Principal Components Analysis: Using Math to Extract Underlying Structures of Iraqi Migration Survey Data

Jackie Lindstrom
Adviser: N. Justice

Pacific Lutheran University, 2023

1 Abstract

Principal Components Analysis (PCA) is a mathematical tool used to reduce the dimension of a data set in the hope of obtaining useful and understandable takeaways. This analysis can be helpful in identifying patterns and performing dimension reduction. PCA uses the vectors in the data table, each vector representing a different variable, and finds a set of new, orthogonal vectors. The original vectors can be expressed as linear combinations of the new principal components. The primary principal component captures the largest variance, or spread, and therefore "explains" the largest amount of information from the original data table. The second principal component will be orthogonal to the first, and usually contains less information.

After discussing the background of PCA from a linear algebra perspective, this paper discusses the process of preparing and conducting PCA on results of a survey sourced from the International Organization for Migration. The survey was conducted in Iraq asking internally displaced persons about their access to certain needs, such as distance from clinics, access to clean water, etc. Results could help the International Organization for Migration detect patterns and visualize underlying structures in the needs of displaced persons. Additionally, PCA could be used to help improve the survey design, as results can suggest where to reduce redundancy in questions or where to add additional questions to better capture the experiences and needs of this population.

2 Introduction/ Context

Thanks to a Wang Center grant, I was able to travel to London and work with the International Organization for Migration this past January 2023. Because of my interest in serving displaced people, I wanted to use my math capstone project to help this organization. We settled on a data set compiled from a phone survey performed in Iraq which asked internally displaced persons about their access to different needs and recorded their migration status information and socio-demographic indicators. An internally displaced person is someone who is forced to leave their home, but does not cross an international border, but rather remains in their home country. The data set is large and composed of quantitative and qualitative variables.

In this capstone report, I will use PCA to look for patterns and relationships among the data. I will begin by reviewing the linear algebra behind the mathematics used to perform this analysis. Then I will discuss the singular value decomposition in relation to PCA. I will then apply PCA to the Iraqui migration data set using R statistical software (Version 4.2.1).

3 Basic Definitions

Other definitions will be offered throughout the paper, but the most basic definitions are shown here.

Definition 3.1 (Linear Combination) *A linear combination is any sum of vectors multiplied by constants, such as*

$$\alpha_1\mathbf{u}_1 + \alpha_2\mathbf{u}_2 + \alpha_3\mathbf{u}_3 + \alpha_m\mathbf{u}_m = \sum_{j=1}^{m} \alpha_j \mathbf{u}_j \tag{1}$$

where $\boldsymbol{u}_i$ are vectors and α_i are constants. [2]

Definition 3.2 (Linear Transformation) *A mapping f from $\mathbb{R}^n$ to $\mathbb{R}^m$ is* ***linear*** *if $f(a\mathbf{x} + b\mathbf{y}) = af(\mathbf{x}) + bf(\mathbf{y})$ for all vectors $\mathbf{x}$ and $\mathbf{y}$ in $\mathbb{R}^n$ and for all scalars a and b. [2]*

We often denote a linear transformation with matrix multiplication, shown by $T(\vec{x}) = A\vec{x}$.

Definition 3.3 (Transpose of a Matrix) *The transpose of a matrix, denoted superscript T, turns a vertical array into a horizontal one and a horizontal array into a vertical one. [2]*

Definition 3.4 (Symmetric Matrix) *A matrix A such that $A = A^T$ is said to be symmetric, where A^T refers to the transpose of the matrix. [2]*

4 Eigenvalues and Eigenvectors

Definition 4.1 *Let* ***A*** *be any square matrix, real or complex. A number is an* ***eigenvalue*** *of* ***A*** *if the equation*

$$A\vec{v} = \lambda\vec{v} \tag{2}$$

is true for some nonzero vector $\vec{v}$. The vector $\vec{v}$ is an eigenvector associated with the eigenvalue λ. [2]

The left hand side of Equation 2 is matrix vector multiplication, and the right hand side is scalar vector multiplication. Therefore, finding the eigenvalues and eigenvectors for a matrix A involves finding values for $\vec{v}$ and λ that satisfy this equation.

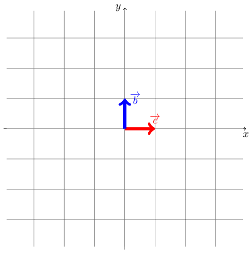
Overall, a vector $\vec{v}$ is an eigenvector of the matrix A, if when multiplied by A, the length of the vector $\vec{v}$ is affected by some scalar λ, but the direction of the vector remains unchanged, or is in the exact opposite direction if λ is negative (but in the context of this analysis, λ is always positive). Another way of phrasing this is that eigenvectors remain on their span when multiplied by matrix A, and the eigenvalue is the factor by which eigenvector $\vec{v}$ scales.

4.1 Visualizing Eigenvectors and Eigenvalues

The following is an example to help illustrate and visualize eigenvalues and eigenvectors. Let A be a square matrix that represents some linear transformation.

$$A = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix}$$

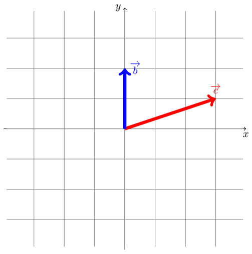
The vectors $\vec{b}$ and $\vec{c}$ are plotted below. Each has a length of 1.



The linear transformation of the vectors by matrix A can be shown with matrix multiplication. For example, the transformation of vector $\vec{b}$ is shown below.

$$A(\overrightarrow{b}) = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix} * \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} (3*0) + (0*1) \\ (1*0) + (2*1) \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

The next graph below shows the result of the transformation by matrix A on the vectors, $A(\vec{b})$ and $A(\vec{c})$



Here, $\overrightarrow{b}$ is an eigenvector of A because it remains on its span (the y-axis) when transformed by A. However, $\overrightarrow{c}$ is not an eigenvector of A because it gets knocked off its span (the x-axis) when transformed by A. The vector $\vec{b}$ is scaled by 2 when transformed by matrix A. Therefore, the eigenvalue associated with eigenvector $\vec{b}$ is 2.

Eigenvector $\vec{b}$, with eigenvalue of 2, is one eigenvector of A, but not the only eigenvector of A. Any other vector that remains on its span when transformed by A is also an eigenvector. The number of linearly independent eigenvectors cannot exceed the dimension of the matrix A. There is the possibility of a second eigenvector for this matrix A, since A has a dimension of 2. The second eigenvector will be revealed and discussed further on in this paper.

4.2 Normalizing Eigenvectors

Often in PCA, eigenvectors are normalized.

Definition 4.2 (Normalize) *To transform a vector so that the direction stays the same, but the length equals 1. To do so, divide the vector by its length. We end up with a unit vector.*

If a vector is normalized, then the transpose of the vector times itself is 1.

$$\mathbf{u}^T\mathbf{u} = 1$$

An example of this is shown below, with the vector $\vec{p}$

$$\vec{p} = [3, 4]$$

The length of $\vec{p}$ is

$$\sqrt{3^2 + 4^2} = 5$$

To normalize $\vec{p}$, we divide the vector by the length 5, to get the resultant vector $\vec{n}$

$$\vec{n} = [\frac{3}{5}, \frac{4}{5}]$$

Now we can see that the normalized vector $\vec{n}$ is a unit vector, and therefore has a length of 1.

5 Eigen-decomposition

Principal components analysis builds on factoring matrices. Before looking at the specific factorization used in PCA, we will examine the general form of factoring matrices.

5.1 Factoring Matrices

Whenever there is a matrix written in the form

$$A = BC$$

we have a factorization of a matrix into the product of the two matrices B and C. [2]

For example, some 3x3 matrix can be factored into two matrices, a 3x2 matrix and a 2x3 matrix.

$$\begin{bmatrix} . & . & . \\ . & . & . \\ . & . & . \end{bmatrix} = \begin{bmatrix} . & . \\ . & . \\ . & . \end{bmatrix} \begin{bmatrix} . & . & . \\ . & . & . \end{bmatrix}$$

Factorization can be useful to visualize and extract information (such as the eigenvalues or eigenvectors) from the matrix.

Another way of thinking about factoring matrices involves delving into what we mean by matrix multiplication. If matrix A performs some transformation, separating A into its factors is breaking A into three matrices, each that perform their own transformation. However, the overall transformation of the three matrices is the same as what A does. A is the composition of the three matrices.

5.2 Eigen-decomposition

One form of factorization is the eigen- decomposition.

Recall Equation 2

$$A\vec{v} = \lambda\vec{v}$$

Take U to be the matrix where each column is an eigenvector of A. Take Λ to be a diagonal matrix that stores the eigenvalues of A. Therefore, we can rewrite Equation 2 as:

$$AU = \Lambda U$$

Thinking in terms of a change of basis, we can write A as

$$A = U\Lambda U^{-1}$$

where U^{-1} is the inverse matrix of U.

Therefore, the eigen-decomposition is a way of rewriting matrix A in a form that clearly shows the eigenvectors and eigenvalues.

5.2.1 Eigen-decomposition Example

Recall matrix A from section 4. We found that [0,1] was an eigenvector for matrix A with an associated eigenvalue of 2. The matrix A can be factored into the following form:

$$A = U\Lambda U^{-1}$$

Inputting the values for A and actually factoring this matrix, we end up with the product of three square 2x2 matrices.

$$A = \begin{bmatrix} 3 & 0 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} * \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} * \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}$$

Eigenvectors: Since U is the matrix where each column is an eigenvector of A, the eigenvectors are easily identifiable as [1,1] and [0,1]. Here we have revealed the second eigenvector of A.

Eigenvalues: Since Λ is the diagonal matrix that stores the eigenvalues of A, the eigenvalues are easily identifiable as 3 and 2. The eigenvalue 3 is associated with eigenvector [1,1], and the eigenvalue 2 is associated with eigenvector [0,1].

Therefore, the eigen-decomposition is a way of rewriting matrix A in a form that clearly shows what the eigenvectors and eigenvalues are.

We can also write the factorization in a different order and put the largest eigenvalue in the first column of the Λ matrix. This is helpful because it is easier to see which are the largest eigenvalues if the matrices under examination are very large. In PCA, the largest eigenvalues are the most useful to look at.

6 Singular Value Decomposition

Theorem 1 *The singular value decomposition (SVD) of a matrix is a generalization of the eigen-decomposition. [1].*

Any mxn matrix can be put into the factored form ***PDQ****, where* ***P*** *and* ***Q*** *are unitary and* ***D*** *is diagonal. [2]*

Given matrix A, the SVD is:

$$A = PDQ$$

P: A matrix of the left singular vectors of A

D: A diagonal matrix with the singular values on its diagonal

Q: A matrix of the right singular vectors of A

When this factorization is written in matrix form, we have something that looks like this:

$$A = PDQ$$

$$\begin{bmatrix} . & . \\ . & . \\ . & . \end{bmatrix} = \begin{bmatrix} . & . & . \\ . & . & . \\ . & . & . \end{bmatrix} * \begin{bmatrix} a & . \\ . & b \\ . & . \end{bmatrix} * \begin{bmatrix} . & . \\ . & . \end{bmatrix}$$

Let A be $m \times n$, P must be $m \times m$, Q must be $n \times n$. Therefore, D must be $m \times n$, and all entries below and above the diagonal of D are zero.

Definition 6.1 *Unitary Matrix*

A complex matrix U is unitary if

$$UU^H = U^H U = I$$

where U^H is the transpose for complex numbers. [2]

The columns of P are called the left singular vectors of A, and the columns of Q are called the right singular vectors of A. The left singular vectors are the normalized eigenvectors of AA^T, and the right singular vectors are the normalized eigenvectors of $A^T A$.

Together, the left and right **singular vectors** combined span all of matrix A. The left singular vectors span the column space of A, and the right singular vectors spans the row space of A.

The **singular values** are the square roots of the eigenvalues of $AA^T$and $A^T A$.

Overall, the largest singular vectors explain the most variance of matrix A, and the singular values associated with each singular vector tell us how much variance is explained.

6.1 Aside: Use of AA^T and AA^T in SVD

Why are we referring to AA^T and $A^T A$ instead of just matrix A? The original matrix A is not necessarily square. The dataset in examination could be any size mxn. Also, A may not have positive eigenvalues that exist.

However, the matrices AA^T and $A^T A$ are positive semidefinite. This means they are symmetric, have positive eigenvalues, and eigenvectors that are pairwise orthogonal. Even though the singular values and singular vectors of the SVD of A provide information about the eigenvectors and eigenvalues of AA^T and $A^T A$, this is still useful information.

Definition 6.2 (Positive Semi-Definite Matrices) *A* ***positive semi-definite matrix*** *C can be obtained as the product of a matrix X and its transpose X^T.*

$$C = XX^T \tag{3}$$

[2]

The properties of a positive semi-definite matrix are:

- Symmetric

- Positive eigenvalues

- Pairwise orthogonal eigenvectors

The eigenvectors for a given matrix are not always necessarily orthogonal. However, for PCA, a certain type of matrix is examined which ensures orthogonal eigenvectors, and ensures that an eigen- decomposition exists. When using PCA, and specifically when I perform PCA using R in this project, analysis is performed on a square correlation matrix. This matrix is a positive semi-definite matrix.

7 Principal Componenets Analysis Overview

PCA is built on the singular value decomposition. The matrix of interest, A, can be factored using the SVD to easily show the eigenvectors and eigenvalues of the normalized matrices AA^T and A^TA. Eigenvectors, known as principal components, are retained based on the amount of variance that they explain, which is quantified by eigenvalues. Therefore, the original data set can be represented by these new, orthogonal, principal components.

The four goals of PCA are to (1) extract only the most important information from the data set, (2) compress and reduce the size of the data set via keeping only the important information, (3) simplify the description of the data set, (4) analyze the structure of (and relationships between) the variables. [1]

7.1 Change of Basis and Dimension Reduction

Data sets with large numbers of variables can be really difficult to visualize. Using PCA, the data is compressed to use a basis which is usually the first principal components. Therefore, we can create plots such as biplots which simply have a principal component on the x and y axis. This is much easier to visualize.

8 Data Cleaning

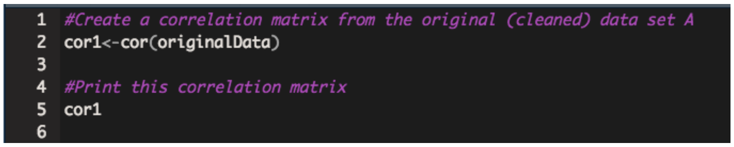
The process of correcting the original data set, known as data cleaning, took longer than expected. Quantitative variables were identified and renamed. PCA works best on quantitative variables. Data cleaning also included removing commas from values with more than three digits, because otherwise these values were separated into different numbers based on the comma location. Then, smaller data frames with select, easy to work with variables were created so that I could run the analysis and learn how to interpret the results before incorporating more variables. Many of the original, uncleaned variables were missing data points. The missing data was replaced with 0 for the question asking about the condition of the residences. This is addressed more thoroughly in Section 11, the Limitations section.

9 Principal Components Analysis Steps

1. **Calculate the Correlation Matrix**

   In order to standardize the data, the original matrix, A, is centered so that the mean of each column equals 0, and A is normalized by dividing each variable by the length of the vector. Therefore, values will range from 0 to $+/-1$. The diagonals of the correlation matrix will be 1 and this is a symmetric matrix. [1]

   The code used in R to do this is shown below:

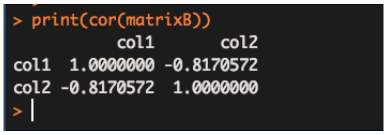
   

   ```
   #Create a correlation matrix from the original (cleaned) data set A
   cor1<-cor(originalData)

   #Print this correlation matrix
   cor1

   ```

   An example of this correlation matrix calculation is shown below. Given some matrix B,

   $$B = \begin{bmatrix} 3 & 4 \\ 1 & 2 \\ -4 & 8 \end{bmatrix}$$

   we can find the correlation matrix in R using the above code. The result is:

   ```
   > print(cor(matrixB))
              col1       col2
   col1  1.0000000 -0.8170572
   col2 -0.8170572  1.0000000
   > |
   ```

   We see that the correlation matrix is a symmetric matrix with 1's along the diagonal. Column 1 clearly has a linear relationship with itself, column 1, so the value for this correlation coefficient is 1. This is the same situation for column 2 and column 2. The correlation coefficient between column 1 and column 2 is -0.817. This suggests that as column 1 increases, column 2 decreases. This is a negative relationship. The relationship holds in the opposite direction as well. All values are between 0 and 1.

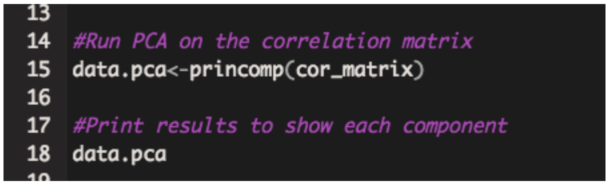
2. **Use SVD to find Eigenvectors and Eigenvalues of Correlation Matrix**

   SVD is used to factor the correlation matrix into a form that clearly shows the eigenvectors and eigenvalues of the correlation matrix. The linear algebra behind this is discussed in Section 6.

3. **Obtain First Principal Component from Largest Eigenvalue**

   The largest eigenvalue tells us the eigenvector associated with capturing the most variance. This eigenvector becomes the first Principal Component.

   Running PCA in R on the correlation matrix is shown below.



```
#Run PCA on the correlation matrix
data.pca<-princomp(cor_matrix)

#Print results to show each component
data.pca
```

4. **Decide How Many Components to Keep**

   Ideally, this is done by looking for natural points where the variance explained drops disproportionately. The next section will discuss this in more depth and show a plot that is useful for this decision.

10 Results

10.1 Scree Plot

A scree plot graphs the explained variance versus the principal component number. It is a tool to visualize how much variance each principal component explains. Therefore, the first principal component should explain the largest amount of variance, the second slightly less, and so on. For example, in Figure 1, principal component 1 is shown by the largest bar on the bar graph, and explains 69% of the variance, while principal component 2 explains 21 % of the variance. A scree plot shows the eigenvalues plotted from left to right. Generally, we will retain the components that are above the scree, or the parts before the slope of the graph levels out.

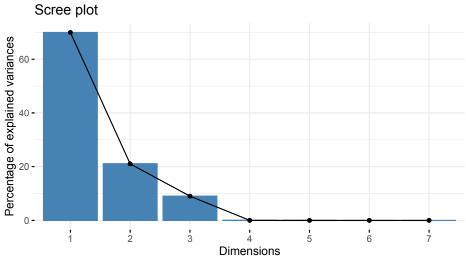


Figure 1. A scree plot that graphs the explained variance on the y-axis versus the principal component number on the x-axis.

10.2 Bi-plot

A bi-plot graphs the loadings on a graph to help visualize relationships between variables and interpret the principal components. The graph has two axis- one for each of the first two principal components. The direction of the vectors provides information on which principle component that variable contributes more to. The length of the vector loadings can tell us how much each variable contributes to a certain principal component.

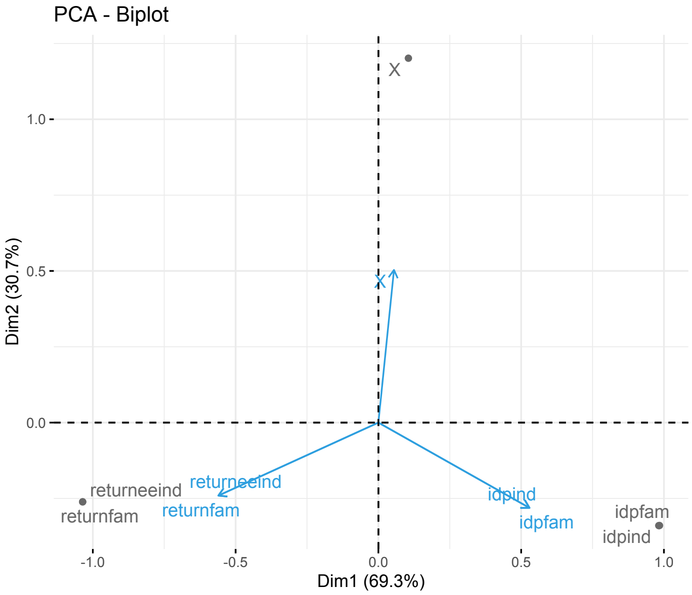


Figure 2. This biplot graphs the loadings onto the two axis, with the x-axis showing the first principal component and the second principal component on the y-axis.

Figure 2 shows the results of a preliminary test run of PCA on a small data frame, to become familiar with how PCA works in R and to learn how to interpret the results. The first principal component explains 69% of the variance, and the second 21%, as is confirmed by the scree plot in Figure 1.

The interesting takeaway from this is the presence of variable X, which at first was a mystery. However, it was determined that X is actually the index of the data frame, labeling each observation in order from 1 to 3,000+. Since the loading for variable X is almost directly on the y-axis, it is clear that variable X contributes almost entirely to principle component 2, while the other variables, `returnfam`, `returnind`, `idpfam`, and `idpind` contribute strongly to principal component 1, as their loadings are more extreme along the x-axis. Additionally, the number of returnees and IDPs are slightly negatively correlated to principal component 2. This suggests that as the index of the data frame increases, the number of individuals and families slightly decreases. This is intriguing and could be due to the prioritization of interviewing locations with larger amounts of people.

Also, from Figure 2, it is clear that the variables for number of individuals and families have the same loading. The variables correspond with vectors of the same length and direction, so show up on top of each other in the bi-plots. From this information, I checked the data points, and sure enough the variables for the number of individuals and number of families are multiples of each other. The number of families times 6 results in the number of individuals. Therefore, these variables are not as unique as was originally thought.

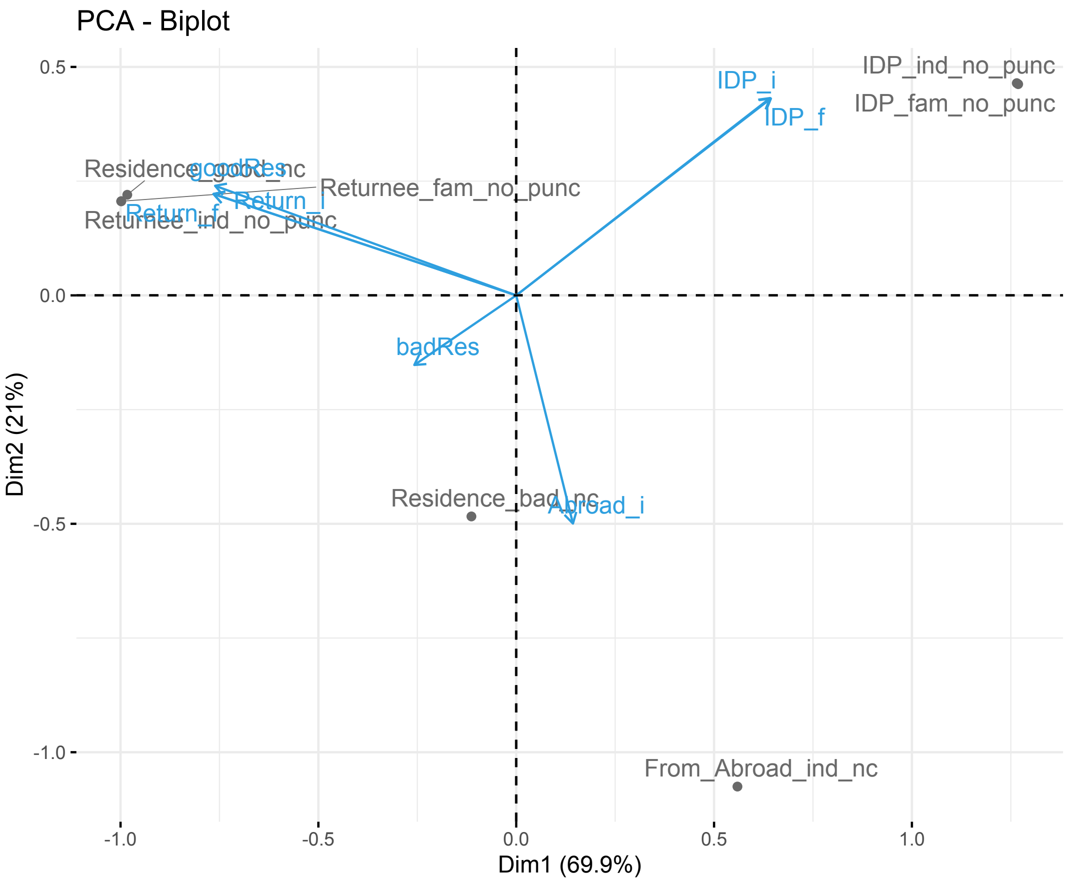


Figure 3. This biplot shows the loadings of more variables onto the two principal components.

More variables than in Figure 2 are included in Figure 3. The index variable, X, was removed for this data frame. The variables for residence quality, labeled `Residence_good` and `Residence_bad` load on vectors that point in opposite directions. Therefore, these two variables contribute to the second principal component in inverse ways. As principal component 2 decreases, the effect on Residence_good versus Residence_bad is opposite. This makes sense, as we would expect these variables to be negatively associated.

11 Limitations

One decision that was made was to replace the missing values for the Residence_good and Residence_bad variables with 0. Each observation of a variable is a value that counts how many people at the collection location have a residence that is in good condition, for the Residence_good variable, and how many people have a residence that is in poor condition, for the Residence_bad variable. Replacing the missing values in these categories with 0 shifts the averages of the values more towards 0, and indicating that the people at this location did not have houses in these conditions. This was done with the goal of providing a neutral solution to the missing values. However, the survey reporters could have left these responses blank for numerous unknown reasons, besides for the respondents not having houses in either good or bad condition. Perhaps the people at these locations did have houses in either good or bad condition, but the survey respondents did not know how to categorize the houses into these two responses. Therefore, inputting 0 for no response may actually be misinterpreting the lack of response. This is a limitation of this analysis, and is shows how decisions about how to deal with missing data are important and can influence the interpretation of data.

12 Conclusion

Principal components analysis uses linear algebra to help us understand a dataset and make it less complicated. Specifically, a normalized and centered data set can be factored into its singular value decomposition to clearly reveal the eigenvectors and eigenvalues. The statistical software R is a useful tool for performing PCA. In this paper, we discussed a real application of PCA, specifically on the results of a survey about access to certain necessities for internally displaced persons in Iraq. One takeaway from running PCA is that it illuminated how the variables for number of individuals and families are multiples of each other. Additionally, the biplots showed that the variables for good condition residences and poor condition residences load on opposite direction vectors. Hopefully some of this information might be useful for the organization, and I plan to continue looking at the data. Further research could involve applying PCA to categorical variables.

Just as importantly, I also learned a lot. I learned that data cleaning is not a passive activity, and requires decisions that can affect your outcomes. Also, I reviewed many core linear algebra concepts while also diving deeper into topics such as specific matrix factorizations like the singular value decomposition. Also, I learned how to create and intepret graphs such as Scree plots and biplots, which

I had never seen before this capstone. I was definitely humbled about my ability to do data analysis, but I also feel empowered and proud of what I was able to get done.

13 Acknowledgements

- International Organization for Migration and H. Tran: collecting and sharing their data with me
- Dr. Justice for patience, support, and tea
- PLU Math Department

References

[1] Herve Abdi and Lynne J. Williams. "Principal Component Analysis". In: *WIREs Comp Stat* 2.4 (2010), pp. 433–459. DOI: 10.1002/wics.101.

[2] Ward Cheney and David Kincaid. *Linear Algebra Theory and Applications*. Jones and Bartlett Publishers, 2009. ISBN: 9780763750206.