ABOUT INDEX CONTACT

LISTEN DATA

MAKE YOUR DATA TELL A STORY

HOME SAS R PYTHON DATA SCIENCE CREDIT RISK SQL EXCEL SPSS INFOGRAPHICS

SEARCH... GO

Home » Data Science » SAS » Statistics » Principal Component Analysis with SAS

Get Free Email Updates

Follow us on Facebook

PRINCIPAL COMPONENT ANALYSIS WITH SAS

Deepanshu Bhalla 2 Comments Data Science, SAS, Statistics

Principal Component Analysis

Principal components are weighted linear combinations of the variables where the weights are chosen to account for the largest amount of variation in the data. The total number of principal components is the same as the number of input variables. It is based on the correlation or covariance matrix.

> *The purpose of principal component analysis is to reduce the information in*

> *many variables into a set of weighted linear combinations of those variables.*

Assumptions

1. 5+ cases per variables (ideal is 20 per)
2. N > 200
3. Each observed variable should be normally distributed.
4. The relationship between all observed variables should be linear.
5. At least some correlations among the variable

How PCA works

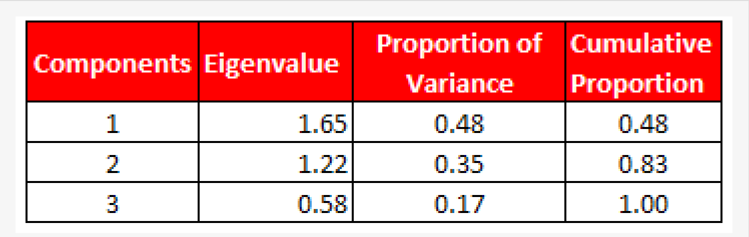
1. **Calculate the covariance matrix of three variables - x, y and z**

	x	y	z
x	1.34	-0.16	0.19
y	-0.16	0.62	-0.13
z	0.19	-0.13	1.49

The **'variance of each variable'** is the diagonal values of the above matrix. It's covariance with itself. The diagonal values are 1.34. 0.62, 1.49. The sum of these values is 3.45 which is **total variation**.

2. Calculate the eigenvectors and eigenvalues of the above covariance matrix

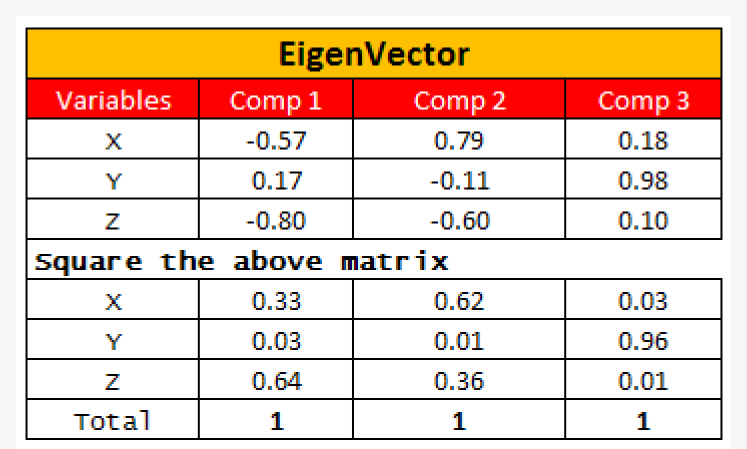
Let A be a matrix, x be a vector.
If Ax=(lambda)*x then x is an eigenvector of A and lambda is an eigenvalue of A



Components	Eigenvalue	Proportion of Variance	Cumulative Proportion
1	1.65	0.48	0.48
2	1.22	0.35	0.83
3	0.58	0.17	1.00

Eigenvalue

The total sum of eigenvalues is 3.45. **The first eigenvalue shows 48% of total variance (= 1.65/3.45) is explained by first component.** Similarly, the second component explains 35% of total variance. The cumulative proportion of variance by first two components is 83%.



EigenVector			
Variables	Comp 1	Comp 2	Comp 3
X	-0.57	0.79	0.18
Y	0.17	-0.11	0.98
Z	-0.80	-0.60	0.10
Square the above matrix			
X	0.33	0.62	0.03
Y	0.03	0.01	0.96
Z	0.64	0.36	0.01
Total	**1**	**1**	**1**

Eigenvector

Eigenvector shows the correlation between component and variable. It explains which variable is highly correlated to a component. The **squared value of eigenvector** explains contribution of variable to a principal component. Hence, the sum of all values of a component is 1.

3. Choosing components with eigenvalue > 1 or cumulative proportion of variance more than 80%.
In this case, we retain first two components.

Terminologies related to PCA

1. **Eigenvalue :** It represents the amount of variance accounted for by a component.
2. **Eigenvector (Loading) :** It represents the weight of the component for each variable (for interpretation of the relative importance of the original variables). In other words, it tells the correlation between a variable and component. It is also called the coefficients of principal component score.
3. **Communality :** It is the sum of the squared eigenvalues for all components for a given variable. It represents proportion of each variable's variance that can be explained by all the components jointly. If the communality for a variable is less than 50%, it is a candidate for exclusion from the analysis because the factor solution contains less that half of the variance

in the original variable, and the explanatory power of that variable might be better represented by the individual variable.

Types of Rotations :

1. **Orthogonal** : Components are uncorrelated, i.e. no correlation between components.
2. **Oblique** : Components are related, with some correlations.

Interpretation of the Principal Components

> *It is the component loading value which tells the correlations between a particular component and a variable.*

Calculation of Principal Component Score

For each case, the component score is computed by multiplying the **case's standardized variable values** by the eigenvectors of each variables in the component (derived from standardized variables).

> *PCA1 = Eigenvector of var1 in PCA1 * (var1) + Eigenvector of var2 in PCA1 * (var2)*

Usage : These component scores could be used either as predictor variables or as criterion variables in subsequent analyses. For example : Three components are selected after running PCA. So, there will be 3 variables for each cases in the output file.

Standardization in PCA

It is important to make sure you **standardize variables** before running PCA. It is because PCA gives more weightage to those variables that have higher variances than to those variables that have very low variances. In effect the results of the analysis will depend on what units of measurement are used to measure each variable. Standardizing raw values makes equal variance so high weight is not assigned to variables having higher variances. By standardization, it means **Z score** i.e. (X - Sample_Mean)/Sample Stdev.

> *With standardization, you may get more components as compared to without standardization.*

Determining the number of components

1. Eigenvalues over 1. Drop all components with eigenvalues under 1.0.
2. Scree-plot: Plots eigenvalues. Where curvature changes.
3. Retain enough components to explain some cumulative total percent of variance, usually 70% to 80%.

Why Eigenvalues > 1

Each observed variable contributes one unit of variance to the total variance in the data set. Any component that displays an eigenvalue greater than 1.0 is accounting for a greater amount of variance than was contributed by one variable. Such a component is therefore accounting for a meaningful amount of variance and is worthy of being retained. On the other hand, a component with an eigenvalue less than 1.0 is accounting for less variance than had been contributed by one variable.

Multicollinearity and PCA

If you run a principal component analysis on a set of 5 variables and observe that the first component explains 85% of the variance. It means the variables are highly correlated to each other. In other words, variables are faced with multicollinearity.

Let's take a sample correlation matrix -

```
 k1 k2
k1 1 0.6890285
k2 0.6890285 1
```

Eigenvalues

```
[1] 1.6890285 0.3109715
```

% of variance explained by first component is 84.45%.

R Code

```
seed(1)
k1 = sample(100:1000,1000,
replace=TRUE)
k2 = sample(10:1010,1000) + k1 - 10**2
* runif(1000)
X=cbind(k1,k2)
c = cor(X)
eigen(c)$values[1]/
sum(eigen(c)$values)
```

Problems with Principal Component Analysis

1. Each principal component involves all the input variables. The coefficients of the principal components—the eigenvectors—are usually non-zero for all the original input variables. This means that, if you use any principal components in the analysis—even one, you must retain all the original inputs. Also, the weights can be challenging to interpret.
2. A variable can have high correlation with two components. It will lead to an ambiguous interpretation in analysis.

Uses of Principal Components

> *The principal components can be used in place of the original variables in the analysis.*

SAS Code : Principal Component Analysis

```
PROC PRINCOMP DATA= readin
OUT=outdata OUTSTAT = stats;
VAR x1 x2 x3;
RUN;
```

Default Method : Based on Correlation Matrix. It should be used when variables are of different type. No need to standardize variables.

Options used in PROC PRINCOMP

COV : You can specify COV option to calculate principal based on covariance matrix. It should be used when variables are of same type.

> *Using the correlation matrix is equivalent to using the covariance matrix of standardized observations.*

N= : Number of Components

OUT= : It creates an output SAS data set that contains all the original data as well as the **principal component scores**.

OUTSTAT= : It creates an output SAS data set that contains means, standard deviations, number of observations, correlations or covariances, eigenvalues, and eigenvectors.

Run correlation matrix

```
PROC CORR DATA=pcstuff;
VAR x1 x2 x3;
WITH prin1 prin2 prin3;
run;
```

PCA SAS Macro

```
%macro Principal(Input, vars, Method, p, scoreout, outdata);
/* Reducing a set of variables (vars) using PCA, by keeping fraction p (p<=1) of the variance.
The output is stored in outdata and the model is stored in scoreout. */
/* First run PRINCOMP to get All the eigenvalues */
proc princomp data=&Input &Method outstat=Temp_eigen noprint;
var &vars;
run;
/* Then select only the top fraction p of
```

```
the variance */
data Tempcov1;
set temp_Eigen;
if _Type_ ne 'EIGENVAL' then delete;
drop _NAME_;
run;
proc transpose data=Tempcov1
out=TempCovT ;
run;
data TempCov2;
set TempCovT;
retain SumEigen 0;
SumEigen=SumEigen+COl1;
run;
proc sql noprint;
select max(SumEigen) into :SEigen from
TempCov2;
quit;
data TempCov3;
set TempCov2;
IEigen=_N_;
PEigen = SumEigen/&SEigen;
run;

/* Count the number of eigenvalues
needed to reach p */
proc sql noprint;
select count(*) into :Nh from Tempcov3
```

```
where PEigen >= &P;
select count(*) into :NN from
TempCov3;
%let N=%eval(&NN-&Nh+1);
quit;

/* Delete from the DSEigen all the rows
above the needed N eigenvectors */
data &scoreout;
set Temp_Eigen;
run;
proc sql noprint;
%do i=%eval(&N+1) %to &NN;
delete from &scoreout where _NAME_ =
"Prin&i";
%end;
quit;
/* And score */
proc score data=&Input
Score=&scoreout Out=&outdata;
Var &vars;
run;
/* Finally, clean workspace */
proc datasets library=work nodetails;
delete Tempcov1 Tempcov2 Tempcov3
Temp_eigen Tempcovt;
run;
```

```
quit;
%mend;
```

Convert PDFs into file formats
like Word, Excel and Powerpoint.

SAS Tutorials : [100 Free SAS Tutorials]

Statistics Tutorials : [50 Statistics Tutorials]

✉ SUBSCRIBE TO GET EMAIL UPDATES!

Related Posts

- Case Study : Sentiment analysis using Python
- 15 Types of Regression in Data Science
- Complete Guide to Marketing Mix Modeling
- Gini, Cumulative Accuracy Profile, AUC
- Precision Recall Curve Simplified
- Identify Person, Place and Organisation in content using Python

About Author:

Deepanshu founded ListenData with a simple objective - Make analytics easy to understand and follow. He has over 8 years of experience in data science. During his tenure, he has worked with global clients in various domains like Banking, Insurance, Telecom and Human Resource.

While I love having friends who agree, I only learn from those who don't

Let's Get Connected: Email | LinkedIn

2 Responses to "Principal Component Analysis with SAS"

Unknown 9 September 2015 at 00:04

Its more useful to publish your code along with your dataset

Reply

HB 5 July 2018 at 07:09

Hi Deepanshu, I am using varclus and PCA in SAS to reduce number of variables. Is there any take on which technique is better than other and why? Also, is there anything else I can use?

Reply

Enter your comment...

Comment as: Jack Mardekia ▼ Sign out

Publish Preview □ Notify me

← PREV NEXT →

Copyright 2019 ListenData