How do I perform Principal Components Analysis in SAS?

Authored by

stats writer

June 23, 2024

Principal Components Analysis (PCA) is a statistical technique used for data exploration and dimension reduction. It involves transforming a set of correlated variables into a smaller set of uncorrelated variables called principal components. This allows for a more efficient and meaningful representation of the data. In SAS, PCA can be performed by using the PROC PRINCOMP procedure, which allows for customization of the analysis through various options such as choosing the number of components and handling missing values. It also provides output that can be used for further analysis and interpretation. By following the appropriate steps and utilizing the available options in SAS, one can effectively perform PCA and gain valuable insights from the data.

Perform Principal Components Analysis in SAS

Principal components analysis (PCA) is an machine learning technique that seeks to find principal components - linear combinations of the predictor variables - that explain a large portion of the variation in a dataset.

The easiest way to perform PCA in SAS is to use the PROC PRINCOMP statement, which uses the following basic syntax:

```
proc princompdata=my_data out=out_data outstat=stats;
var var1 var2 var3;
run;
```

Here is what each statement does:

data: The name of the dataset to use for PCAout: The name of the dataset to create that contains all original data along with the principal component scoresoutstat: Specifies that a dataset should be created that contains the means, standard deviations, correlation coefficients, eigenvalues, and eigenvectors.var: The variables to use for PCA from the input dataset.

The following step-by-step example shows how to use the PROC PRINCOMP statement in practice to perform principal components analysis in SAS.

Step 1: Create Dataset

Suppose we have the following dataset that contains various information about 20 basketball players:

```
/*create dataset*/
data my_data;
input points assists rebounds;
datalines;
22 8 4
29 7 3
10 4 12
5 5 15
```

```
35 6 2
8 3 10
10 4 8
8 4 3
2 5 17
4 5 19
9 9 4
7 6 4
31 5 3
4 6 13
5 7 8
8 8 4
10 4 8
20 4 6
25 8 8
18 8 3
;
run;

/*view dataset*/
proc printdata=my_data;
```

Obs	points	assists	rebounds
1	22	8	4
2	29	7	3
3	10	4	12
4	5	5	15
5	35	6	2
6	8	3	10
7	10	4	8
8	8	4	3
9	2	5	17
10	4	5	19
11	9	9	4
12	7	6	4
13	31	5	3
14	4	6	13
15	5	7	8
16	8	8	4
17	10	4	8
18	20	4	6
19	25	8	8
20	18	8	3

Step 2: Perform Principal Components Analysis

We can use the PROC PRINCOMP statement to perform principal components analysis using the points, assists and rebounds variables in the dataset:

```
/*perform principal components analysis*/
proc princompdata=my_data out=out_data
outstat=stats;
```

var points assists rebounds;
run;

The first portion of the output shows various descriptive statistics including the mean and standard deviations of each input variable, a correlation matrix, and the values for the eigenvalues and eigenvectors:

The PRINCOMP Procedure

Observations	20
Variables	3

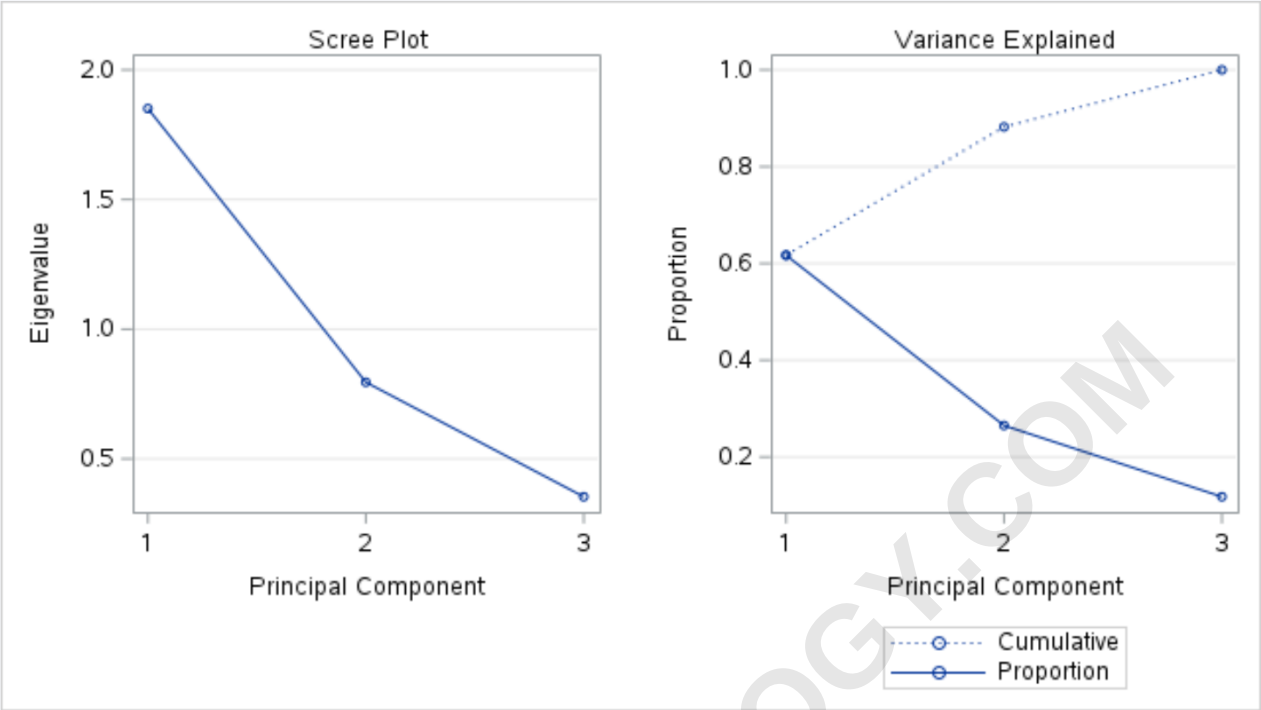
Simple Statistics			
	points	assists	rebounds
Mean	13.50000000	5.800000000	7.700000000
StD	10.06034424	1.765159900	5.120649630

Correlation Matrix			
	points	assists	rebounds
points	1.0000	0.2341	-.6232
assists	0.2341	1.0000	-.3855
rebounds	-.6232	-.3855	1.0000

Eigenvalues of the Correlation Matrix				
	Eigenvalue	Difference	Proportion	Cumulative
1	1.85089366	1.05544499	0.6170	0.6170
2	0.79544867	0.44179099	0.2651	0.8821
3	0.35365768		0.1179	1.0000

Eigenvectors			
	Prin1	Prin2	Prin3
points	0.603439	-.478471	0.637908
assists	0.460853	0.862106	0.210683
rebounds	-.650750	0.166848	0.740733

The next portion of the output displays a Scree Plot and a Variance Explained plot:



The table in the output titled Eigenvalues of the Correlation Matrix allow us to see exactly what percentage of total variation is explained by each principal component:

The first principal component explains 61.7% of the total variation in the dataset.The second principal component explains 26.51% of the total variation in the dataset.The third principal component explains 11.79% of the total variation in the dataset.

Notice that all of the percentages sum to 100%.

The plot titled Variance Explained then allows us to visualize these values.

The x-axis displays the principal component and the y-axis displays the percentage of total variance explained by each individual principal component.

Step 3: Create Biplot to Visualize Results

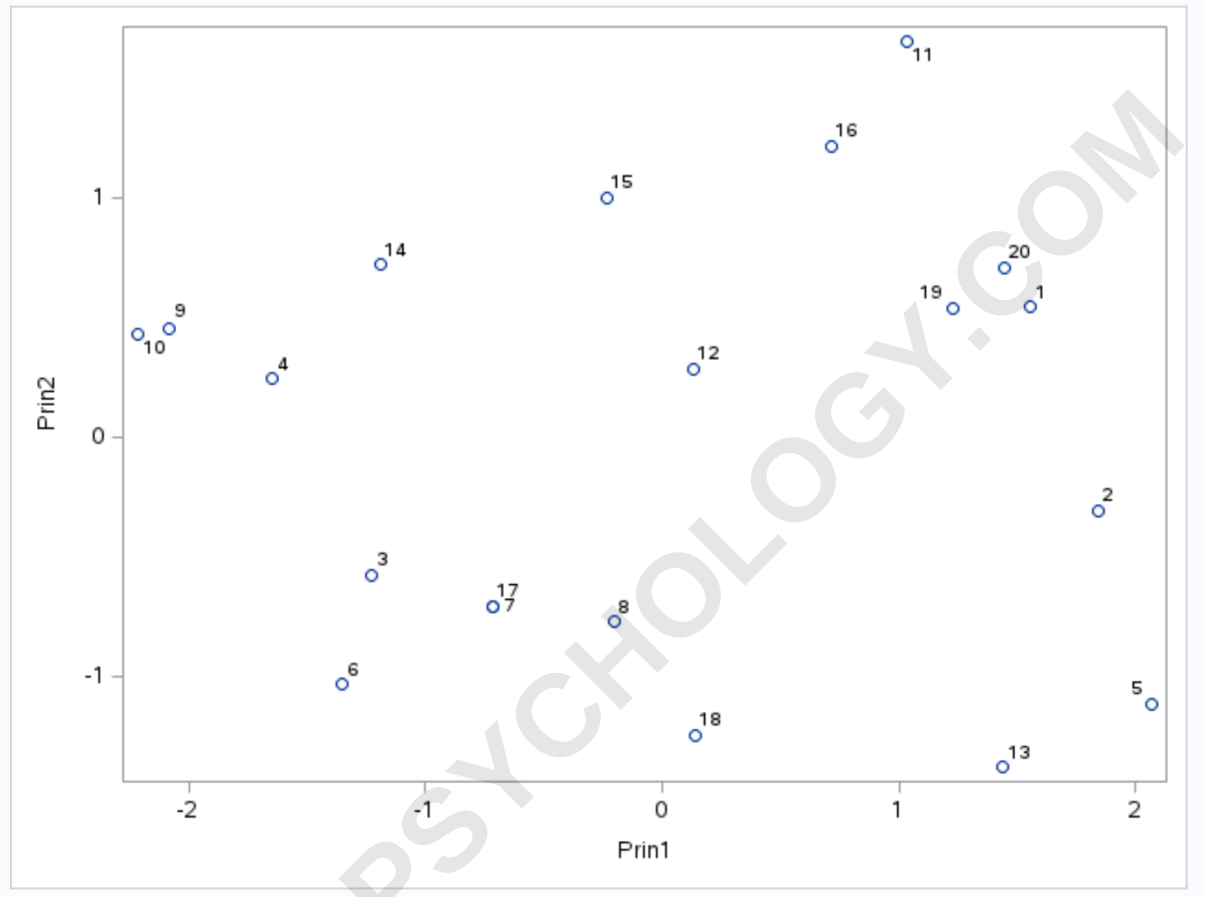
To visualize the results of PCA for a given dataset we can create a biplot, which is a plot that displays every observation in a dataset on a plane that is formed by the first two principal components.

We can use the following syntax in SAS to create a biplot:

```
/*create dataset with column called obs to represent row numbers of original data*/data biplot_data;
set out_data;
obs=_n_;
run;

/*create biplot using values from first two principal components*/
proc sgplotdata=biplot_data;
```

```
scatterx=Prin1 y=Prin2 / datalabel=obs;
run;
```



The x-axis displays the first principal component, the y-axis displays the second principal component, and the individual from the dataset are shown inside the plot as tiny circles.

Observations that are next to each other on the plot have similar values across the three variables of points, assists and rebounds.

For example, on the far left side of the plot we can see that observations #9 and #10 are extremely close to each other.

If we refer to the original dataset, we can see the following values for these observations:

Observation #9: 2 points, 5 assists, 17 reboundsObservation #10: 4 points, 5 assists, 19 rebounds

The values are similar across each of the three variables, which explains why these observations are so close to each other on the biplot.

We also saw from the table in the output titled Eigenvalues of the Correlation Matrix that the first two principal components account for 88.21% of the total variation in the dataset.

Since this percentage is so high, it's valid to analyze which observations in the biplot are near each other because the two principal components that make up the biplot account for almost all of the variation in the dataset.

The following tutorials explain how to perform other common tasks in SAS: