12. Principal Components Analysis

LPO.8800: Statistical Methods in Education Research

Sean P. Corcoran

What is principal components analysis (PCA)?

PCA is best thought of as a technique for **data reduction**.

- For example, suppose you have a dataset of variables $X_1, ..., X_k$
- Could be multiple student test scores, measures of school performance, or measures of neighborhood disadvantage. (Variables should be at least interval-measured).
- PCA is method of creating a smaller set of variables (scores) that contain much of the same information as the original set.
- These scores can be analyzed on their own, or used in place of the larger set of variables.

What is principal components analysis (PCA)?

Principal components are *linear combinations* of the X chosen in a specific way:

- They are mutually **uncorrelated**.
- They *sequentially* contain as much information from the original variables as possible.

Think of each component as a "recipe": a set of weights applied to the original Xs to produce a new score.

- With k variables there will generally be k principal components, unless $n < k$, in which case there are only n. (Exception: if there is perfect collinearity between any variables, there will be fewer than k components).

What is PCA used for?

It is most often used when many variables of interest in the dataset are highly correlated with each other:

- Creating **indices** from multiple measures: e.g., for student academic aptitude, school performance, student engagement, neighborhood disadvantage, SES.
- Easier data visualization or cluster analysis (identifying units with similar "profiles" as defined by the PC).
- Addressing **multicollinearity** in regression models

PCA can also be useful in identifying **latent constructs**.

First principal component (with 2 variables)

The first principal component (PC_1) is the linear combination of the X with the maximum variance (i.e., capturing as much of the underlying variation in X as possible). With two variables X_1 and X_2:

$$PC_1 = w_1 X_1 + w_2 X_2$$

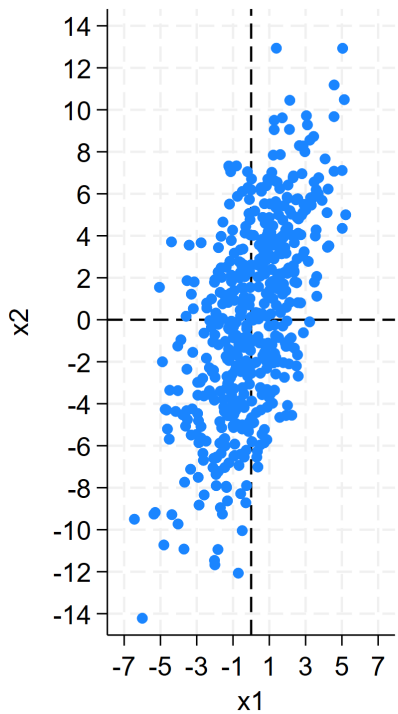
where w_1 and w_2 are the principal component weights (or **loadings**).

- Simulated data: X_1 and X_2 are mean zero and bivariate normal.
- $\sigma_1^2 = 4$, $\sigma_2^2 = 20$, $\sigma_{12} = 5$. In covariance matrix form:

$$\Sigma = \begin{bmatrix} 4 & 5 \\ 5 & 20 \end{bmatrix}$$

Simulated data example

Scatter plot between x_2 and x_1:



Simulated data example

Goal: Find w_1, w_2 that **maximize the variance** in PC_1. In this example, $w_1 = 0.286$ and $w_2 = 0.958$ (see next slide):

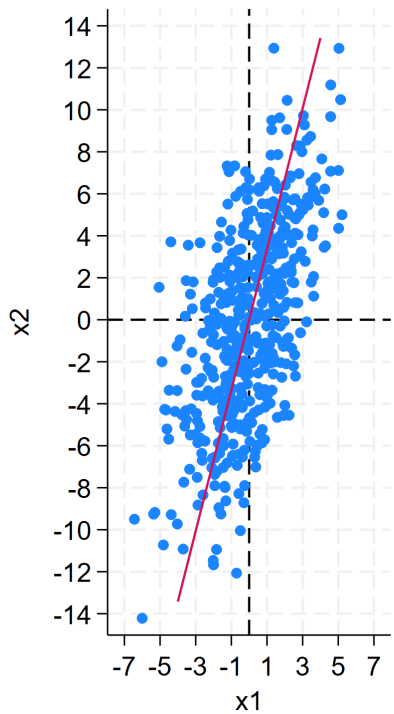
$$PC_1 = 0.286X_1 + 0.958X_2$$

Note: one could maximize variance of PC_1 by simply increasing w_1 and w_2 indefinitely. Thus, it is standard practice to constrain the weights, for example such that $w_1^2 + w_2^2 = 1$. That is the case here.

"Recipe" interpretation: PC_1 involves 0.958 units of X_2 and 0.286 units of X_1. X_2 is more important to PC_1 in this example since it has greater variance.

Simulated data example

Scatter plot and PC_1:



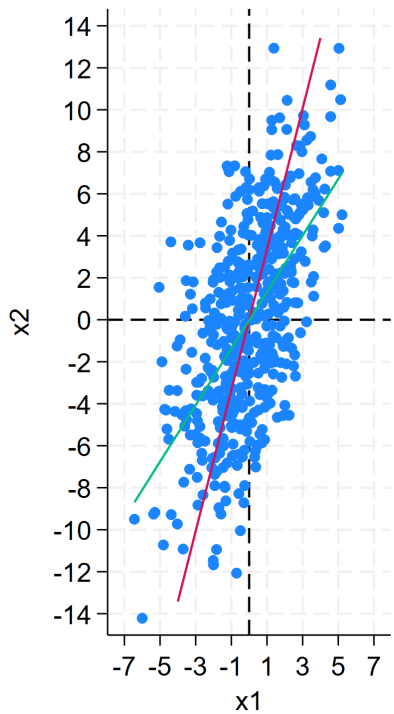
First principal component (with 2 variables)

Note the "best fit line" for PC_1 is not the same thing as the linear regression of x_2 on x_1 (see next slide). Why?

- PC_1 minimizes the perpendicular distances from the data points to the line—it is equally interested in x_1 and x_2.
- Regression finds the line that best predicts x_2 given x_1 (minimizes vertical distances from the point to the line)—the focus is on x_2.
- PC_1 is trying to maximize variance; if the variances of the two variables are different, the line will tilt toward the one with the greater variance.
- The lines will coincide only in special cases: if x_2 is an exact linear function of x_1; or equal variances and perfect correlation.

Simulated data example

PC_1 versus linear regression of X_2 on X_1:



Second principal component (with 2 variables)

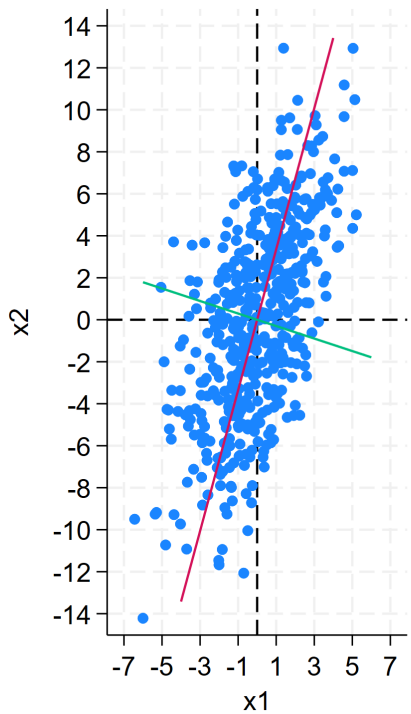
The second principal component (PC_2) is the linear combination of the X that is uncorrelated with PC_1 and captures as much of the underlying variation in X as possible, given PC_1. In this example, $w_1 = 0.958$ and $w_2 = -0.286$ (see next slide).

$$PC_2 = 0.958X_1 - 0.286X_2$$

- The sum of its squared weights is again equal to 1.
- In the case of two variables, PC2 simply captures residual variation.

Simulated data example

PC_1 and PC_2:



Principal components with k variables

The principal components of k variables $X_1, ..., X_k$ are linear combinations of those variables that:

- Are mutually uncorrelated.
- Have squared weights that sum to 1.
- Maximize the variance of the linear combination, controlling for the previous principal components.

Principal components with k variables

Note the sum of the variances of the PCs equals the sum of the variances of the Xs:

$$\sum_{j=1}^{k} Var(PC_j) = \sum_{j=1}^{k} Var(X_j)$$

- The PCs are uncorrelated, so one can calculate the fraction of the total variance "explained" by each PC (like an R^2).
- Important: PCA is not scale invariant. Variables measured in different units can affect weights since variables with higher variance tend to get higher weight. Solution: convert variables to standardized (z) scores. Stata will do this by default.

Stata pca command

To perform a principal components analysis in Stata, use the `pca` command:

`pca` *`varnames, options`*

Output includes:

- Principal component weights (**"eigenvectors"**)
- Variances of each principal component (**"eigenvalues"**)—these add up to the total variance of the Xs (the "trace")
- Proportion of the total variance "explained" by each principal component.

Note: when all variables are in z-score form (the default) the total variance is the number of variables ($1 + 1 + ... = k$).

Example: Duflo, Dupas, and Kremer (2011)

This was a RCT of primary school tracking in Kenya. Their dataset includes endline test scores for 5,795 first grade students. There is a *totalscore* variable but also 7 subtest scores (words, sentences, letters, spelling, addition, subtraction, multiplication). It is possible we could learn more from a PCA of these 7 scores than simply relying on the total score.

```
. corr wordscore sentscore letterscore spellscore additions_score ///
>         substractions_score multiplications_score
(obs=5,789)
```

	wordsc~e	sentsc~e	letter~e	spells~e	additi~e	substr~e	multip~e
wordscore	1.0000						
sentscore	0.5510	1.0000					
letterscore	0.6230	0.4546	1.0000				
spellscore	0.7983	0.5479	0.7081	1.0000			
additions_~e	0.4886	0.3216	0.5318	0.5215	1.0000		
substracti~e	0.4085	0.2563	0.4764	0.4486	0.7038	1.0000	
multiplica~e	0.4131	0.2827	0.4285	0.4506	0.5141	0.5009	1.0000

Example: Duflo, Dupas, and Kremer (2011)

Principal components/correlation — Number of obs = 5,789; Number of comp. = 7; Trace = 7; Rho = 1.0000

Rotation: (unrotated = principal)

Component	Eigenvalue	Difference	Proportion	Cumulative
Comp1	4.01783	2.97255	0.5740	0.5740
Comp2	1.04528	.48359	0.1493	0.7233
Comp3	.561694	.036229	0.0802	0.8035
Comp4	.525465	.152919	0.0751	0.8786
Comp5	.372546	.0829405	0.0532	0.9318
Comp6	.289606	.10203	0.0414	0.9732
Comp7	.187576	.	0.0268	1.0000

Principal components (eigenvectors)

Variable	Comp1	Comp2	Comp3	Comp4	Comp5	Comp6	Comp7	Unexplained
wordscore	0.4104	-0.3221	-0.0829	-0.2202	-0.5641	0.0581	0.5923	0
sentscore	0.3160	-0.4940	0.3257	0.7197	0.1786	0.0077	-0.0052	0
letterscore	0.4040	-0.1265	-0.2761	-0.3277	0.7665	-0.0143	0.2222	0
spellscore	0.4295	-0.2767	-0.1149	-0.2812	-0.2081	0.0690	-0.7737	0
additions_~e	0.3848	0.4079	-0.2415	0.2576	-0.1192	-0.7390	-0.0239	0
substracti~e	0.3548	0.5154	-0.2484	0.3149	-0.0463	0.6674	0.0070	0
multiplica~e	0.3317	0.3572	0.8231	-0.2856	0.0516	-0.0078	0.0234	0

Example: Duflo, Dupas, and Kremer (2011)

Things to note in the above output:

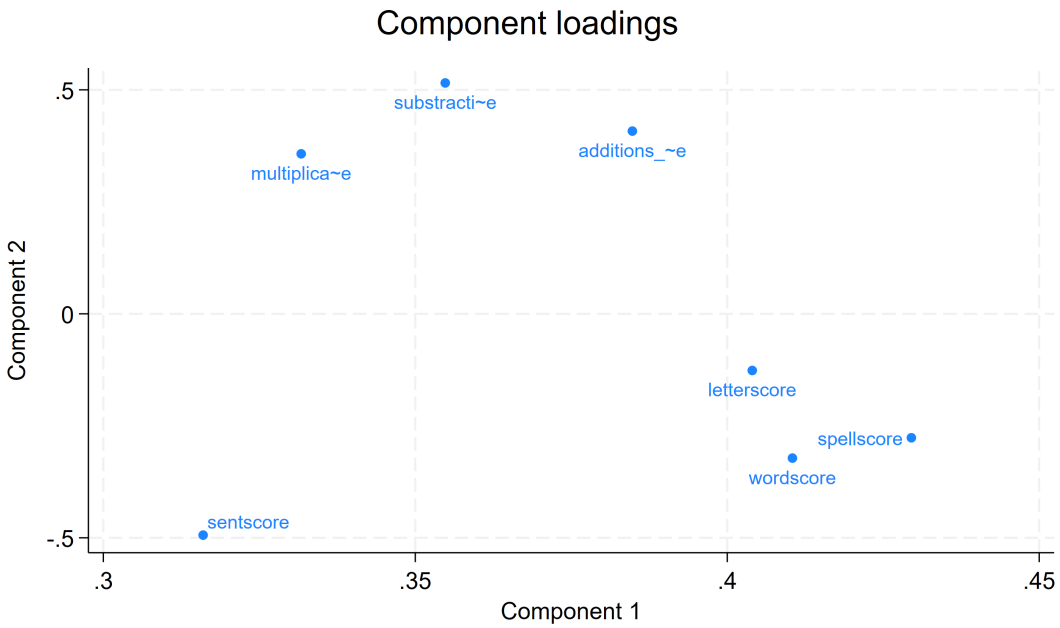
- The sum of the squared weights for each component equals 1.
- The sum of the eigenvalues equals 7 (the number of variables).

Interpretation:

- Over 80% of the total variance is accounted for by PC_1, PC_2, and PC_3 (72% by the first two components)
- PC_1: weights are all positive and roughly the same, implying a simple average of these 7 scores.
- PC_2: positive weights on math scores and negative weights on literacy scores. Similar to a difference between the math and literacy scores.

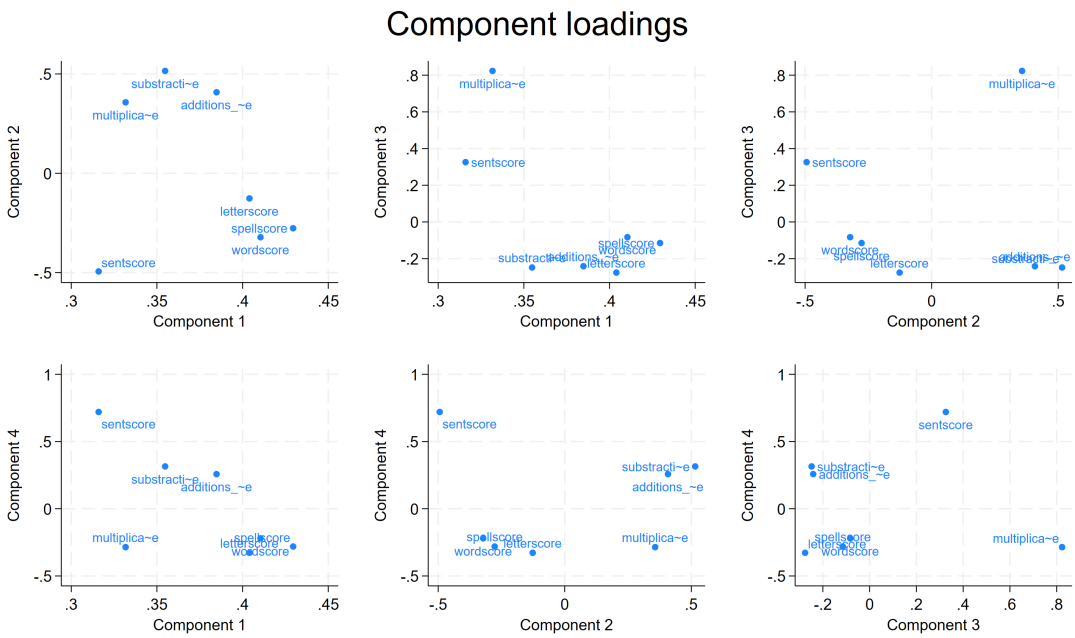
Loading plot (1)

Loading plot (PC_2 vs. PC_1): how the variables load in component space



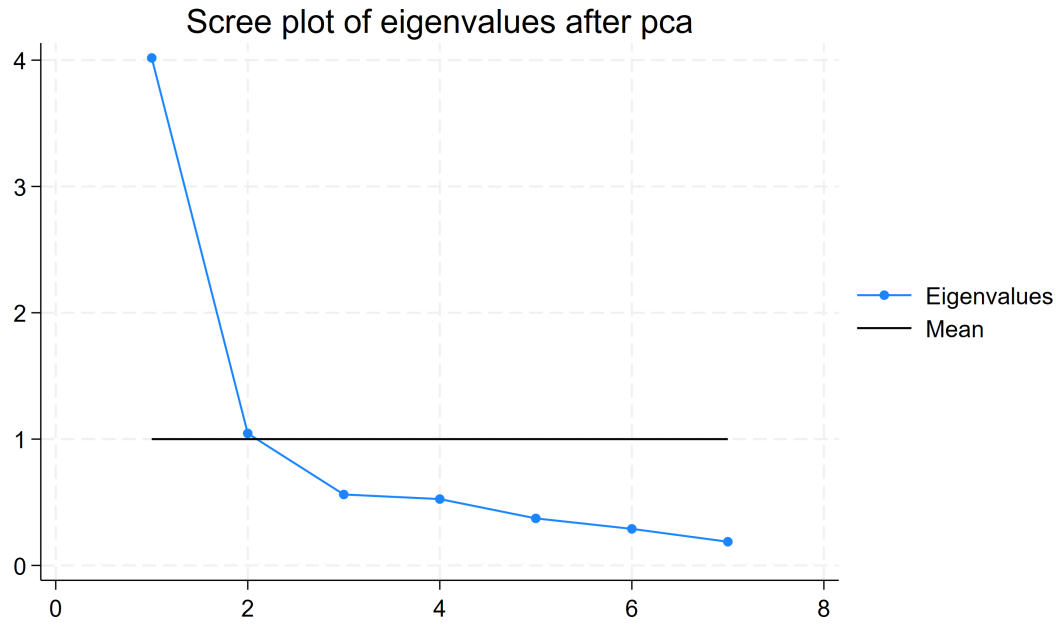
Loading plot (2)

Loading plot (all PCs):



Scree plot

Scree plot:



Scree plot

The **scree plot** shows the eigenvalues by component. (Sometimes this plot is shown using proportions of the total).

- Sometimes analysts use the threshold of 1—the mean eigenvalue—to decide which principal components to focus on.
- Alternatively, can take components up to the point where there is significant diminishing returns.

Calculating principal component scores

Given the principal component weights, you can calculate the scores and add them to your dataset:

`predict` *pc**`, score` *options*

```
. summ pc1-pc7, sep(0)

    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
         pc1 |      5,789   -3.24e-09    2.004453  -2.693618   7.201586
         pc2 |      5,789    1.77e-11    1.022391     -4.371   3.601964
         pc3 |      5,789    9.72e-11    .7494623  -2.430281    3.34169
         pc4 |      5,789   -6.75e-11    .7248895  -2.190043    3.32636
         pc5 |      5,789   -2.92e-12    .6103655  -2.636242   2.570263
         pc6 |      5,789   -3.13e-10    .5381502  -2.248965   1.990383
         pc7 |      5,789   -1.71e-10    .4331005  -2.268605   2.190427

. tabstat pc1-pc7, stat(var) col(stat)

    Variable |  Variance
-------------+----------
         pc1 |   4.01783
         pc2 |  1.045284
         pc3 |  .5616937
         pc4 |  .5254648
         pc5 |  .3725461
         pc6 |  .2896056
         pc7 |   .187576
-------------------------
```


Other notes on the Stata pca command

- You can choose to only report a limited number of components, e.g., the first three: `pca` *varnames*`, components(3)`.
- You can choose to only report components with a minimum eigenvalue, e.g. 1: `pca` *varnames*`, mineigen(1)`
- You can show blank spaces for loadings below a certain threshold (#): `pca` *varnames*`, blanks(#)`. This draws attention to groups of variables important for a factor.
- PCA can be used descriptively or inferentially. You may wish to view your principal component weights and eignenvalues as estimates of population values. Assuming normality, you can compute standard errors, confidence intervals, test hypotheses, etc: `pca` *varnames*`, vce(normal)`.

Other topics

- Another example: hearing loss (from Stata manual)
- Rotation
- PCA versus regression (more)
- PCA versus factor analysis