**Simplifying Sensory Data: A Practical Guide to QDA Using an R Shiny app**

\*Alexis Zhaid Carrillo-García1

\*1Ingeniero agroindustrial. Departamento de Ingeniería Agroindustrial. Universidad Autónoma Chapingo. Km. 38.5 Carretera México-Texcoco. Chapingo, Texcoco, Estado de México CP 56230. Correo electrónico: [alexiszcg@gmail.com](alexiszcg@gmail.com%20) ; Orcid ID: <https://orcid.org/0000-0003-0593-4657>

**Abstract**

This article introduces a sensory evaluation application developed to analyze Quantitative Descriptive Analysis (QDA) data efficiently. Designed with R Shiny, the application combines advanced statistical analyses with intuitive visualizations, addressing the needs of sensory scientists and product developers. It facilitates tasks such as exploring correlation matrices, conducting Principal Component Analysis (PCA), performing Analysis of Variance (ANOVA) with post hoc tests, and generating reproducibility graphs and external preference maps. The application supports researchers in identifying significant sensory attributes, evaluating treatment differences, and assessing panelist performance. Its user-friendly design incorporates interactive elements like pickerInputs and selectInputs, enabling customized analyses based on specific attributes or treatments. By reducing the complexity of sensory data interpretation, it empowers users to make informed decisions about product development and optimization. The visual outputs, such as radial plots and biplots, offer an accessible way to communicate findings and uncover patterns in sensory profiles. With its comprehensive set of tools and streamlined workflows, this application addresses the growing demand for efficient, reliable, and customizable solutions in sensory analysis, bridging the gap between traditional QDA methodologies and modern data science.

**Keywords:**  Quantitative Descriptive Analysis (QDA), Sensory Evaluation, R Shiny

**Introduction**

**Quantitative Descriptive Analysis (QDA®)** is a **sensory evaluation method** developed in the early **1970s** by **Tragon Corporation** in collaboration with the **Department of Food Science at the University of California, Davis** (Stone et al., 1974). It emerged as a response to the limitations of earlier descriptive methods, such as the **Flavor Profile Method**, which lacked robust **statistical treatment of sensory data**.

Unlike traditional methods, **QDA® emphasizes panelist independence** in evaluations, aiming to **reduce bias** and **ensure objectivity**. At its core, QDA® is characterized by a **statistical and systematic approach to sensory evaluation**, where trained panelists evaluate products based on **predefined sensory attributes**.

Candidates are carefully **screened based on their ability to discriminate sensory differences**. There are various **techniques available for candidate selection**—although these methods will not be covered in detail in this article, they are worth exploring if you are planning to conduct a **QDA® study**. Panelists undergo **training using product and ingredient references** to establish a **common sensory vocabulary** and **reduce inconsistencies in evaluations** (Meilgaard, Civille, & Carr, 2007).

Once collected, the sensory data undergo a series of statistical analyses to identify patterns, relationships, and significant differences among samples. The most common statistical tools and techniques used in QDA® include:

1. Correlation Matrix: examines the relationships between sensory attributes.

2. Principal Component Analysis (PCA): used to reduce the dimensionality of the dataset while preserving as much variability as possible. It generates biplots that visually represent relationships between sensory attributes, panelists, and samples.

3. Analysis of Variance (ANOVA): applied to identify significant differences between treatments for each sensory attribute.

4. Post hoc Tests: these tests determine which specific treatments differ from each other.

5. Interaction Plots: Interaction plots visualize how panelists and treatments interact across sensory attributes. They are essential for detecting potential inconsistencies or biases in panelist evaluations.

6. Reproducibility Analysis This analysis assesses the consistency of panelist evaluations across repetitions and treatments.

Each of these analytical techniques contributes uniquely to understanding the sensory profile of products, providing a robust foundation for data-driven decision-making in product development and quality control. In the Results section of this document, each statistical analysis will be discussed in detail, with examples and visualizations generated directly from the application.

QDA® remains a cornerstone methodology in sensory science. Its structured approach to panelist training, attribute quantification, and statistical validation has set a gold standard in the industry. However, modern sensory analysis increasingly relies on technological tools to handle large datasets, improve reproducibility, and create more interactive and meaningful visualizations. In this context, the Shiny-based sensory analysis application presented in this document serves as a bridge between traditional QDA® methodologies and modern data science techniques, offering a user-friendly interface for performing complex statistical analyses and generating intuitive visual outputs.

**Methodology**

**1. Software Architecture**

The Shiny-based sensory analysis application is built using the R Shiny framework, a powerful tool for developing interactive web applications directly from R. Its modular design is structured into two primary components: User Interface (UI) and Server. These components work together to provide a seamless and interactive user experience while performing robust statistical analyses.

**1.1 Libraries and Tools Used**

The functionality of the application relies on several **R libraries** that provide statistical analysis, data manipulation, and graphical capabilities:

* **FactoMineR**: Performs Principal Component Analysis (PCA) and multivariate statistical analyses.
* **factoextra**: Enhances PCA results with visually appealing and interpretable plots.
* **DT**: Creates interactive tables for data exploration.
* **ggplot2**: Generates custom visualizations with high-quality graphical outputs.
* **lme4** and **lmerTest**: Facilitate linear mixed-effects models for statistical testing.
* **agricolae**: Provides tools for ANOVA and post hoc tests like LSD.
* **highcharter**: Creates interactive radial plots and advanced visualizations.

**1.2 Workflow Overview**

The application follows a **reactive programming paradigm**, ensuring that changes made by the user in the **UI** are instantly reflected in the outputs displayed.

1. **Data Upload:** Users upload datasets in Excel format via the **UI**.
2. **Data Preprocessing:** The **Server** validates and processes the uploaded data.
3. **Statistical Analysis:** Calculations for PCA, ANOVA, and other tests are executed on the **Server**.
4. **Visualization and Interaction:** Results are rendered as tables, graphs, and interactive visualizations in the **UI**.
5. **Customization and Exportation:** Users can filter data dynamically and export results for reporting purposes.

This modular design allows the application to remain **scalable**, **flexible**, and **user-friendly**, accommodating both novice users and experienced statisticians.

### ****2 Data Input and Pre-processing****

The **Shiny-based sensory analysis application** begins its workflow with a **data upload stage**, where users can input sensory datasets for analysis. Proper data structuring and pre-processing are critical to ensure accurate statistical results and meaningful visualizations.

### ****2.1 Data Format and Structure****

The application accepts datasets in **Excel format (.xlsx)**. Each dataset must adhere to a predefined structure to ensure compatibility with the statistical functions integrated into the application.

* **Rows:** Represent individual **observations** or evaluations.
* **Columns:** Include the following key variables:
  + **Panelist:** Identifies the individual evaluating the sample.
  + **Treatment:** Represents the different experimental conditions or product variations.
  + **Repetition:** Tracks repeated evaluations for reproducibility analysis.
  + **Attributes:** Numerical scores evaluating sensory attributes such as **taste**, **aroma**, and **texture**. Any column added after Repetition will be treated as an attribute no matter the name, it is important that the name assigned to the column be unique with no spaces or special charaters.

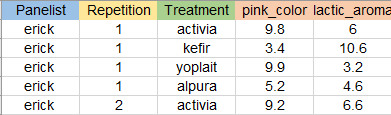


Figure 1. data structure example

### ****2.2 Data Upload****

* Users upload datasets through the **fileInput widget** on the **UI**.
* The uploaded file is read and processed using the **readxl** package.
* Panelists and treatments are converted into **factors**, ensuring compatibility with subsequent statistical models.

**Validation Checks:**

1. Verify that all **required columns** are present.
2. Ensure **no missing values** in critical columns (Panelist, Treatment, Repetition).
3. Validate **data types** (e.g., numerical attributes must contain numeric values).
4. Remove or flag inconsistent or duplicate entries.

### ****2.3 Pre-processing Steps****

Once the dataset is successfully uploaded, it undergoes a series of **pre-processing steps** to prepare it for statistical analysis:

1. **Data Cleaning:**
   * Removal of rows with **missing or invalid values**.
   * Standardization of column names to avoid inconsistencies.
2. **Factor Conversion:**
   * Columns like **Panelist** and **Treatment** are converted into **factors** to enable proper statistical grouping.
3. **Numeric Verification:**
   * Ensure that sensory attributes are stored as **numeric variables**.
4. **Reactive Updates:**
   * Any changes or filtering by the user (e.g., excluding panelists via pickerInput) are dynamically reflected across all analyses and visualizations.

### ****2.4 Data Readiness for Analysis****

After successful preprocessing, the dataset is considered **analysis-ready**. At this point:

* Data is stored in a **reactive object**, ensuring that updates (e.g., exclusion of specific panelists) propagate across all statistical analyses and visualizations.
* The dataset is then passed to subsequent modules for **PCA, ANOVA, Interaction Plots**, and other statistical analyses.

### ****3. Statistical Analyses Workflow****

The **statistical workflow** of the Shiny-based sensory analysis application is designed to streamline data processing, analysis, and visualization through an intuitive step-by-step approach. The workflow integrates **exploratory data analysis (EDA), hypothesis testing**, and **visualization techniques** to facilitate the interpretation of sensory data.

The statistical analysis process follows a structured sequence of steps:

1. **Data Upload and Pre-processing:**
   * Users upload sensory datasets in Excel format.
   * Data undergo validation, cleaning, and pre-processing.
2. **Exploratory Data Analysis (EDA):**
   * Initial data exploration includes **summary statistics** and **correlation matrices**.
3. **Correlation Matrix:**
   * Examines relationships between sensory attributes.
   * Identifies positive and negative correlations that may influence product perception.
4. **Principal Component Analysis (PCA):**
   * Reduces dataset dimensionality while preserving maximum variance.
   * Generates visual representations (**eigenvalue plots**, **biplots**) to identify patterns and relationships among sensory attributes, treatments, and panelists.
5. **Analysis of Variance (ANOVA):**
   * Identifies significant differences between treatments for each sensory attribute.
   * P-values are used to determine statistical significance.
6. **Post hoc Tests (LSD Test):**
   * Conducted when ANOVA detects significant differences.
   * Compares treatment means to identify specific pairs with significant differences.
7. **Interaction Plots:**
   * Visualize interactions between **panelists** and **treatments** across sensory attributes.
   * Help detect inconsistencies in panelist evaluations.
8. **Reproducibility Analysis:**
   * Evaluates the consistency of panelist across multiple repetitions.
   * Ensures reliability of panelist performance.
9. **Preference Mapping:**
   * Integrates consumer preference data with sensory profiles.
   * Highlights key sensory drivers influencing sample preference.
10. **Radial Plot:**
    * **Displays sensory attributes as spokes radiating from a central point.**
    * **Each treatment is represented by a distinct polygon, with the vertices corresponding to attribute intensity scores.**

### ****4. User Interaction and Customization****

One of the strengths of this **Shiny-based sensory analysis application** lies in its **high degree of interactivity and customization**. Users can dynamically filter, select, and adjust parameters to tailor the analysis and visualizations to their specific needs. This flexibility enhances both the **exploration of data** and the **interpretation of results**.

### ****4.1 Dynamic Filters and Controls****

The application incorporates multiple **interactive widgets** that allow users to adjust the parameters of their analysis in real time.

* **SelectInput:** Allows users to **choose specific sensory attributes** for focused analysis (e.g., in ANOVA or interaction plots).
* **PickerInput:** Enables **multi-selection of panelists or treatments** for inclusion or exclusion in the analysis.
* **CheckboxGroupInput:** Provides options to toggle the inclusion of specific data subsets.
* **FileInput:** Facilitates **dataset uploads** in an intuitive way.

### ****4.2 Excluding Panelists or Treatments****

* Users can **dynamically exclude specific panelists or treatments** from the analysis using the pickerInput widget.
* Exclusions are immediately propagated across all analysis modules (PCA, ANOVA, reproducibility, etc.).
* This feature is especially useful for identifying and mitigating the influence of **outliers or inconsistent panelists**.

### ****4.3 Seamless Integration Across Modules****

All interactive controls and user selections are **synchronized across the entire application**:

* Excluding a panelist in one module affects all subsequent analyses.
* Filtering an attribute in the ANOVA module automatically updates **interaction plots** and **reproducibility graphs**.
* Changes propagate to both **tabular outputs** and **visualizations**.

**Results**

The dataset comprises sensory evaluations of yogurt samples conducted by a trained panel. Each evaluation is characterized by the following key variables:

* **Panelist**
* **Repetition**
* **Treatment**
* **Sensory Attributes,** such as:
  + **Pink Color**
  + **Lactic Aroma**
  + **Sweet Aroma**
  + **Strawberry Flavor**
  + **Lactic Flavor**
  + **Sour Flavor**
  + **Sweet Flavor**
  + **Viscosity**

The purpose of this analysis is to identify patterns and differences among treatments based on their sensory profiles, evaluate panelist consistency and reproducibility and to provide actionable insights into key sensory drivers influencing yogurt differentiation and consumer preferences. The following subsections will delve into each statistical analysis applied to the dataset, supported by interactive visualizations and interpretations.

**Table**

The **Table** tab provides an interactive view of the dataset (Figure 1), allowing users to explore the sensory evaluations in detail. It includes key variables such as panelists, repetitions, treatments (e.g., Activia, Kefir, Yoplait, Alpura), and sensory attributes like pink color, lactic aroma, strawberry flavor, and viscosity. Users can filter, sort, and search within the table to quickly identify specific data points or patterns. This functionality ensures that the dataset is organized and ready for further statistical analysis, serving as a preliminary step to verify data integrity and consistency.

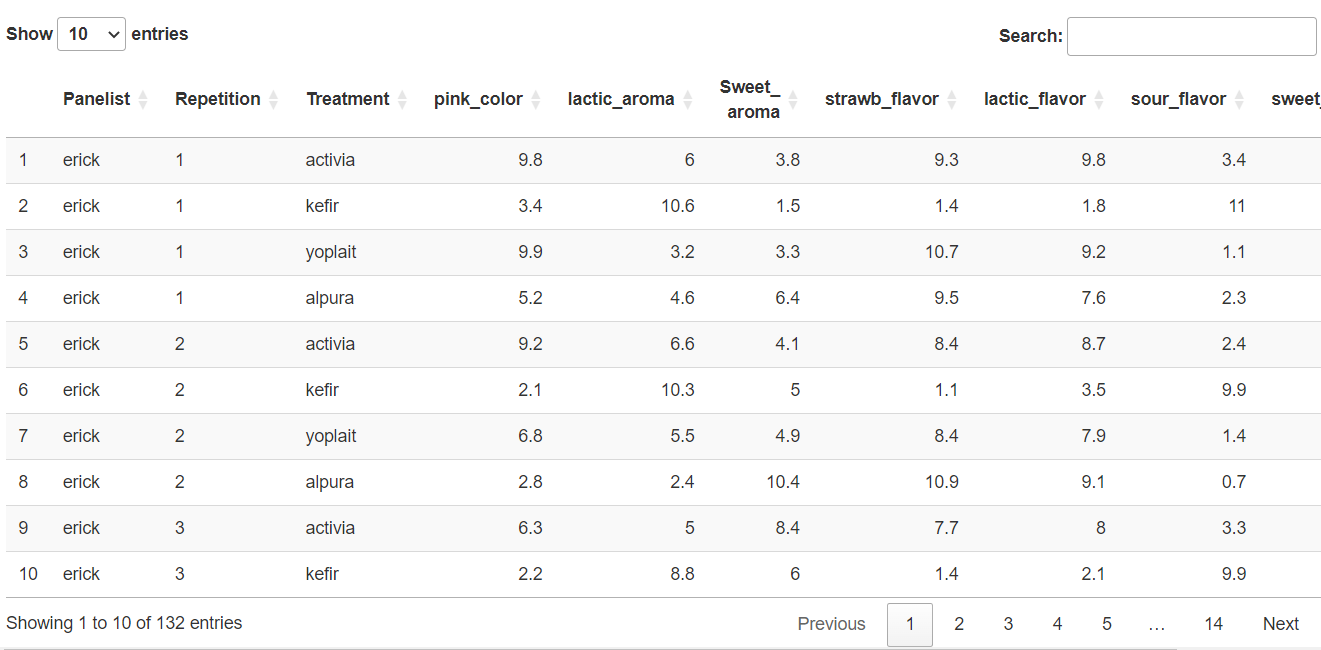


Figure 2. Table tab

**The Correlation Matrix**

The Correlation Matrix is a foundational tool in multivariate analysis that measures the strength of linear relationships between pairs of variables using correlation coefficients. These coefficients, ranging from -1 to 1, summarize the degree and direction of association between variables, with diagonal elements always being 1 since each variable is perfectly correlated with itself. By condensing the original dataset into a smaller, manageable matrix, the correlation matrix provides a preliminary step in understanding relationships among variables, often serving as the basis for more advanced analyses like PCA (Bartholomew, D. J., 2010).

The correlation matrix shown in Figure 2 reveals the relationships between sensory attributes in yogurt samples. Strong positive correlations are observed between Strawberry Flavor and Sweet Flavor, as well as between Pink Color and Strawberry Flavor, suggesting that sweeter and more visually vibrant yogurts tend to emphasize strawberry flavor. In contrast, strong negative correlations, such as between Sour Flavor and Sweet Flavor or Sour Flavor and Strawberry Flavor, indicate trade-offs between sourness and sweetness in product formulations. Attributes like Viscosity and Lactic Flavor show weak correlations with most other attributes, suggesting they play a more independent role in defining sensory profiles. These findings highlight key sensory dynamics that can guide both product optimization and consumer preference mapping.

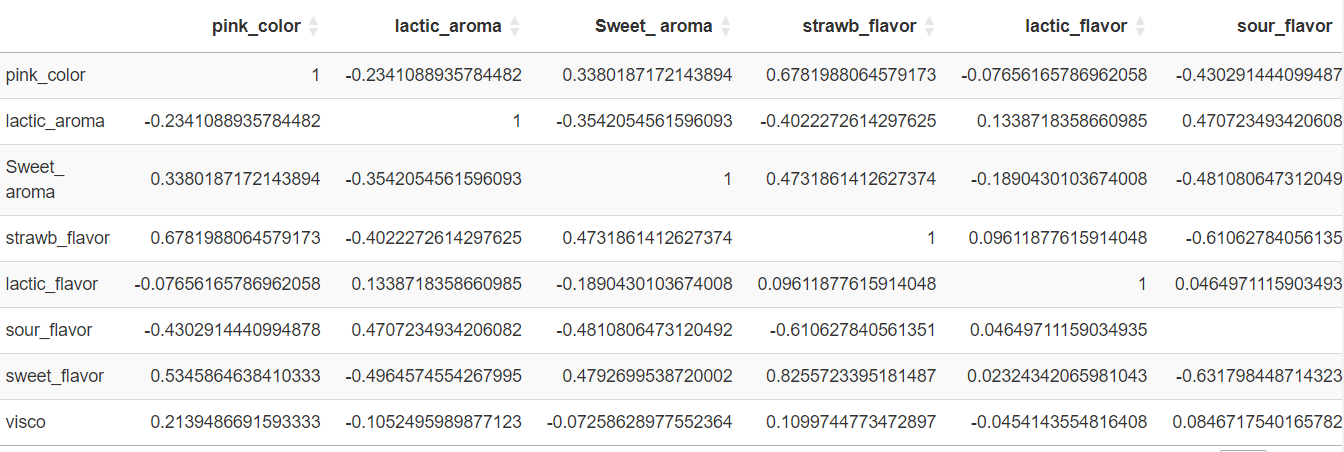


Figure 3. Correlation Table

Performing a correlation matrix as the first step in PCA is essential to identify relationships between variables, detect anomalies, and assess data suitability. It highlights redundancy, multicollinearity, or unexpected patterns, ensuring accurate interpretation. This preliminary step simplifies data structure, guiding subsequent PCA for meaningful and reliable dimensionality reduction.

**Principal Componet Analysis (PCA)**

The correlation matrix can be overwhelming when many variables are involved. PCA reduces this complexity by identifying principal components, which are linear combinations of the original variables that capture the maximum variance in the data. By focusing on these components, PCA highlights the most relevant dimensions, allowing analysts to concentrate on the aspects that matter most (Greenacre *et al.,* 2022).

*Eigenvalues*

Eigenvalues are derived from the correlation matrix or covariance matrix of the dataset. Each eigenvalue corresponds to a principal component, indicating how much of the dataset’s total variance is captured by that component.

Eigenvalues help determine the percentage of total variance accounted for by each principal component. For instance, if the first eigenvalue accounts for 60% of the variance, it means that Dim.1 retains 60% of the original dataset's information. Summing up eigenvalues progressively shows the cumulative variance explained by the selected components. A common threshold is to retain components that collectively explain 70-90% of the variance, ensuring a balance between dimensionality reduction and information retention. Components with eigenvalues less than 1 (or another set threshold) are often discarded, as they contribute less variability than a single original variable (Bartholomew, 2010).

In the particular case that we are studying (Figure 4), Dim.1 has the largest eigenvalue (3.546), explaining 44.32% of the variance. This indicates that Dim.1 captures the most important structure in the dataset. Dim.2 adds an additional 14.26%, bringing the cumulative variance explained to 58.58%. Dim.3 contributes 13.92%, raising the cumulative variance to 72.50%, which meets the commonly used threshold of retaining 70-90% of the variance.

Dimensions beyond Dim.3 have eigenvalues less than 1 and contribute progressively less variance, based on the eigenvalue rule (retain components with eigenvalues > 1), we would keep Dim.1, Dim.2, and Dim.3, which together explain 72.50% of the dataset’s variance. This ensures a significant reduction in dimensionality while preserving most of the original information. In an ideal case, the first two dimensions should be enough to explain more than 70% which will allow us to make a biplot that can help us to visualize our data, we will review this topic further in the article.

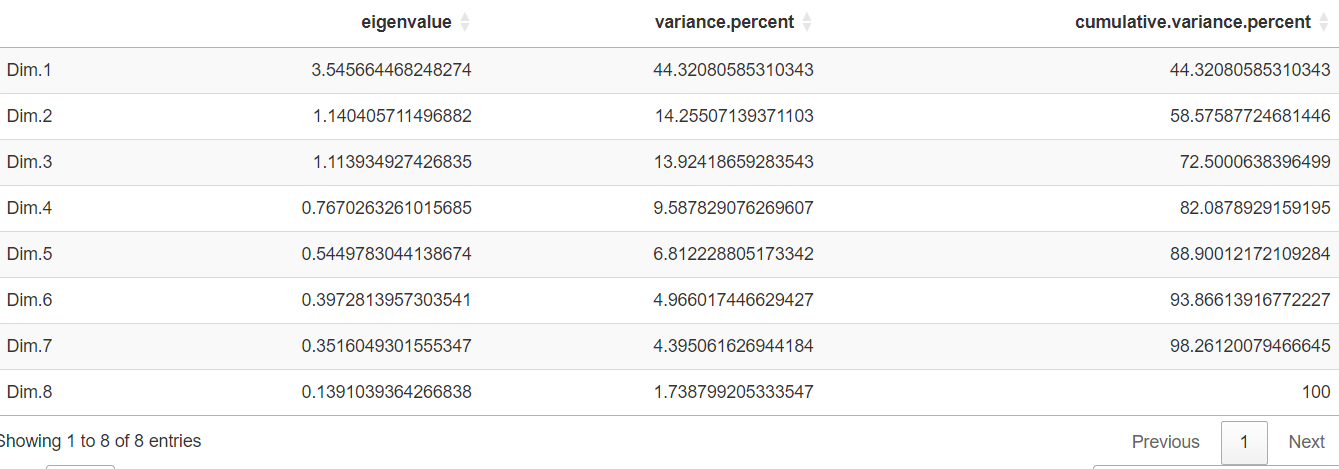


Figure . Eigenvalue table

*Eigenvectors*

Eigenvectors are the backbone of Principal Component Analysis (PCA). They define the direction of the principal components (PCs) in the multidimensional data space. Each eigenvector is a unit vector associated with an eigenvalue, and together, they determine how the original variables are combined to form each PC. Eigenvectors allow us to understand the contribution of each variable to the principal components. An eigenvector indicates the weight or importance of each variable in forming a principal component. For example, in figure 5 the eigenvector Dim.1 combines variables like Strawberry Flavor, Sweet Flavor, and Pink Color, which are positively correlated, suggesting these attributes collectively define the primary axis of differentiation in the dataset. Attributes with negative loadings, such as Lactic Aroma and Sour Flavor, counterbalance this relationship, contributing to the diversity captured by this dimension.

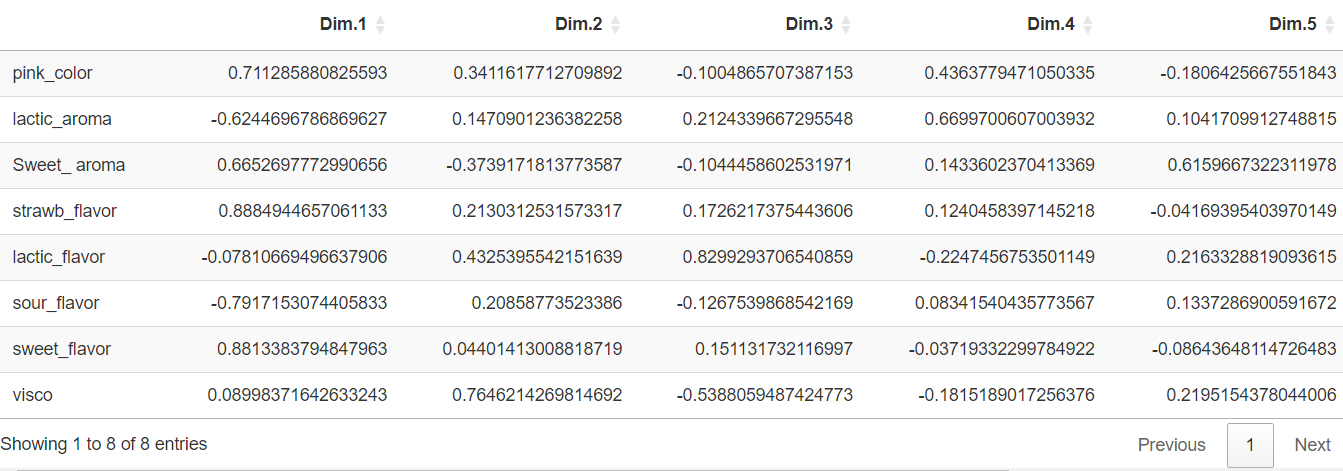


Figure 5. Eigenvectors table

**Biplots**

In the Biplot tab, users can dynamically filter the dataset using a pickerInput widget (Figure 6). This functionality allows for the selection of a specific treatment, excluding all others from the analysis. By focusing on one treatment at a time, the resulting biplot visualizes the individual performance of each panelist, providing a clearer understanding of how sensory attributes are perceived for that treatment.

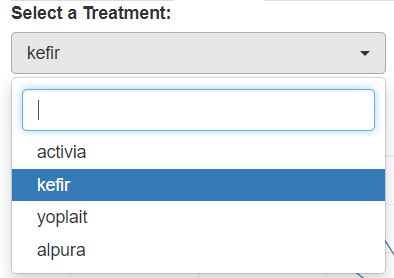
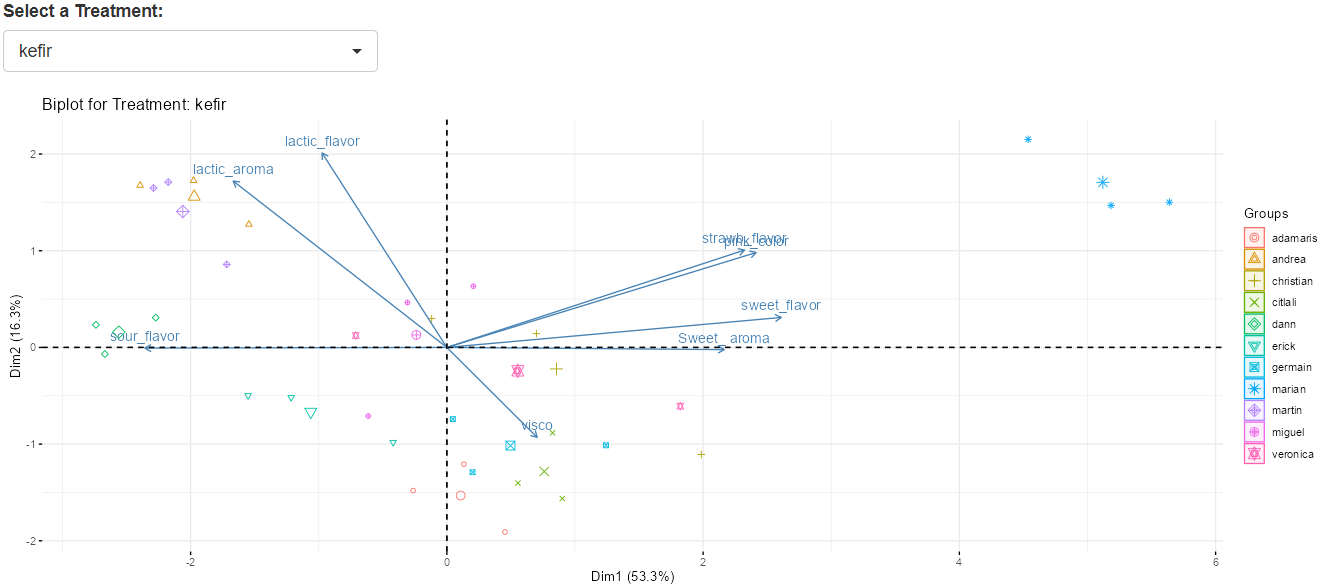


Figure 6. Biplot pickerInput

Points in the biplot are projected onto the principal components (e.g., Dim.1 and Dim.2), which summarize the majority of variance in panelists’ evaluations. Vectors represent sensory attributes (e.g., Pink Color, Sweet Aroma, etc.) and their relationship to the principal components. The length of a vector indicates the attribute’s importance in explaining variability: longer vectors represent attributes with strong influence, while shorter vectors indicate minimal contribution. The angle between vectors reflects the correlation between attributes, with acute angles indicating positive correlations and obtuse angles indicating negative correlations (Young, 2025).

The spread of points in the biplot reflects variability among panelists for the selected treatment. Clustered points indicate consistent evaluations among panelists, while scattered points suggest divergent perceptions. Each panelist is represented by a unique symbol; when symbols are closely grouped, this suggests a consensus within the panel. In contrast, isolated symbols indicate panelists whose evaluations differ significantly, highlighting the potential need for additional training.

Vectors that align closely with points provide insights into the attributes driving variability. For instance, if points are aligned with Strawberry Flavor and Sweet Flavor vectors, these attributes heavily influence panelists’ evaluations of the treatment. By repeating the analysis for different treatments using the pickerInput widget, users can identify attributes that consistently drive panelists’ perceptions or highlight treatment-specific sensory characteristics.



Scattered points

Clustered points

Isolated panelist

Figure 7. Biplot for one of the treatments

**ANOVA**

In sensory analysis, panelists are treated as blocking factors to account for variability introduced by individual differences. Products are treated as the experimental treatments being compared. ANOVA assumes no interaction between blocks (panelists) and treatments (samples). This assumption holds for highly trained and calibrated panels but can deviate in less experienced groups due to experimental error.

*Split-Plot Design in Sensory Evaluation*

In split-plot designs, repetitions are treated as blocks, and treatments (samples) are randomized within these blocks. Panelists are assigned randomly to subplots within the blocks, ensuring a balance in evaluating both treatments and panelists' performance. However, Split-plot designs are particularly useful for identifying interactions between panelists and treatments. These interactions help assess the panel’s consistency and the validity of the additive effects assumption.

When analyzing sensory data, it is important to choose a statistical approach that aligns with the study’s objectives and the experimental design. Although SensoMineR R package offers a practical and sensory-focused solution, I opted for a classical split-plot design implemented through the agricolae package for this study. The SensoMineR package simplifies the analysis by using a global Mean Square Error (MSerror) to calculate the F-statistics for treatments. While this approach is suitable for exploratory analyses, it overlooks the separation of error types (Type A and Type B) inherent to a classical split-plot design. This can lead to over- or under-estimation of treatment effects when significant interactions between panelists and treatments exist (Hernandéz-Montes, 2016).

*Comparison of Results*

Figure 8 illustrates the *p-values* obtained from SensoMineR, highlighting significant and non-significant results across sensory attributes. In contrast, the classical split-plot design uncovers nuanced differences due to its more precise handling of errors.

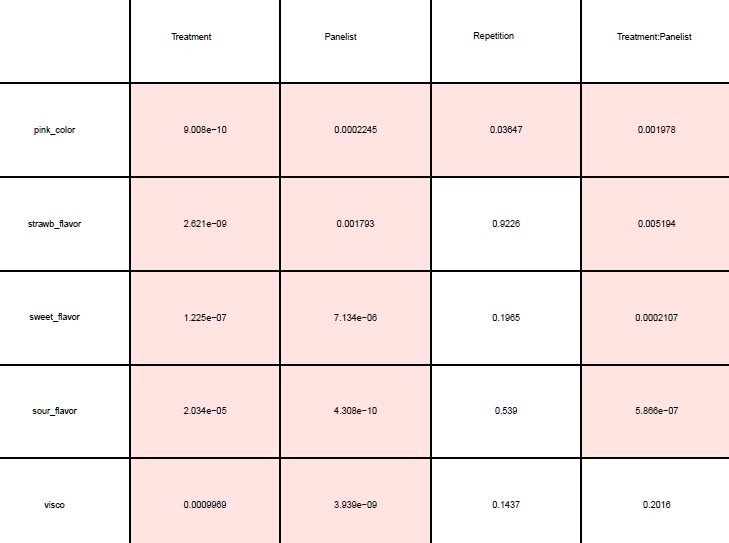


Figure 8. p-values obtained from SensoMineR package

For example, the pink\_color attribute shows a *p-value* < 0.05 for the variable repetition in the SensoMineR results. However, this value does not match the result shown in Figure 9, which was calculated using the classical split-plot ANOVA design. This discrepancy is not an isolated case; other attributes and variables exhibit similar inconsistencies.

By adopting the classical split-plot design, this analysis provides a more rigorous and accurate interpretation of the data, effectively addressing both treatment and panelist variability. While SensoMineR offers a valuable exploratory overview, the classical approach aligns better with the study's objectives, ensuring reliability and precision.

*ANOVA results and interpretation*

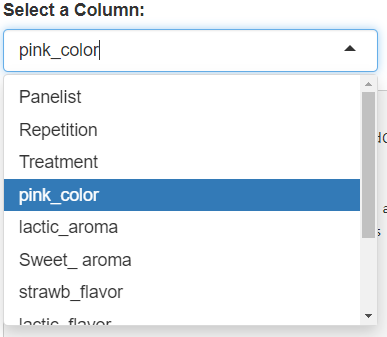


Figure 10. Picker Input

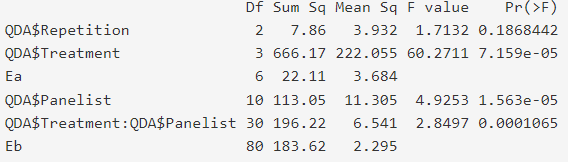


Figure 9. ANOVA split-plot table obteined with agricolae package for pink\_color attribute

In the ANOVA tab of the application, a pickerInput widget allows users to select a specific sensory attribute for analysis (Figure 10). This interactive functionality streamlines the process of focusing on one attribute at a time, ensuring a detailed and targeted evaluation.

For this example, we analyze the attribute pink\_color using the ANOVA table presented in the Figure 9:

1. **Repetition (*p = 0.1868*):**
   * Not statistically significant. This indicates that repetitions do not introduce significant variability, suggesting that the panel is consistent across sessions.
2. **Treatment (*p < 0.001*):**
   * Statistically significant. This confirms that the treatments differ significantly in their **pink\_color** attribute, which is critical for sensory evaluation.
3. **Panelist (*p < 0.001*):**
   * Statistically significant. This implies that panelists differ in their perception of **pink\_color**, which could indicate variability in panelist training or perception.
4. **Treatment × Panelist Interaction (*p < 0.001*):**
   * Statistically significant. This suggests that certain panelists perceive treatments differently, potentially indicating outliers or the need for further training.

The presence of an interaction can lead to misleading conclusions, making it crucial to evaluate the type of interaction and its potential impact on our results. We will explore this topic in greater detail when analyzing the Interaction Graphs.

**Post hoc Tests**

Similar to the previous tab, the Post Hoc Tests tab includes a pickerInput widget that allows users to select a specific sensory attribute for analysis. For this example, we analyze the attribute pink\_color Figure 11, the Least Significant Difference (LSD) method was chosen for this analysis due to its sensitivity and simplicity.

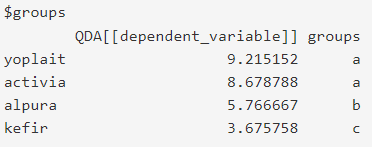


Figure 11. Post Hoc test

Products are grouped based on their mean scores for pink\_color. Yoplait and Activia (group "a") have similar mean scores and are not significantly different. Meanwhile, Alpura (group "b") and Kefir (group "c") are significantly different from both Yoplait and Activia, and also between them. Kefir has the lowest mean score and differs significantly from all other products. These results highlight Kefir as the product with the least intensity in the pink\_color attribute, while Yoplait and Activia have the most intense pink color.

**Reproducibility Graphs**

The reproducibility graphs in the application are generated using line plots, where the x-axis represents the Treatments being evaluated, and the y-axis displays the scores for the selected attribute. Each panel corresponds to a specific panelist, providing a detailed view of their individual performance. Repetitions are grouped and represented within each panel using distinct line styles or markers.

A selectInput widget allows users to choose a specific sensory attribute (e.g., pink\_color, sour\_flavor), ensuring a focused analysis of the reproducibility of panelists’ evaluations for the selected attribute. In an ideal case, the lines within a panel (for a specific panelist) should be parallel and closely aligned across repetitions. This indicates consistent scoring by the panelist across all treatments. Conversely, large variations between repetitions suggest panelist inconsistency, which may indicate a need for additional training.

Differences in score trends between treatments suggest that the panelist can effectively discriminate between the sensory attributes of the products. On the other hand, minimal differences between treatments may indicate low attribute variability or difficulty distinguishing between them.

By examining multiple panels, it is possible to identify panelists whose scores differ significantly between repetitions, as shown in Figure 12 for the panelist "Marian." A more consistent panelist, such as "Andrea" or "Adamaris," would exhibit scores that vary minimally across repetitions and treatments. Additionally, consistency in the pattern across multiple panelists indicates a well-trained and calibrated panel.

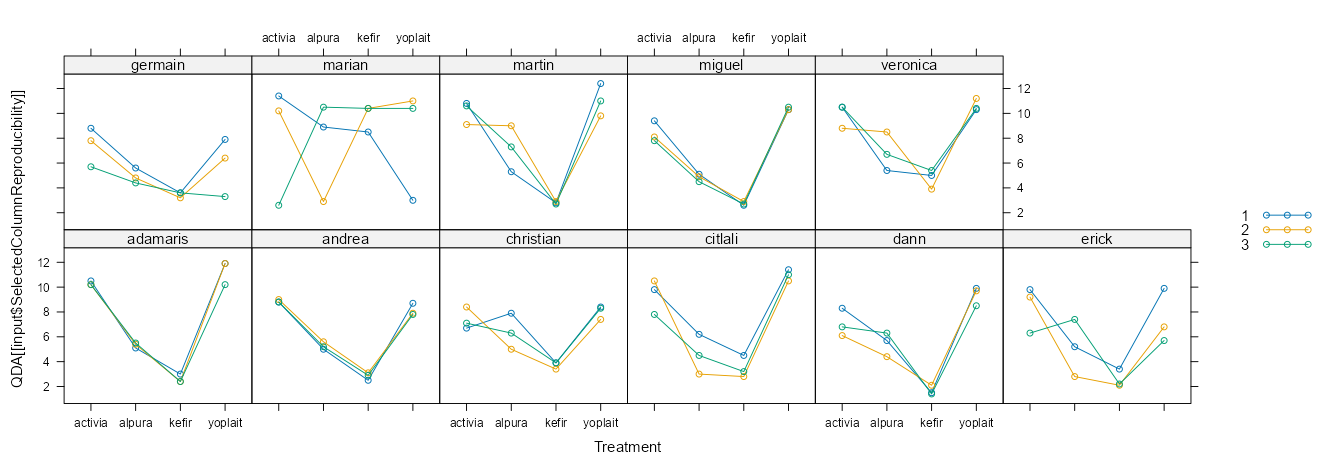


Figure 12. Reproducibility Graphs for pink\_color attribute

**Interaction graphs**

In the context of factorial experiments, an interaction effect occurs when the influence of one factor (e.g., treatment) on the response variable depends on the level of another factor (e.g., panelist or repetition). This means that the effect of a factor cannot be fully understood in isolation but is influenced by the other factor(s) in the experiment.

For example, in sensory analysis, a treatment’s sensory attribute may be perceived differently depending on the panelist. This variability is what we describe as interaction, and its presence can complicate the interpretation of main effects. When interaction is not significant, the main effects (e.g., treatment differences) can be interpreted directly and are considered reliable. In this case, the overall trends observed across all levels of the other factor (e.g., panelists) reflect meaningful differences. For instance, a treatment might show a positive effect for one panelist but a negative effect for another. In such cases, the average treatment effect may appear non-significant, even if meaningful differences exist at specific levels of the other factor. This phenomenon, known as masking, highlights the importance of examining the interaction effects before drawing conclusions about the main effects (Walpole *et al.*, 2012).

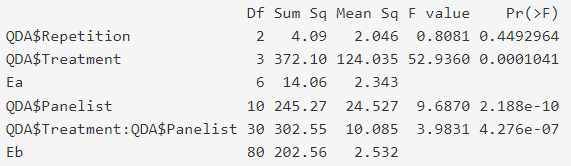


Figure 13. ANOVA split plot for sour\_falvor

Consider the sour\_flavor attribute shown in Figure 13, where the interaction between Treatment × Panelist is significant. This indicates that some panelists perceive the sour flavor differently across treatments. Such variability may stem from differences in training, calibration, or unique perceptions, all of which need to be addressed before interpreting treatment differences accurately. To analyze these interactions, the "Interaction Graphs" tab was developed. Similar to other sections, a selectInput widget allows users to choose a specific attribute (in this case, sour\_flavor), which automatically generates the corresponding interaction graphs (Figure 14).

When there is no interaction, the graphs display parallel lines, indicating consistent responses across treatments. However, two types of interactions can occur:

1. Magnitude Interaction: Lines are not parallel but do not intersect. In sensory evaluation, this type of interaction is generally not critical.
2. Cross Interaction: Lines intersect or are perpendicular to one another. This type of interaction is highly relevant in sensory evaluation, as it highlights significant variability among panelists.

In Figure 14, we observe three panelists ("Marian," "Martin," and "Erick") exhibiting cross interaction. This suggests these panelists require additional training for the sour\_flavor attribute. If further training is not feasible, the pickerInput tool included in this tab (Figure 15) can be used to exclude these panelists from all calculations, including graphs, ANOVA, PCA, and more. By excluding these panelists and revisiting the ANOVA table, the interaction effect is notably reduced, improving the reliability of the analysis.

**Radial plot**

The radial plot is a visualization tool used to compare the intensity of sensory attributes across treatments. It provides an intuitive way to understand how different treatments perform across multiple attributes simultaneously, making it easier to identify strengths and weaknesses in the sensory profiles of products. The axes radiating from the center represent the different sensory attributes (e.g., pink\_color, sweet\_aroma, sour\_flavor). Each treatment is represented by a distinct line or polygon, connecting the points for its mean scores across attributes.

**External Preference Map (EPM)**

The EPM is generated using a Principal Component Analysis (PCA), which reduces the multidimensional data from sensory evaluations into two primary components (e.g., Dim.1 and Dim.2). These components capture most of the variability in the dataset, providing a simplified view of the relationships between sensory attributes and treatments.

Points represent treatments (e.g., Yoplait, Activia), colored based on their categorical grouping (e.g., treatments). The bigger points are the centered data or mean. Vectors represent sensory attributes (e.g., pink\_color, sour\_flavor). The direction and length of a vector indicate the strength and influence of that attribute on the components. Points (treatments) close to a vector are influenced by the corresponding attribute.

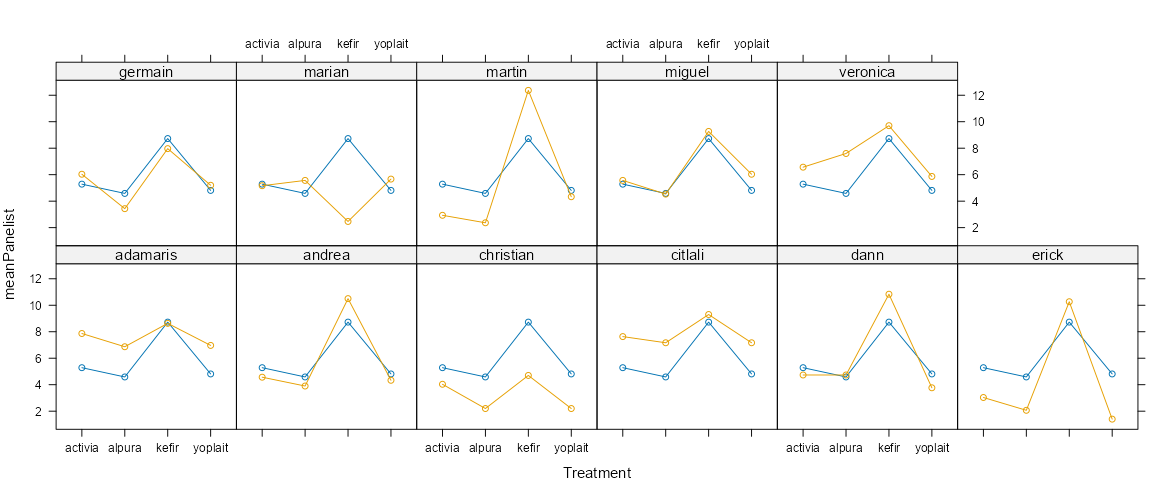


Figure 14. Interaction graphs for sour\_flavor

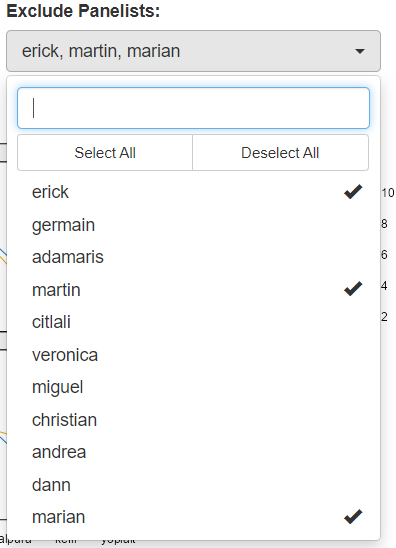


Figure 15. Tool for exclude panelist

1. **Conclusión**

This article has presented a comprehensive overview of a sensory evaluation application designed to facilitate the analysis of Quantitative Descriptive Analysis (QDA) data. By integrating robust statistical methodologies and interactive visualization tools, this application streamlines the sensory analysis workflow and provides actionable insights for product development and improvement.

Throughout the article, we emphasized the importance of tailoring statistical methods to align with the experimental design. For instance, the decision to use a split-plot ANOVA instead of relying on SensoMineR was based on the need for a more precise decomposition of variance and interaction effects, enhancing the reliability of the results.

The application not only simplifies the sensory analysis process but also ensures flexibility through interactive widgets like pickerInputs and selectInputs, allowing users to customize their analyses based on specific attributes or treatments. These features enable researchers to uncover deeper insights into their data, leading to better-informed decisions.

However, despite the extensive tools already included, I am aware that the application still requires improvements in certain aspects. For example, the current implementation of radial plots could benefit from more dynamic functionalities, such as real-time adjustments or interactive comparisons. Similarly, there is room to expand features that further simplify the analysis for non-technical users or provide deeper customization for advanced researchers.

**References**

Bartholomew, D. J., 2010. Analysis and Interpretation of Multivariate Data. International Encyclopedia of Education, 12–17. doi:10.1016/b978-0-08-044894-7.01303-8

Bartholomew, D. J. 2010. Principal Components Analysis. International Encyclopedia of Education, 374–377. doi:10.1016/b978-0-08-044894-7.01358-0

Chang W, Cheng J, Allaire J, Sievert C, Schloerke B, Xie Y, Allen J, McPherson J, Dipert A, Borges B (2024). shiny: Web Application Framework for R\_. R package version 1.10.0, <https://CRAN.R-project.org/package=shiny>.

de Mendiburu F (2023). \_agricolae: Statistical Procedures for Agricultural Research\_. R package version 1.3-7, <https://CRAN.R-project.org/package=agricolae>

Douglas Bates, Martin Maechler, Ben Bolker, Steve Walker (2015). Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, 67(1), 1-48. doi:10.18637/jss.v067.i01.

Greenacre, M., Groenen, P.J.F., Hastie, T. *et al*. Principal component analysis. Nat Rev Methods Primers 2, 100 (2022). <https://doi.org/10.1038/s43586-022-00184-w>

Hernandéz Montes, A. (2016).Evaluación Sensorial de productos Agroalimentarios. Texcoco, México: Universidad Autónoma Chapingo. 95-108

Husson F, Le S, Cadoret M (2023). \_SensoMineR: Sensory Data Analysis\_. R package version 1.27, <https://CRAN.R-project.org/package=SensoMineR>.

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2016

Kassambara A, Mundt F (2020). \_factoextra: Extract and Visualize the Results of Multivariate Data Analyses\_. R package version 1.0.7, <https://CRAN.R-project.org/package=factoextra>.

Kunst J (2022). \_highcharter: A Wrapper for the 'Highcharts' Library\_. R package version 0.9.4, <https://CRAN.R-project.org/package=highcharter>.

Kuznetsova A, Brockhoff PB, Christensen RHB (2017). “lmerTest Package: Tests in Linear Mixed Effects Models.” \_Journal of Statistical Software\_, \*82\*(13), 1-26. doi:10.18637/jss.v082.i13 <https://doi.org/10.18637/jss.v082.i13>.

Lenth R (2024). \_emmeans: Estimated Marginal Means, aka Least-Squares Means\_. R package version 1.10.6, <https://CRAN.R-project.org/package=emmeans>.

Meilgaard, M. C., Civille, G. V., & Carr, B. T. (2007). Descriptive Analysis Techniques. Sensory Evaluation Technique (págs. 209-210). New york: CRC Press. doi:https://doi.org/10.1201/b16452

Perrier V, Meyer F, Granjon D (2024). \_shinyWidgets: Custom Inputs Widgets for Shiny\_. R package version 0.8.7,

<https://CRAN.R-project.org/package=shinyWidgets>.

Sarkar D (2008). \_Lattice: Multivariate Data Visualization with R\_. Springer, New York. ISBN 978-0-387-75968-5, <http://lmdvr.r-forge.r-project.org>.

Sebastien Le, Julie Josse, Francois Husson (2008). FactoMineR: An R Package for Multivariate Analysis. Journal of Statistical Software, 25(1), 1-18. 10.18637/jss.v025.i01

Stone, H., Sidel, J., Oliver, S., Woolsey, A., & Singleton, R. C. (1974). Sensory Evaluation by Quantitative Descriptive Analysis. Food Technology(8), 24-32. doi:10.1002/9780470385036.ch1c

Walpole, R. E., Myers, R. L., Myers, S. L., & Ye, K. (2012). Probabilidad y estadística para ingeniería y ciencias. México: PEARSON EDUCACIÓN.

Wickham H, Bryan J (2023). \_readxl: Read Excel Files\_. R package version 1.4.3, <https://CRAN.R-project.org/package=readxl>.

Wickham H, François R, Henry L, Müller K, Vaughan D (2023). \_dplyr: A Grammar of Data Manipulation\_. R package version 1.1.4, <https://CRAN.R-project.org/package=dplyr>.

Xie Y, Cheng J, Tan X (2024). \_DT: A Wrapper of the JavaScript Library 'DataTables'\_. R package version 0.33, <https://CRAN.R-project.org/package=DT>.

Young, F. W. (09 de 01 de 2025). Universitat de València. Obtenido de https://www.uv.es/visualstats/vista-frames/help/lecturenotes/lecture13/biplot.html