PCA

Klaus Smit

June 5, 2020

Artificial Data Generation

Model $z = a_x x + a_y y + 3$

z = Response variable x, y = predictos or featrues coefficients = $a_x = 1 \quad a_y = 1$ Intercept = 3

Original Data Points	x	y	z
P_0	0	0	3
P_1	1	1	5
P_2	2	2	7

```
data = data.frame(x=c(0,1,2),y=c(0,1,2),z=c(3,5,7))
data
```

```
##   x y z
## 1 0 0 3
## 2 1 1 5
## 3 2 2 7
```

Applying PCA through code

```
data_scaled = (data[,1:2]-sapply(data[,1:2],mean)/sapply(data[,1:2],sd))
print(data_scaled)
```

```
##    x  y
## 1 -1 -1
## 2  0  0
## 3  1  1
```

```
PCA <- prcomp(data[,1:2], scale=TRUE)
print(c('mean',PCA$center)) # mean for each predictor
```

```
##              x      y
## "mean"    "1"    "1"
```

```
print(c('sigma',PCA$scale)) # standard deviation for each predictor using N-1
```

```
##                x       y
## "sigma"     "1"     "1"
```

The Math behind scaling the data

The data is scaled through standard the $standard\ score$ (https://en.wikipedia.org/wiki/Standard_score)

$x_{scale} = \frac{x-\mu_x}{\sigma_x}$ $\mu = mean$ and $\sigma = standard\ deviation$

$x_{scale} = \frac{x-1}{1}$

Scaled Data Points	x	y	z
P_0	-1	-1	3
P_1	0	0	5
P_2	1	1	7

Note:
1. That PCA$x is just the scaled data
2. PCA excludes z the response variable

Rotating the data

```
print(as.matrix(data_scaled)%*%PCA$rotation)
```

```
##            PC1           PC2
## [1,] -1.414214 -1.110223e-16
## [2,]  0.000000  0.000000e+00
## [3,]  1.414214  1.110223e-16
```

```
PCA$x
```

```
##            PC1           PC2
## [1,] -1.414214 -1.110223e-16
## [2,]  0.000000  0.000000e+00
## [3,]  1.414214  1.110223e-16
```

data_scaled times the rotation matrix is equal to the matrix containing the principal components PCA$x

Scaled + Rotated Data Points	x	y	z
P_0	-\sqrt{2}	0	3
P_1	0	0	5
P_2	$\sqrt{2}$	0	7