A Practical Guide to Visualizing PCA Results with Biplots in R

Authored by

Mohammed looti

October 30, 2025

Principal Component Analysis (PCA) stands as a cornerstone technique in unsupervised machine learning, primarily utilized for effective dimensionality reduction. The fundamental objective of PCA is to transform a complex dataset composed of many correlated variables into a smaller, more manageable set of uncorrelated variables. These new variables, termed principal components, are constructed specifically to maximize the retention of the original data's variance. By achieving this transformation, PCA significantly simplifies intricate datasets, making them far easier to visualize, interpret, and analyze without sacrificing critical underlying information.

The theoretical basis of PCA revolves around identifying the directions within the data (the principal components) along which the data exhibits the greatest variability. The first principal component is always aligned to capture the largest possible variance present in the data. Subsequently, each following component is calculated to capture the maximum remaining variance, while maintaining orthogonality (being uncorrelated) to all preceding components. These components are mathematically defined as linear combinations of the original variables, and they are inherently ordered by the amount of variance they are able to explain.

Despite the powerful analytical insights provided by Principal Component Analysis, the interpretation of its multivariate output can often be challenging, particularly when dealing with datasets that contain a large number of original variables. This is precisely where the **biplot** proves indispensable. A biplot is an advanced graphical tool designed to simultaneously visualize both the observations (rows) and the variables (columns) of a dataset onto a single, two-dimensional plane. This plane is typically defined by the first two principal components. This unified visualization dramatically assists in deciphering complex relationships: it shows how observations relate to one another, how variables influence the components, and how these elements collectively contribute to the overall structure discovered in the data.

Understanding the Biplot in PCA

Serving as a comprehensive visual synopsis of a Principal Component Analysis, the biplot effectively projects high-dimensional data onto a reduced-dimensional space. This space is chosen to be the dimension that captures the most significant variance--usually the plane defined by the first two principal components. Within this plot, every individual data point, representing an observation, is marked by a symbol, while each original variable is represented as a vector, or arrow, originating from the plot's central origin.

The spatial arrangement of the observation points within the biplot immediately reveals crucial information about the similarities and dissimilarities between them. Observations that are clustered closely together are highly similar across their original variable values. Conversely, observations that are located far apart suggest a high degree of dissimilarity. This powerful visual clustering capability allows analysts to quickly identify inherent patterns, distinct groups, or potential outliers

within the dataset, providing immediate structural insights.

Furthermore, the variable arrows offer deep insights into two key aspects: their contribution to the principal components and their interrelationships. The length of an arrow is directly proportional to how much that variable contributes to the variation explained by the displayed principal components. Longer arrows denote variables that exert a stronger influence on the data structure captured. The angle between any two arrows indicates the correlation between the corresponding variables: a very small acute angle suggests a strong positive correlation, an angle near 90 degrees implies negligible or zero correlation, and an angle approaching 180 degrees signals a strong negative correlation.

Beyond these relationship indicators, the biplot allows for an approximate assessment of an observation's value for a specific variable. By projecting an observation point orthogonally onto a variable's arrow (or its extension), one can gauge whether that observation has a high or low value for that variable. This powerful dual representation is what makes the biplot an exceptionally versatile tool for simultaneously exploring relationships among observations, relationships among variables, and how specific observations are characterized by those variables.

Implementing PCA and Biplots in R

The R programming language provides highly efficient and robust tools for executing PCA and visualizing the resulting output. The standard function for performing PCA on a numeric data matrix or data frame in R's base `stats` package is `princomp()`. This function is designed to compute the principal components, typically using the spectral decomposition of the covariance matrix, providing a mathematically sound basis for the analysis.

Once the principal components have been successfully computed, the `biplot()` function is immediately employed to generate the critical visual representation. This function accepts the results object produced by `princomp()` as its primary argument and automatically constructs the biplot. A key feature of `biplot()` is its intelligent scaling of both the variable vectors and the observation points, ensuring that all elements fit cohesively within the same plot space, which greatly facilitates immediate and accurate interpretation.

The following fundamental syntax demonstrates the straightforward steps required to first execute PCA and then generate the corresponding biplot visualization within R:

```
# Step 1: Perform Principal Component Analysis on the dataset
results <- princomp(df)

# Step 2: Create a biplot to visualize the PCA results
biplot(results)
```

In this example, `df` must be a strictly numeric matrix or data frame containing the variables for analysis. The `princomp()` function returns a comprehensive object that encapsulates all components of the PCA, including the standard deviations of the components, the loadings (which are the eigenvectors or variable coefficients), and the scores (the projected data points). The subsequent `biplot()` command then utilizes these calculated results to construct the final visual plot.

Practical Example: Analyzing the USArrests Dataset

To demonstrate the practical application of creating and interpreting a biplot in R, we will utilize the classic, built-in R dataset known as `USArrests`. This dataset compiles statistics on arrests per 100,000 residents for three distinct violent crimes--assault, murder, and rape--across all 50 US states in 1973, along with the percentage of the urban population in each state. Given the presence of multiple related variables measured across geographic units, this dataset is an ideal candidate for demonstrating PCA and subsequent dimensionality reduction.

Before initiating the PCA, it is prudent to first inspect the structure of the `USArrests` dataset by examining its initial rows. This preliminary step allows us to confirm the variables involved and understand the inherent nature and scale of the data before we subject it to statistical transformation.

 # Command to view the first six rows of the USArrests dataset
head(USArrests)

Murder Assault UrbanPop Rape
Alabama 13.2 236 58 21.2
Alaska 10.0 263 48 44.5
Arizona 8.1 294 80 31.0
Arkansas 8.8 190 50 19.5
California 9.0 276 91 40.6
Colorado 7.9 204 78 38.7

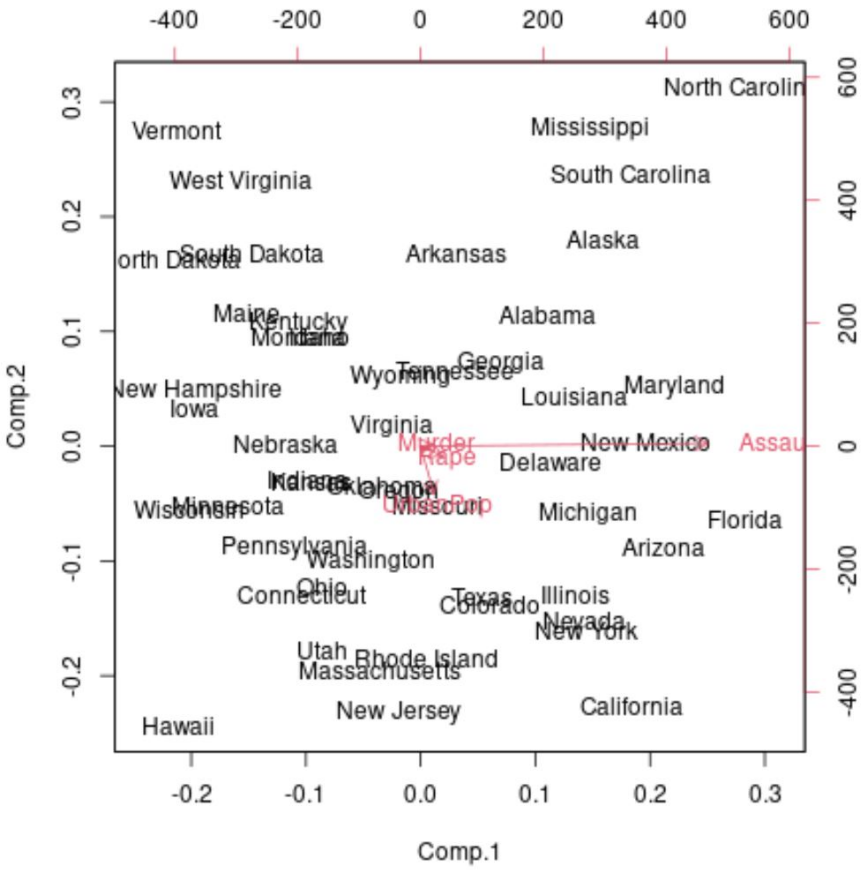
As the output confirms, the dataset consists of four quantitative variables: `Murder`, `Assault`, `UrbanPop` (percentage of urban residents), and `Rape`. Our primary objective is now to apply PCA to these variables to uncover underlying factors or patterns that drive crime statistics across US states, and then to visually articulate these findings using a biplot.

We proceed by executing the PCA using `princomp()` on the `USArrests` data, immediately followed by generating the default biplot. This sequence provides the essential first visual interpretation, allowing us to immediately assess the structure of the principal components and the

complex relationships connecting the states and the crime statistics.

 # Perform PCA on the USArrests dataset
results <- princomp(USArrests)

# Visualize the results of PCA in a biplot
biplot(results)



Interpreting the Default Biplot

The resulting biplot is a highly informative visualization of the underlying structure within the `USArrests` dataset. The horizontal axis represents the **First Principal Component**, which, by definition, captures the maximum amount of variance in the data. The vertical axis represents the **Second Principal Component**, which explains the second largest portion of variance, orthogonal to the first.

Each labeled point on the visualization corresponds to a US state (observation). States positioned in close proximity on the biplot share similar profiles in terms of their crime rates and urban

population percentages. For example, states grouped toward the far left of the plot generally indicate lower overall crime rates across the measured variables, while those clustered on the right likely exhibit significantly higher rates. This spatial separation immediately suggests underlying demographic or socioeconomic factors driving the differences.

The red arrows symbolize the original variables: `Murder`, `Assault`, `UrbanPop`, and `Rape`. Their orientation and magnitude are critical for interpretation:

Direction and Correlation: Arrows pointing in nearly the same direction indicate variables that are highly positively correlated; they tend to increase or decrease together across states. Conversely, arrows pointing in completely opposite directions suggest a strong negative correlation.
Length and Influence: Longer arrows signify variables that contribute more substantially to the structure captured by the displayed principal components. This indicates that these variables possess a greater influence on the overall pattern of the data being visualized.
Relationship to Observations: To interpret a specific state's relationship to a variable, one mentally projects the state's point onto the variable's arrow. States projecting further along the arrow's positive direction tend to have markedly higher values for that particular variable, whereas states projecting in the opposite direction have lower values.

Based on this initial biplot, we can readily identify major trends. For instance, states characterized by high `Assault` and `Rape` rates often cluster together, suggesting a common latent factor underpinning these crimes. However, a limitation of the default settings is often visible here: the labels for states and variables can overlap significantly, which impedes precise interpretation and necessitates customization.

Customizing Biplot Appearance for Clarity

Although the default biplot provides a rapid initial sketch of the data structure, its overall readability can almost always be enhanced through strategic customization. The `biplot()` function in R is equipped with a rich set of arguments that allow fine-tuning of the plot's aesthetics, thereby maximizing clarity and ensuring that specific analytical insights are immediately apparent to the viewer.

These customization arguments grant precise control over fundamental visual elements, including color schemes, label sizes, axis limits, and descriptive titles. Careful and thoughtful application of these controls ensures that the biplot serves as an effective communication tool, successfully conveying complex relationships within the data without unnecessary visual clutter. Below is a summary of the most useful arguments for aesthetic improvement:

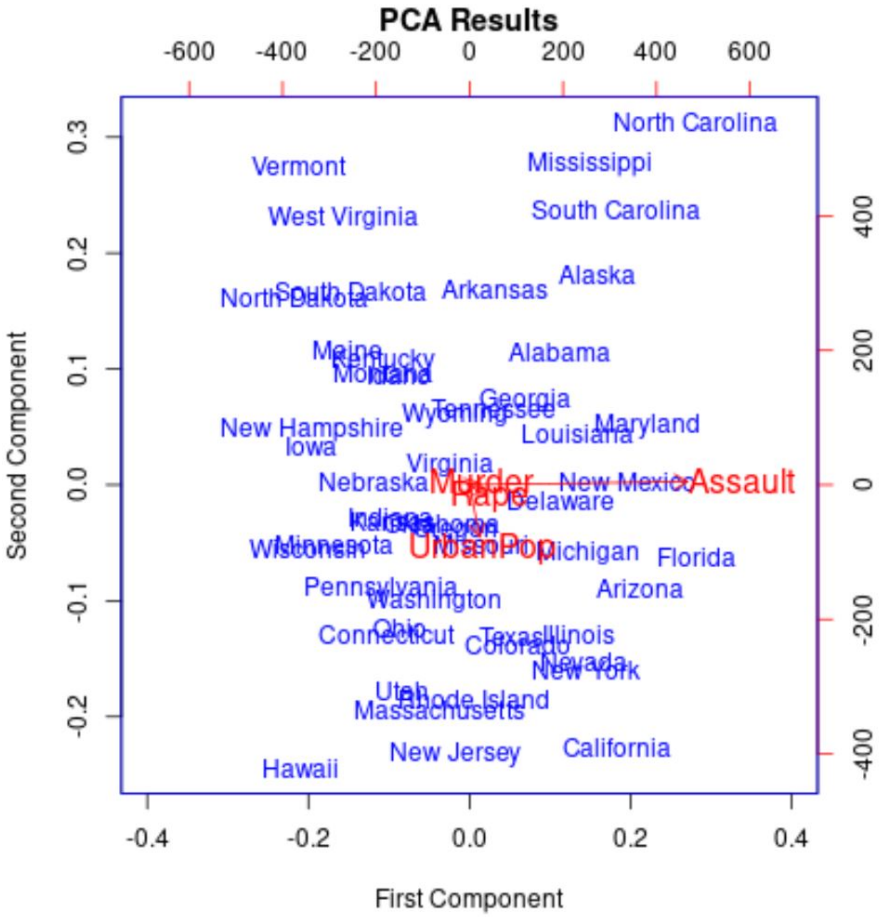
`col`: Accepts a vector of two colors. The first color is applied exclusively to the observation labels (states), and the second color is used for the variable labels and their corresponding arrows.

`cex`: Stands for 'character expansion' and governs the size of text labels. It also requires a vector of two values: the first scales the observation labels, and the second scales the variable labels. Values greater than 1 increase size; values less than 1 decrease it.
`xlim` and **`ylim`**: These define the manual limits for the x-axis and y-axis, taking a vector of two values (e.g., `c(min, max)`). Setting these manually is often essential to focus on specific data clusters or to prevent crucial labels from being truncated at the plot edges.
`main`: Used to assign a descriptive main title to the plot, clearly stating what the biplot represents.
`expand`: This argument serves as a crucial scaling factor for the variable arrows. Increasing this value lengthens the arrows relative to the observation points, which is particularly helpful if the variable vectors are initially very short and difficult to visually resolve.

By implementing these customizations, we can dramatically improve the interpretability of our biplot. For instance, color differentiation makes it easier to distinguish between states and variables, while appropriate scaling of text and arrows ensures that all elements are legible and contribute positively to the visualization.

Let's apply several of these arguments to the previous biplot generated from the `USArrests` data to enhance its readability and visual impact:

```
# Create biplot with custom appearance for USArrests data
biplot(results,
col=c('blue', 'red'), # Blue for state points, Red for variable vectors
cex=c(1, 1.3), # State labels normal size, variable labels 30% larger
xlim=c(-.4, .4), # Custom x-axis limits for better centering
main='PCA Results: USArrests Data', # Custom plot title
xlab='First Principal Component', # Descriptive x-axis label
ylab='Second Principal Component', # Descriptive y-axis label
expand=1.2) # Expand variable arrows by 20% for better visibility
```



As clearly demonstrated by the customized biplot, these adjustments substantially improve readability and analytical precision. The use of distinct colors for observations and variables, combined with appropriately scaled labels and informative axis titles, makes it significantly easier to identify patterns and comprehend the relationships within the `USArrests` data. For example, we can now more readily discern which states share similar crime profiles and understand the specific variables that contribute most strongly to those observed similarities or differences, a crucial step in exploratory data analysis.

Conclusion and Further Exploration

The biplot is unequivocally an essential visualization technique for the rigorous interpretation of results derived from Principal Component Analysis. It offers the unique capability to concurrently visualize both observations and variables within a single reduced-dimensional space, thereby significantly facilitating a deeper, more intuitive understanding of complex, multivariate datasets. By skillfully leveraging R's built-in statistical functions--specifically `princomp()` and `biplot()`--data analysts can efficiently uncover hidden structures, identify natural clusters, detect influential outliers, and clearly understand the degree of influence exerted by different variables on the overall

data patterns.

Furthermore, the extensive customization options inherent in the `biplot()` function empower users to meticulously tailor their visualizations to meet specific analytical requirements, maximizing both clarity and communicative impact. Mastery of creating and accurately interpreting biplots is therefore a highly valuable and necessary skill for anyone routinely engaged in exploratory data analysis and dimensionality reduction. We strongly encourage practitioners to experiment thoroughly with their own datasets and explore the full spectrum of available customization arguments to unlock the deepest possible insights from their PCA results.

For those seeking to delve further into the theoretical and practical aspects of Principal Component Analysis and related multivariate techniques, the following resources are highly recommended for supplementary information and practical examples: