 0.96107 0.98259 1.0000
```

The first 5 principal component explain ~90% of the variation in the collection of 506 data points.

PCA results

```r
pca_boston$rotation[ , 1:5] # check the loadings for 5 components
```

```
##                PC1          PC2         PC3         PC4         PC5
## CRIM     0.2653555  0.005326466  0.69615209 -0.63199277 -0.09225548
## INDUS    0.3858040 -0.161284875 -0.02754606  0.17735055 -0.52587819
## NOX      0.3776718 -0.315809534 -0.13507662 -0.05651541 -0.18264780
## RM      -0.2721135 -0.486149250  0.40513709  0.10361741  0.35581083
## AGE      0.3591693 -0.313625989 -0.09037383  0.12853860  0.58955195
## DIS     -0.3473877  0.415947651  0.01111363 -0.14784380  0.11806503
## PTRATIO  0.2322728  0.356013692  0.53264090  0.68418539  0.03822821
## LSTAT    0.3863031  0.183393397 -0.17524468 -0.20216886  0.40169556
## MEDV    -0.3334643 -0.454014830  0.09758974  0.08854101 -0.17506546
```

- PC1 largest loadings come from `LSTAT` (percentage of lower status of the population) , `INDUS` (percentage of land occupied by non-retail business), `NOX` (nitric oxide concentration)
- PC2 largest loadings come from (positive) `DIS` (weighted distances to 5 Boston employment centers), and (negative) `MEDV` (median value of owner-occupied homes), (negative) `RM` (average number of rooms)

PCA Visualization

Biplots

A ***biplot*** is a plot which aims to represent both the observations and variables of a matrix of multivariate data on the same plot. There are many variations on biplots.

A loading plot shows *how strongly* each characteristic influences a principal component. The angles between the vectors tell us how characteristics *correlate with one another*.

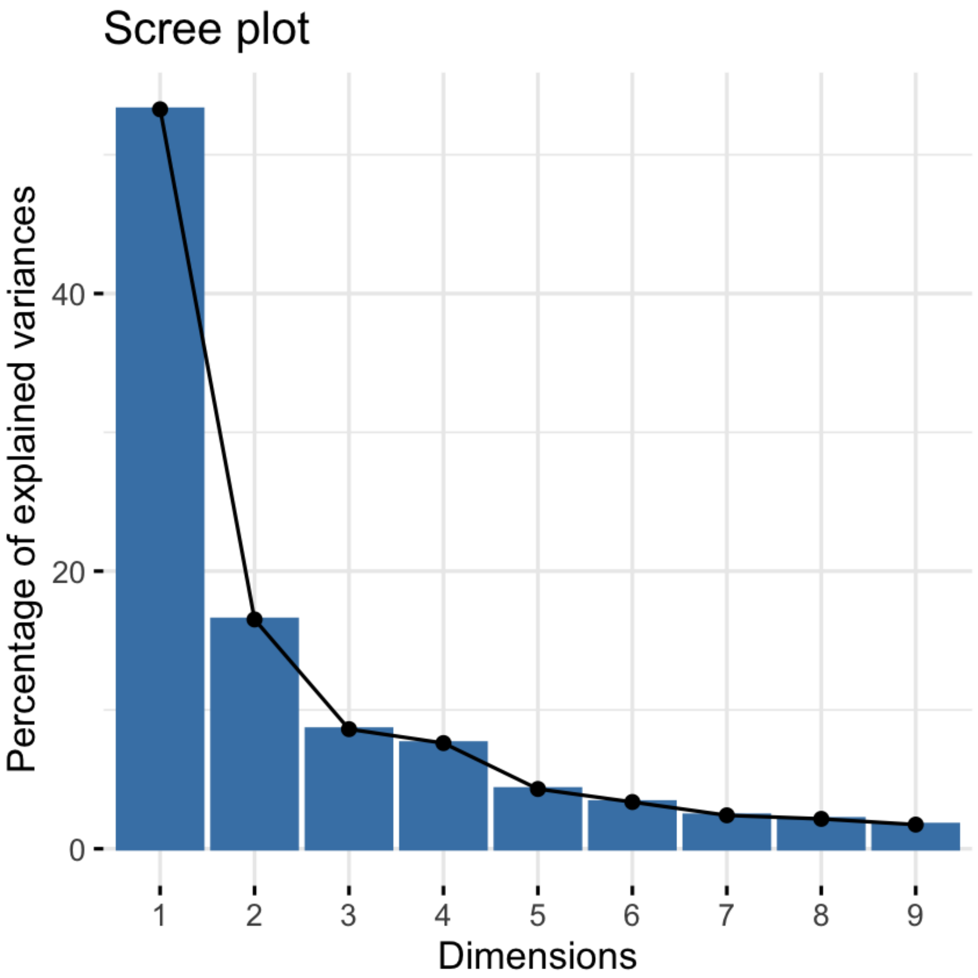
> When two vectors are close, forming a small angle, the two variables they represent are positively correlated. If they meet each other at a right angle, they are not likely to be correlated. When they diverge and form a large angle (close to 180 degrees), they are negative correlated.

Scree-plot

A ***scree-plot*** can be easily generated to show the contribution of each component to explain the variation in the data.

```
library(factoextra)
```

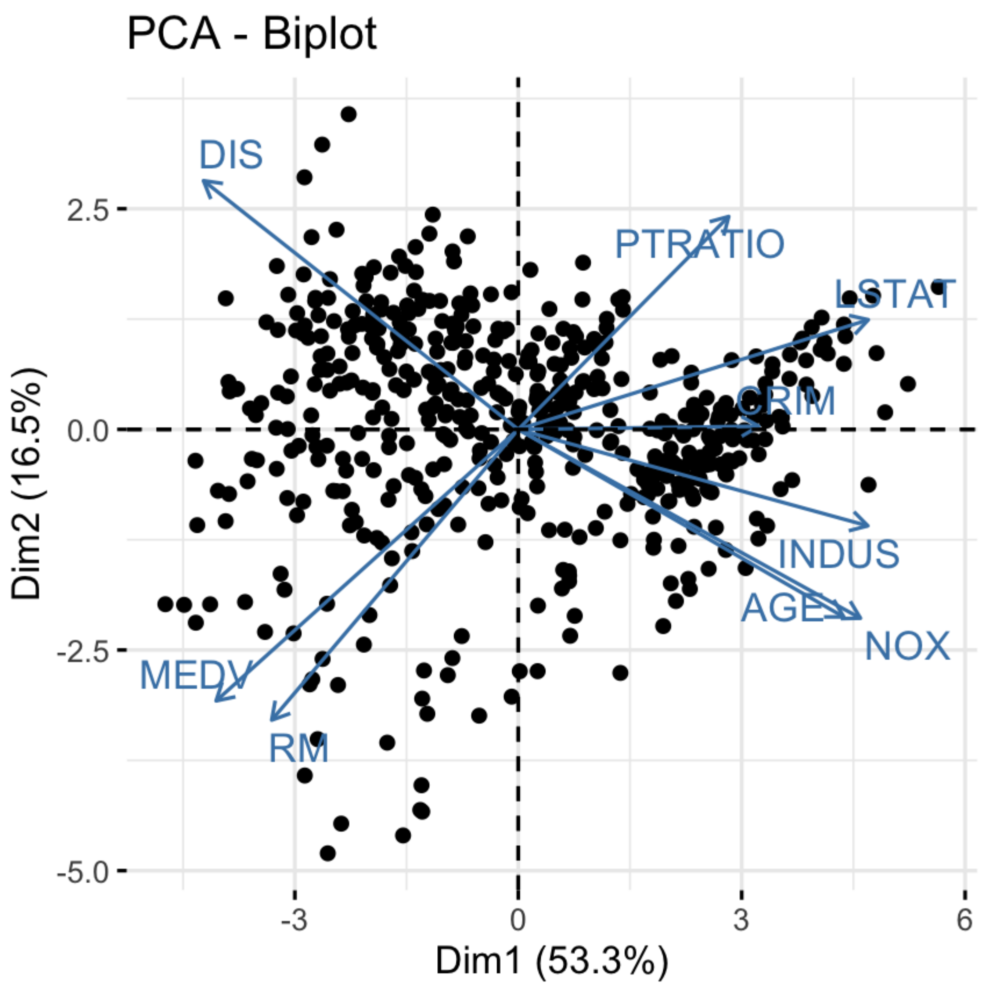
```
fviz_screeplot(pca_boston)
```



Biplot

Biplots can be generated using `factoextra::fviz_pca()`.

```
fviz_pca(pca_boston,
         geom = "point",
         repel = TRUE)
```



Loadings

The location of the loading vectors for each feature in the PC1-PC2 plane can be found by looking at the `rotation` list element of the PCA object

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

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

```
##                PC1          PC2
## CRIM     0.2653555  0.005326466
## INDUS    0.3858040 -0.161284875
## NOX      0.3776718 -0.315809534
## RM      -0.2721135 -0.486149250
## AGE      0.3591693 -0.313625989
## DIS     -0.3473877  0.415947651
## PTRATIO  0.2322728  0.356013692
## LSTAT    0.3863031  0.183393397
## MEDV    -0.3334643 -0.454014830
```

Correlations

Does it make sense that the loading vectors for `DIS` and `NOX` point in opposite directions?

```r
# check correlation for DIS and NOX
red_boston %>%
  select(DIS, NOX) %>%
  cor()
```

```
##              DIS        NOX
## DIS  1.0000000 -0.7692301
## NOX -0.7692301  1.0000000
```

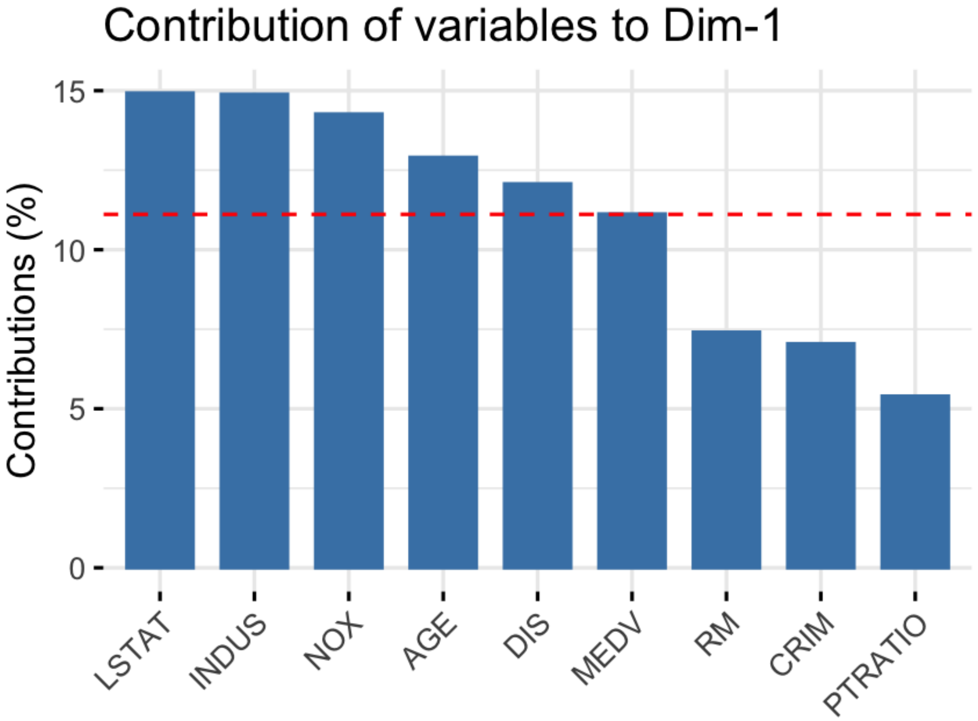
Similarly for `PTRATIO` and `MEDV` which are negatively correlated variables:

```r
# check correlation for PTRATIO and MEDV
red_boston %>%
  select(PTRATIO, MEDV) %>%
  cor()
```

```
##              PTRATIO       MEDV
## PTRATIO  1.0000000 -0.5077867
## MEDV    -0.5077867  1.0000000
```

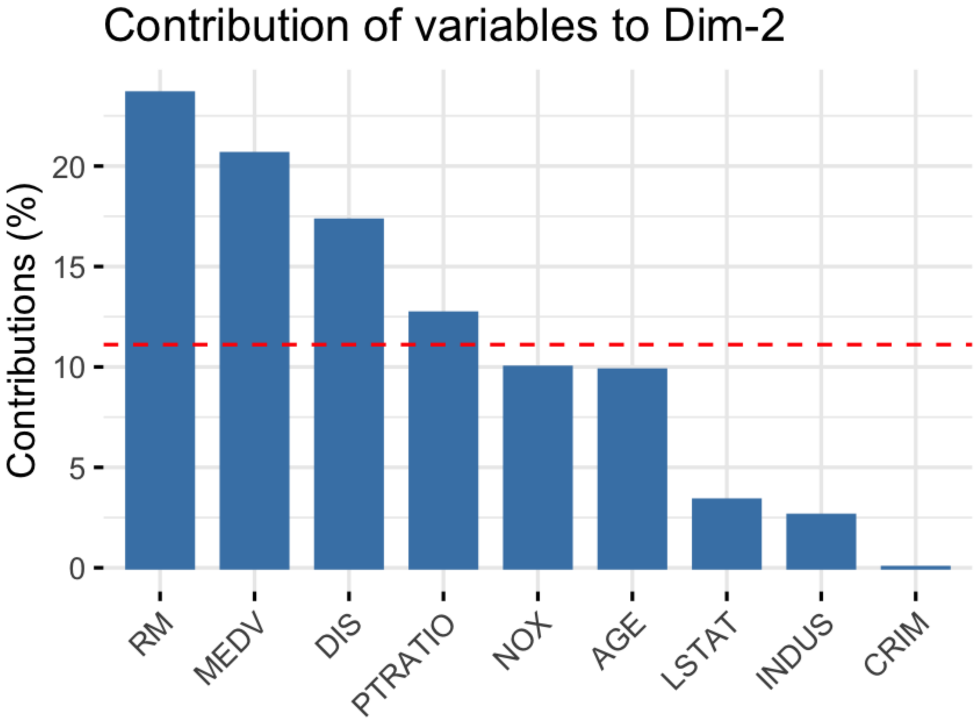
Contributions to principal components

```r
fviz_contrib(pca_boston, choice = "var",
             axes = 1)
```



Contribution of variables to Dim-1

```r
fviz_contrib(pca_boston, choice = "var",
             axes = 2)
```



Contribution of variables to Dim-2