Principal Components Analysis (part I)

Stat 133 CCwD

Gaston Sanchez

Introduction

Data set state.x77

```r
dim(state.x77)
```

```
[1] 50  8
```

```r
head(state.x77, 10)
```

```
            Population Income Illiteracy Life Exp Murder HS Grad Frost   Area
Alabama           3615   3624        2.1    69.05   15.1    41.3    20  50708
Alaska             365   6315        1.5    69.31   11.3    66.7   152 566432
Arizona           2212   4530        1.8    70.55    7.8    58.1    15 113417
Arkansas          2110   3378        1.9    70.66   10.1    39.9    65  51945
California       21198   5114        1.1    71.71   10.3    62.6    20 156361
Colorado          2541   4884        0.7    72.06    6.8    63.9   166 103766
Connecticut       3100   5348        1.1    72.48    3.1    56.0   139   4862
Delaware           579   4809        0.9    70.06    6.2    54.6   103   1982
Florida           8277   4815        1.3    70.66   10.7    52.6    11  54090
Georgia           4931   4091        2.0    68.54   13.9    40.6    60  58073
```

Data set `state.x77`

US State Facts and Figures

- `Population`: population estimate as of July 1, 1975
- `Income`: per capita income (1974)
- `Illiteracy`: illiteracy (1970, percent of population)
- `Life Exp`: life expectancy in years (1969-71)
- `Murder`: murder rate per 100,000 population (1976)
- `HS Grad`: percent high-school graduates (1970)
- `Frost`: avg num of days with minimum temp below freezing (1931-1960) in capital or large city
- `Area`: land area in square miles

Data Perspectives

We are interested in analyzing a data set from both perspectives: **objects** (rows) and **variables** (columns)

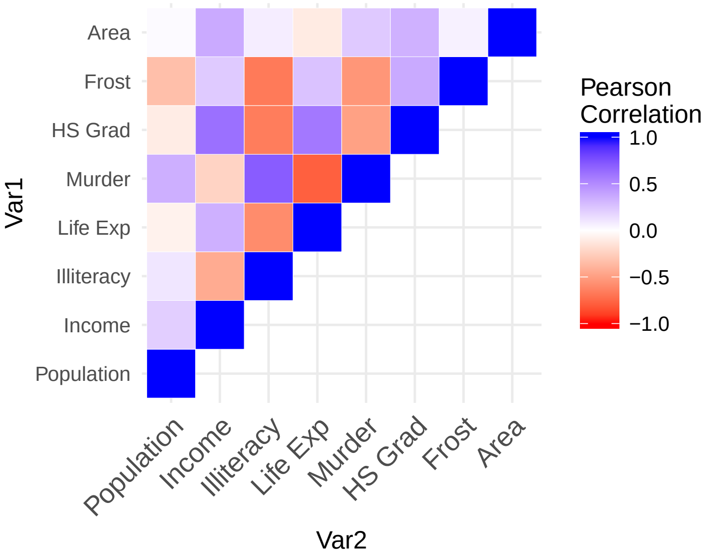
At its simplest we are interested in 2 fundamental purposes:

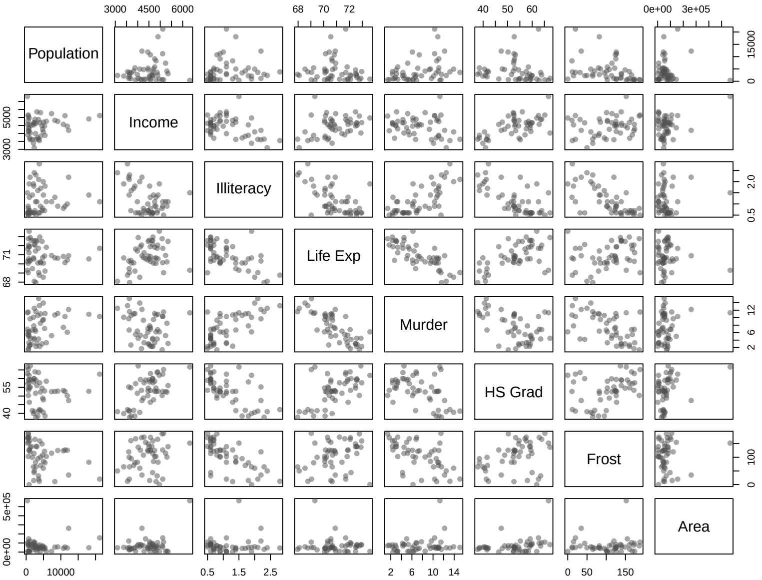
- Study relationship among variables
  (relationship among state statistics)
- Study resemblance among individuals
  (resemblance among states)

Relationship between Variables

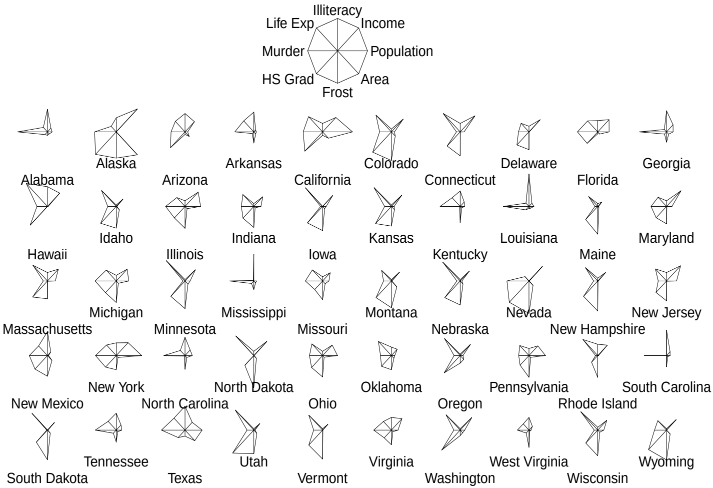
Matrix of Correlations

	Population	Income	Illiteracy	Life Exp	Murder	HS Grad	Frost	Area
Population	1.000							
Income	0.208	1.000						
Illiteracy	0.108	-0.437	1.000					
Life Exp	-0.068	0.340	-0.588	1.000				
Murder	0.344	-0.230	0.703	-0.781	1.000			
HS Grad	-0.098	0.620	-0.657	0.582	-0.488	1.000		
Frost	-0.332	0.226	-0.672	0.262	-0.539	0.367	1.000	
Area	0.023	0.363	0.077	-0.107	0.228	0.334	0.059	1


Area
Frost
HS Grad
Murder
Life Exp
Illiteracy
Income
Population
Var1
Population
Income
Illiteracy
Life Exp
Murder
HS Grad
Frost
Area
Var2
Pearson
Correlation
1.0
0.5
0.0
−0.5
−1.0


Population
Income
Illiteracy
Life Exp
Murder
HS Grad
Frost
Area

Resemblance among individuals


Illiteracy
Life Exp
Income
Murder
Population
HS Grad
Area
Frost
Alabama
Alaska
Arizona
Arkansas
California
Colorado
Connecticut
Delaware
Florida
Georgia
Hawaii
Idaho
Illinois
Indiana
Iowa
Kansas
Kentucky
Louisiana
Maine
Maryland
Massachusetts
Michigan
Minnesota
Mississippi
Missouri
Montana
Nebraska
Nevada
New Hampshire
New Jersey
New Mexico
New York
North Carolina
North Dakota
Ohio
Oklahoma
Oregon
Pennsylvania
Rhode Island
South Carolina
South Dakota
Tennessee
Texas
Utah
Vermont
Virginia
Washington
West Virginia
Wisconsin
Wyoming

Code chunks

```r
# correlation matrix
cormat <- cor(state.x77)
cormat[upper.tri(cormat)] <- NA
print(round(cormat, 3), na.print = '')
```

```r
# scatterplot matrix
pairs(state.x77, pch = 19, col = "#50505080")
```

```r
# looking at individuals (star plot)
stars(state.x77, nrow = 5, key.loc = c(12, 14))
```

Proposal

Let's try to summarize the systematic variation of the variables in `state.x77`

Naive approach

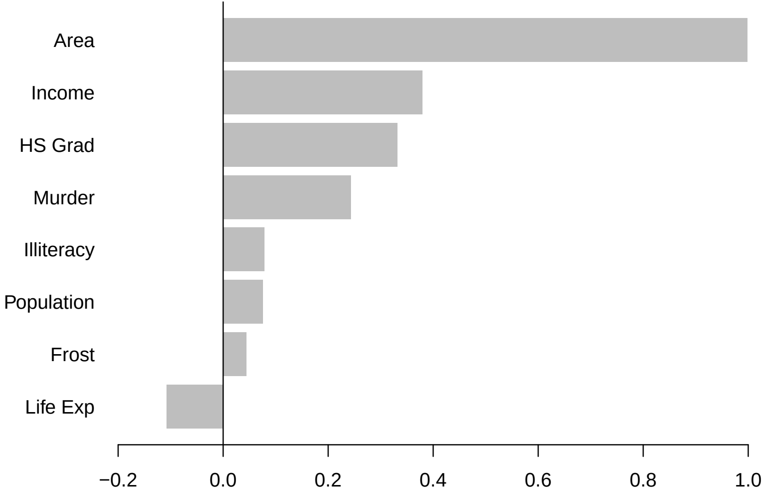
Summarizing variability with a new synthetic variable obtained by adding all the observed variables:

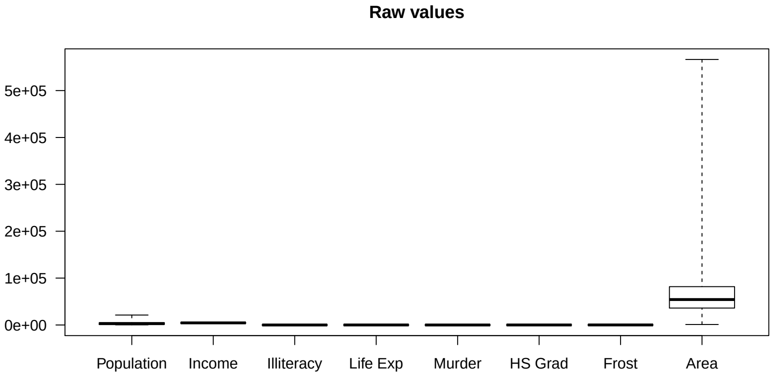
```
# first attempt: adding all variables
var_sum1 <- rowSums(state.x77)

# top-10 states
head(sort(var_sum1, decreasing = TRUE), n = 10)

##     Alaska      Texas California    Montana New Mexico     Ari
##   573412.8   278726.7   182838.7   150970.4   126414.4    1203
##   Colorado     Oregon    Wyoming
##   111500.5   103308.9   102458.7
```

Naive approach

```r
# correlations with var_sum1
corrs1 <- cor(state.x77, var_sum1)
corrs1

##                    [,1]
## Population  0.07576495
## Income      0.37958012
## Illiteracy  0.07888152
## Life Exp   -0.10767663
## Murder      0.24308112
## HS Grad     0.33139064
## Frost       0.04386621
## Area        0.99855742
```

Naive approach





If you use the raw scales, `Area` will dominate the analysis due to its larger scale.

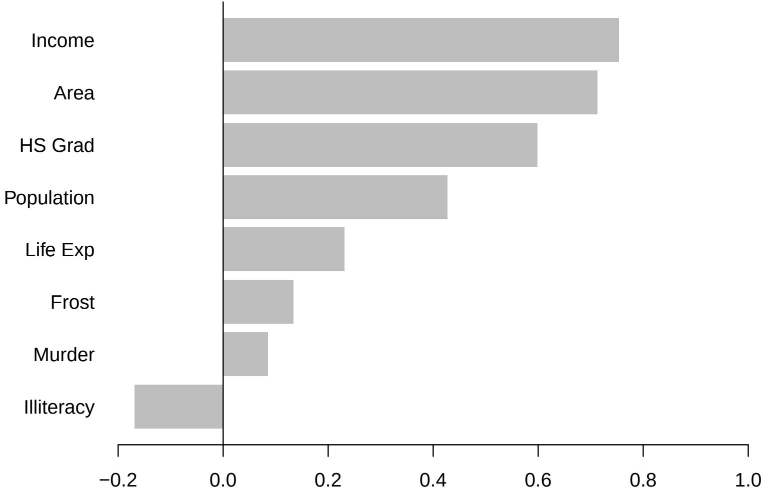
Naive approach II

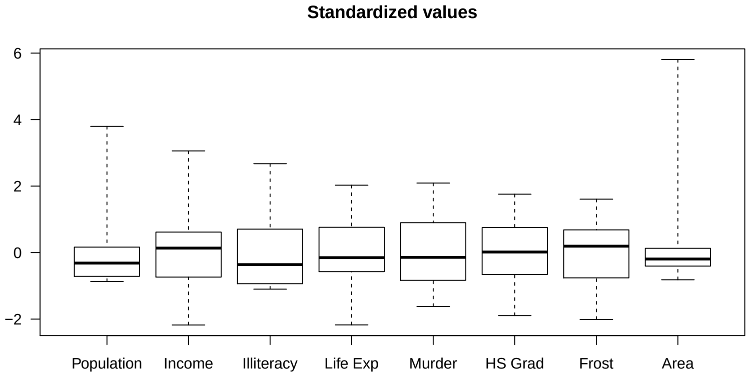
```
# second attempt: adding all standardized variables
state2 <- scale(state.x77)
var_sum2 <- rowSums(state2)

# top-10 states
head(sort(var_sum2, decreasing = TRUE), n = 10)

##      Alaska  California       Texas    New York    Colorado
##   11.030886    6.750584    4.598668    4.193607    3.206963
##    Michigan Connecticut      Hawaii
##    2.176565    1.397496    1.332289
```

Naive approach II

```
# correlations with var_sum2
corrs2 <- cor(state2, var_sum2)
corrs2

##                   [,1]
## Population  0.42668030
## Income      0.75390591
## Illiteracy -0.16832589
## Life Exp    0.23070560
## Murder      0.08553865
## HS Grad     0.59812456
## Frost       0.13390789
## Area        0.71283299
```

Naive approach II

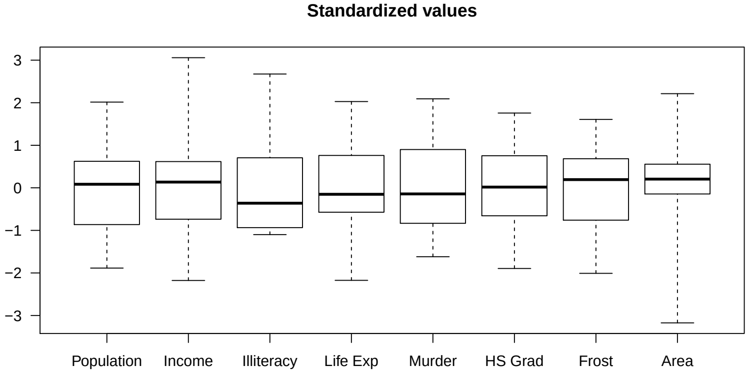




By standardizing the variables, they have a better balance, although some variables have extremely skewed distributions.

Naive approach III

```r
# log-transform Area and Population
state3 <- state.x77
state3[ ,'Population'] <- log(state.x77[ ,'Population'])
state3[ ,'Area'] <- log(state.x77[ ,'Area'])
state3 <- scale(state3)
```


Standardized values
3
2
1
0
−1
−2
−3
Population
Income
Illiteracy
Life Exp
Murder
HS Grad
Frost
Area

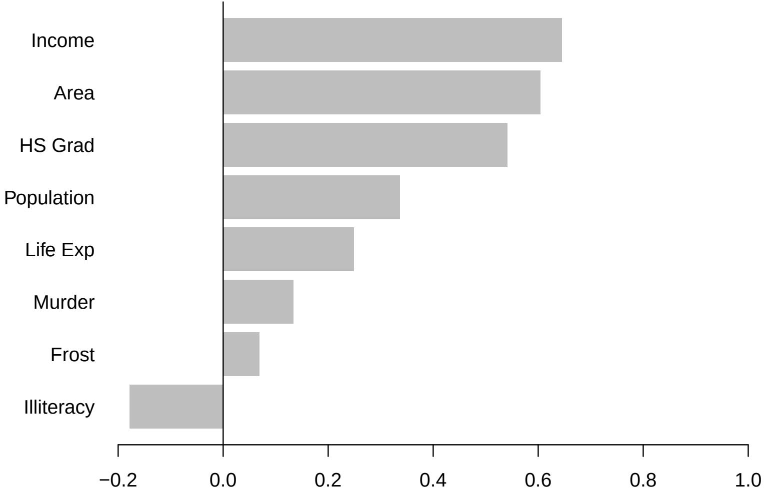
Naive approach III

```
# third attempt: adding transformed and standardized variables
var_sum3 <- rowSums(state3)

# top-10 states
head(sort(var_sum3, decreasing = TRUE), n = 10)

##     Alaska California   Colorado      Texas   New York   Illi
##   6.418510   5.075508   3.939468   3.605958   3.322728   2.89
##     Nevada  Minnesota     Kansas
##   2.289938   2.177462   1.386401
```

Naive approach III

```
# correlations with var_sum3
corrs3 <- cor(state3, var_sum3)
corrs3

##                   [,1]
## Population  0.33618819
## Income      0.64470509
## Illiteracy -0.17831628
## Life Exp    0.24895860
## Murder      0.13385965
## HS Grad     0.54116343
## Frost       0.06876995
## Area        0.60393702
```

Naive approach III



Naive approaches ...

```
new_vars <- data.frame(
  sum1 = var_sum1,
  sum2 = var_sum2,
  sum3 = var_sum3,
  row.names = rownames(state.x77)
)

# which synthetic variable is better?
print(head(new_vars), print.gap = 3, digits = 3)

##                 sum1       sum2     sum3
## Alabama        58095   -2.52867   -1.689
## Alaska        573413   11.03089    6.419
## Arizona       120312   -0.00204    0.634
## Arkansas       57621   -3.04247   -2.377
## California    182839    6.75058    5.076
## Colorado      111500    3.20696    3.939
```

About PCA

Data Structure

Principal Components Analysis (PCA) is a multivariate method that allows us to study and explore a set of quantitative variables measured on some objects.

About PCA

Approaches:

PCA can be presented using various—different but equivalent—approaches. Each approach corresponds to a unique perspective and a way of thinking about data.

- Data dispersion from the individuals standpoint
- Data variability from the variables standpoint
- Data that follows a decomposition model

I will present PCA by mixing and connecting all of these approaches.

Landmarks

- PCA was first introduced by Karl Pearson (1904)
  On lines and planes of closest fit to systems of points in space
- Further developed by Harold Hotelling (1933)
  Analysis of a complex of statistical variables into principal components
- Singular Value Decomposition (SVD) theorem by Eckart-Young (1936)
  The approximation of a matrix by another of a lower rank
- Computationally implemented in the 1960s

Core Idea

With PCA we seek to **reduce the dimensionality** (condense information in variables) of a data set while retaining as much as possible of the variation present in the data

PCA: Overall Goals

- Summarize a data set with the help of a small number of synthetic variables (i.e. the Principal Components).
- Visualize the position (resemblance) of individuals.
- Visualize how variables are correlated.
- Interpret the synthetic variables.

Maximizing Variability Approach

Data Matrix

The analyzed data can be expressed in matrix format $\mathbf{X}$:

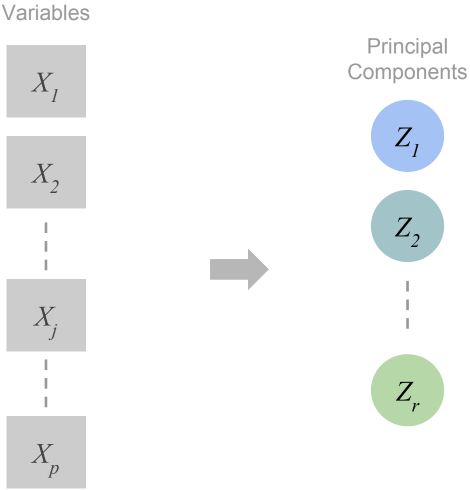
$$\underset{n \times p}{\mathbf{X}} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$$

- n objects in the rows
- p variables in the columns
- We'll assume standardized variables (mean $=$ 0, var $=$ 1)

Looking for PCs

Given a set of p variables $X_1, X_2, \ldots, X_p$, we want to obtain new r variables $Z_1, Z_2, \ldots, Z_r$, called the **Principal Components** (PCs).

Looking for PCs



Looking for PCs

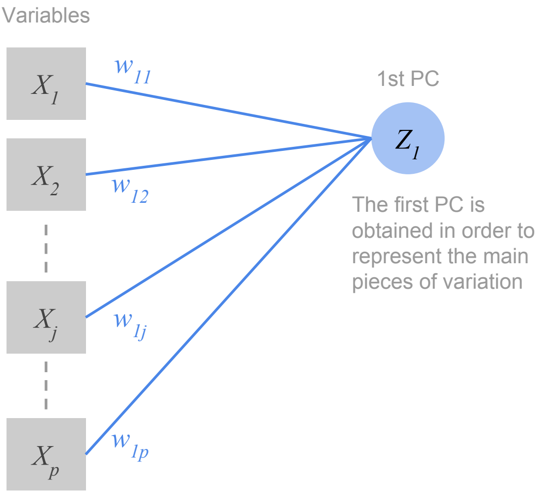
PC as linear combinations

We want to compute the **PCs as linear combinations** of the original variables.

$$
\begin{array}{ccl}
\mathsf{PC_1} \longrightarrow & Z_1 = w_{11}X_1 + w_{12}X_2 + \cdots + w_{1p}X_p \\
\mathsf{PC_2} \longrightarrow & Z_2 = w_{21}X_1 + w_{22}X_2 + \cdots + w_{2p}X_p \\
\vdots & \vdots \\
\mathsf{PC_r} \longrightarrow & Z_r = w_{r1}X_1 + w_{r2}X_2 + \cdots + w_{rp}X_p
\end{array}
$$

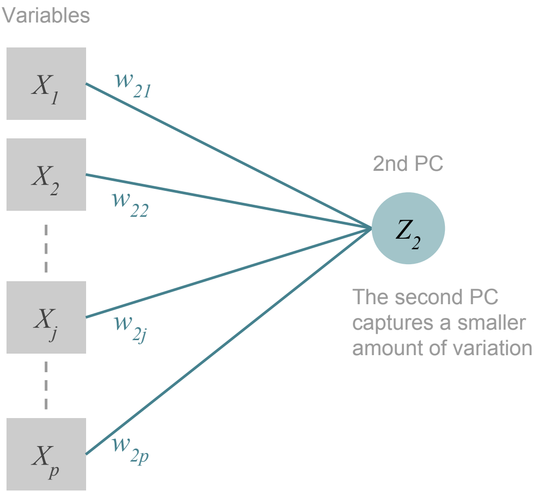
(i.e. linear combination = weighted sum)

1st PC

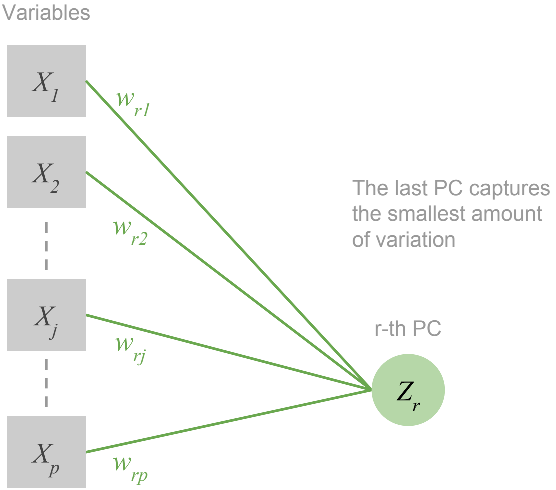


Variables
X_1
X_2
X_j
X_p
w_{11}
w_{12}
w_{1j}
w_{1p}
1st PC
Z_1
The first PC is obtained in order to represent the main pieces of variation

2nd PC

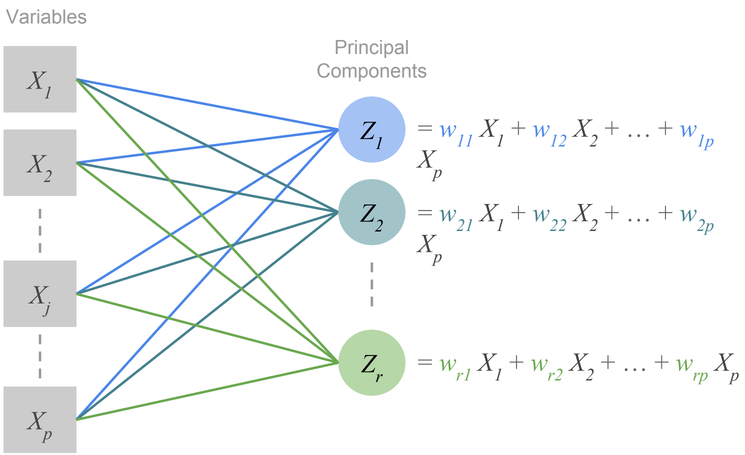

Variables
X_1
X_2
X_j
X_p
w_{21}
w_{22}
w_{2j}
w_{2p}
2nd PC
Z_2
The second PC captures a smaller amount of variation

k-th PC



Variables
X_1
X_2
X_j
X_p
w_{r1}
w_{r2}
w_{rj}
w_{rp}
The last PC captures the smallest amount of variation
r-th PC
Z_r

PCs as linear combinations



Introductory Recap

Summarize Variation

We look to transform the original variables into a smaller set of new variables, the Principal Components, that summarize the variation in data.

PCs

The PCs are obtained as linear combinations (i.e. weighted sums) of the original variables. We look for PCs having maximum variance, and being mutually uncorrelated.

Finding all PCs

Diagonalization

All Principal Components can be found simultaneously by **diagonalizing** $\frac{1}{n-1}\mathbf{X}^\intercal\mathbf{X}$

Eigenvalue Decomposition (EVD)

Diagonalizing a matrix is nothing more than obtaining its eigenvalue decomposition (a.k.a. spectral decomposition)

Data Decomposition

Algebraically

PCA involves an **Eigen-Value Decomposition** (EVD) of the data matrix $\frac{1}{n-1}\mathbf{X}^\top\mathbf{X}$, that is:

$$\frac{1}{n-1}\mathbf{X}^\top\mathbf{X} = \mathbf{W}\mathbf{\Lambda}\mathbf{W}^\top$$

- $\mathbf{W}$ is orthonormal matrix of eigenvectors (i.e. $\mathbf{W}^\top\mathbf{W} = \mathbf{I}$)
- $\mathbf{\Lambda}$ is a diagonal matrix of eigenvalues

EVD Approach

PCs

Principal components $\mathbf{Z} = [Z_1|Z_2|\ldots|Z_k]$ are obtained as:

$$\mathbf{Z} = \mathbf{XW}$$

Note that the variance of each component turns out to be equal to its associated eigenvalue:

$$var(\mathbf{z_k}) = \frac{1}{\sqrt{n-1}} \mathbf{z_k}^\mathsf{T} \mathbf{z_k} = \lambda_k$$

PCA in R

Eigenvalues, Scores, Loadings

The minimal output from any PCA should contain 3 things:

- **Eigenvalues** provide information about the amount of variability captured by each principal component
- **Scores** or PCs that provide coordinates to graphically represent objects in a lower dimensional space
- **Loadings** provide information to determine what variables characterize each principal component

PCA in R

Some PCA functions (and packages) in R

Function	Package	Author
`prcomp()`	`stats`	R Core Team
`princomp()`	`stats`	R Core Team
`PCA()`	`FactoMineR`	Husson, Josse, Le, Mazet
`dudi.pca()`	`ade4`	Chessel, Dufour, Dray
`acp()`	`amap`	Lucas
`nipals()`	`plsdepot`	Sanchez
`rda()`	`vegan`	Oksanen *et al*
`pca()`	`pcaMethods`*	Stacklies, Redestig, Wright

*See `http://www.bioconductor.org/packages/release/bioc/html/pcaMethods.html`

PCA with prcomp()

One of the default PCA functions in R is `prcomp()`:

```
# PCA with prcomp()
pca <- prcomp(state3, scale. = TRUE)

# what does prcomp() provide?
names(pca)

## [1] "sdev"     "rotation" "center"   "scale"    "x"
```

`scale.= TRUE` indicates that PCA is performed on standardized data (mean = 0, variance = 1)

Table of eigenvalues

```
# eigenvalues
eigs <- data.frame(
  eigenvalue = pca$sdev^2,
  proportion = round(100 * pca$sdev^2 / sum(pca$sdev^2), 3)
)
eigs

##   eigenvalue proportion
## 1  3.6940711     46.176
## 2  1.3227218     16.534
## 3  1.1200498     14.001
## 4  0.7368854      9.211
## 5  0.6460975      8.076
## 6  0.2369520      2.962
## 7  0.1394354      1.743
## 8  0.1037870      1.297
```

PCA with prcomp() con't

```
# scores
round(head(pca$x, 10), 2)
```

```
##              PC1   PC2   PC3   PC4   PC5   PC6   PC7   PC8
## Alabama    -3.81 -0.12  0.26  0.03  0.43  0.34  0.03  0.52
## Alaska      1.03  2.51  2.87 -2.46  1.10 -1.14 -0.29 -0.20
## Arizona    -0.94  1.05  0.03  0.25  1.64  0.09 -0.37 -0.58
## Arkansas   -2.35 -1.08  0.24  1.03  0.32 -0.36 -0.02  0.49
## California -0.33  3.07 -1.22  0.43  0.33  0.48  0.17  0.02
## Colorado    1.93  1.02  0.59  0.10 -0.27 -0.17  0.67  0.11
## Connecticut 1.90 -0.58 -1.83 -1.05  0.00 -0.73  0.32 -0.14
## Delaware    0.78 -1.86 -0.68 -2.09  0.66  0.81 -0.27  0.13
## Florida    -1.31  1.51 -1.07 -0.15  0.43  0.45 -0.40  0.27
## Georgia    -3.36  0.14  0.38 -0.52 -0.35 -0.09 -0.14  0.14
```

PCA with prcomp() con't

```r
# loadings (or weights)
round(pca$rotation, 2)
```

```
##              PC1   PC2   PC3   PC4   PC5   PC6   PC7   PC8
## Population -0.22  0.43 -0.54  0.20 -0.56  0.06  0.23 -0.25
## Income      0.29  0.48 -0.20 -0.65  0.03 -0.39 -0.29  0.05
## Illiteracy -0.46 -0.06 -0.03 -0.07  0.40 -0.58  0.37 -0.39
## Life Exp    0.40  0.04 -0.37  0.46  0.24 -0.35  0.27  0.49
## Murder     -0.44  0.28  0.20 -0.28 -0.01  0.16  0.41  0.64
## HS Grad     0.42  0.35  0.11 -0.08  0.37  0.44  0.49 -0.35
## Frost       0.36 -0.23  0.39 -0.15 -0.58 -0.34  0.44 -0.05
## Area       -0.05  0.57  0.58  0.47 -0.02 -0.24 -0.23 -0.05
```

```r
# weights of PC1
round(pca$rotation[ ,1], 3)
```

```
## Population     Income Illiteracy   Life Exp     Murder    HS Grad      Frost
##     -0.221      0.286     -0.458      0.400     -0.442      0.416      0.360
##       Area
##     -0.046
```

Let's compare all the indices

Experimental comparison

```r
# table with various weighted sums
composites <- data.frame(
  # 1st principal component
  PC1 = pca$x[ ,1],
  # plain index
  Sum1 = var_sum3,
  # average
  Avg1 = rowMeans(state3),
  # random sum
  Rand1 = apply(state3, 1, function(x) sum(rnorm(length(x)) * x)),
  row.names = rownames(state.x77)
)

# squared correlations with observed scaled variables
corr2_composites <- (cor(state3, composites))^2
```

Experimental comparison

```
print(round(corr2_composites, 4), print.gap = 3)

##                 PC1      Sum1      Avg1     Rand1
## Population   0.1798    0.1130    0.1130    0.0158
## Income       0.3017    0.4156    0.4156    0.0042
## Illiteracy   0.7751    0.0318    0.0318    0.0018
## Life Exp     0.5908    0.0620    0.0620    0.0008
## Murder       0.7208    0.0179    0.0179    0.0116
## HS Grad      0.6391    0.2929    0.2929    0.0046
## Frost        0.4788    0.0047    0.0047    0.0452
## Area         0.0079    0.3647    0.3647    0.0181

colSums(corr2_composites)

##       PC1       Sum1       Avg1      Rand1
## 3.6940711 1.3026897 1.3026897 0.1020266
```