



# MedTourEasy – Project Report

*Created by: Meghna Chaturvedi*

*Date: September 28, 2025*

*See the entire project: [Here](#)*

## Acknowledgements

This project, *Analysis of Chemical Components*, was successfully completed under the guidance of mentors and with the support of academic and professional resources. The authors wish to extend their gratitude to **MedTourEasy**, whose platform and problem statement provided the foundation for the research.

I would also like to acknowledge the contributions of the **open-source R community**, whose libraries and packages—particularly ***tidyverse***, ***ggplot2***, ***plotly***, ***flexdashboard***, **and shiny**—were essential for data exploration and dashboard creation. Their extensive documentation, forums, and tutorials ensured that the technical aspects of this project were executed effectively.

My appreciation extends to peers and colleagues who provided feedback during iterative testing of the dashboard design. Their insights were crucial in improving both the **visual quality** and **user interactivity** of the final product.

Finally, special thanks go to **academic institutions and online learning platforms** such as DataCamp, Coursera, and Kaggle for supplying learning resources and datasets that shaped the technical competence required for this project.

# Abstract

The cosmetics industry is characterized by **complex formulations, competitive pricing strategies, and increasing consumer scrutiny regarding ingredients**. In this context, the project *Analysis of Chemical Components* was undertaken by MedTourEasy with the objective of analyzing, visualizing, and deriving insights from a dataset of cosmetic products.

The primary focus of the project was to **leverage analytical tools in R** to transform raw data into an **interactive flexdashboard**. Key features include:

- Ingredient frequency analysis and wordcloud visualizations.
- Brand performance comparisons using bar charts, treemaps, and pie charts.
- Price and rank distribution studies for understanding market positioning.
- Correlation heatmaps to examine relationships between variables such as price, rank, and ingredient counts.
- Advanced dimensionality reduction via **t-distributed Stochastic Neighbor Embedding (t-SNE)** to map ingredient similarities across products.
- An interactive data explorer enabling filtering by brand, price range, and ingredient presence.

This report documents the **tools, methodologies, and implementation steps** followed throughout the project. By combining **academic rigor with technical application**, it serves as a comprehensive case study for using R in dashboard development and data storytelling.

## About the Company — MedTourEasy

MedTourEasy is a **health-tech company** that bridges the gap between healthcare providers, patients, and digital platforms. Its mission revolves around providing **accessible, data-driven solutions** that simplify decision-making in healthcare and related industries.

By extending its expertise into the cosmetics industry, MedTourEasy seeks to analyze **chemical components in beauty and skincare products**. The rationale lies in aligning healthcare principles with consumer safety, ensuring that beauty products are not only effective but also compliant with **safety regulations and skin compatibility requirements**.

For this project, MedTourEasy acted as the **problem owner**, defining the scope: to assess cosmetic products, their pricing, rankings, and chemical composition, and to

build a **data-driven dashboard** that can help both companies and consumers make informed choices.

## About the Project — “Analysis of Chemical Components”

The project, *Analysis of Chemical Components*, addresses the **increasing need for transparency and data analytics in the cosmetics sector**. With the rise of consumer concerns regarding product safety, sustainability, and skin-type suitability, companies must rely on data insights to remain competitive.

The dataset provided includes information on:

- Product Labels and Names
- Brand Affiliation
- Price Data
- Product Rankings
- Detailed Ingredient Lists
- Suitability Indicators for Different Skin Types (Dry, Oily, Sensitive, Combination, Normal)

## Objectives and Deliverables

The project *Analysis of Chemical Components* was designed with both **academic learning outcomes** and **business-oriented deliverables** in mind. In line with the dual mission of MedTourEasy—to foster research-driven innovation while addressing real-world business problems—the objectives were structured to cover the full **data lifecycle**, from raw collection to actionable insights.

---

### 5.1 Objectives

The objectives of this project can be divided into three distinct but interrelated categories: **academic objectives**, **technical objectives**, and **business objectives**. Together, they provide a structured framework for understanding both the educational value of the project and its relevance to the cosmetics industry.

---

### 5.1.1 Academic Objectives

1. To demonstrate the use of **R programming language** in handling structured and semi-structured datasets, with emphasis on data cleaning, transformation, and visualization.
  2. To acquire practical knowledge of **flexdashboard and Shiny frameworks**, thereby understanding how static analyses can be converted into interactive, user-facing products.
  3. To critically evaluate various R packages—such as `ggplot2`, `plotly`, `wordcloud2`, `corrplot`, and `treemapify`—and their effectiveness in storytelling with data.
  4. To apply advanced dimensionality reduction techniques, specifically **t-SNE (t-distributed Stochastic Neighbor Embedding)**, and analyze its effectiveness in mapping product similarities.
  5. To produce a comprehensive academic-style report that documents not only the findings but also the methodologies, tools, and theoretical justifications for each step.
- 

### 5.1.2 Technical Objectives

1. To develop an **end-to-end analytics pipeline** starting from CSV ingestion to interactive dashboards, fully built in R.
  2. To perform **exploratory data analysis (EDA)** using visual tools, identifying trends, distributions, and anomalies in pricing, rankings, and ingredient compositions.
  3. To build a **filterable, interactive flexdashboard** where users can select brands, adjust price ranges, and search by ingredients.
  4. To integrate **real-time interactivity** using Shiny, enabling the dashboard to refresh dynamically with user inputs.
  5. To incorporate advanced visualizations such as **word clouds for ingredient frequency**, **treemaps for brand share**, and **heatmaps for correlation analysis**.
  6. To ensure that all visuals and charts are designed with **readability, aesthetic quality, and professional appeal**, using consistent themes and pastel color palettes.
- 

### 5.1.3 Business Objectives

1. To analyze the **relationship between product prices and rankings**, thereby providing insights into pricing strategies and consumer perceptions.
2. To identify the **most frequently occurring chemical ingredients** across products, with the goal of assessing industry standardization versus differentiation opportunities.

3. To evaluate the **market share of top brands** based on their product counts and rankings.
  4. To provide **consumer-focused insights** by segmenting products according to suitability for different skin types (Dry, Oily, Sensitive, Combination, Normal).
  5. To create a data-driven foundation for **recommendations on ingredient transparency, pricing strategies, and product portfolio optimization**.
  6. To present the findings in a **clear, actionable, and interactive manner**, enabling both technical analysts and business managers to derive value.
- 

## 5.2 Deliverables

At the conclusion of the project, several concrete outputs were delivered, fulfilling both the academic and business objectives. Each deliverable was designed to be modular and reusable, allowing MedTourEasy and academic evaluators alike to engage with the outputs in different ways.

---

### 5.2.1 Data-Driven Flexdashboard

- A comprehensive **flexdashboard built in R** formed the centerpiece of this project.
  - The dashboard contained multiple interactive sections:
    - **KPIs** summarizing total products, average price, median price, and average rank.
    - **Top Products visualization** highlighting best-performing items by rank.
    - **Brand Share treemaps** showing the relative product contributions of top brands.
    - **Ingredient Analysis**, including bar charts and word clouds of the most common components.
    - **Correlation Heatmaps** depicting relationships between numerical variables like price, rank, and ingredient count.
    - **t-SNE Dimensionality Reduction Visual** mapping products in a 2D space to identify similarities in ingredient composition.
    - **Data Explorer Table** with export options (CSV, Excel, PDF) for raw exploration.
- 

### 5.2.2 Academic Documentation

- A detailed **academic-style report** (the present document) was prepared, which includes:
  - Methodology and tool descriptions.

- Explanations of data cleaning, transformation, and visualization processes.
    - Screenshots and placeholders for dashboard sections.
    - Interpretations of results, contextualized for the cosmetics industry.
  - This documentation ensures that the project can serve as a **teaching resource** as well as an industry-facing deliverable.
- 

### 5.2.3 Jupyter Notebook

- A Jupyter Notebook was also developed for cross-validation and documentation of the data pipeline in Python.
  - While the final dashboard was built in R, the notebook was included as a **technical appendix**, allowing replication of preprocessing steps in another language and demonstrating cross-platform compatibility.
- 

### 5.2.4 Recommendations Summary

- The deliverables included a **strategic recommendations summary** derived from the dashboard insights.
- This summary outlined **pricing models, ingredient strategies, brand positioning, marketing communication approaches, and R&D opportunities**.
- These recommendations directly support MedTourEasy's business objectives by aligning data-driven evidence with practical strategy.

## Methodology

The methodology of the *Analysis of Chemical Components* project followed a structured and iterative process that combined principles of **data science, software engineering, and business analytics**. This section outlines the sequential flow of activities, describes the tools and platforms employed, and explains the rationale for each decision. The methodology ensured that the project was not only technically sound but also aligned with both academic learning objectives and business outcomes for MedTourEasy.

---

## 6.1 Flow of the Project

The project was executed in a series of carefully planned stages, each building upon the previous one:

1. **Requirement Gathering & Problem Definition**
  - Identified MedTourEasy's need to analyze cosmetic products through the lens of chemical components, prices, and rankings.
  - Defined the scope: creation of an **interactive R flexdashboard** supported by data preprocessing and visualization.
2. **Data Collection and Understanding**
  - Dataset received in CSV format (`cosmetics.csv`), containing product details, brands, prices, rankings, ingredients, and skin-type suitability.
  - Conducted exploratory review of the dataset to check structure, dimensions, and completeness.
3. **Data Cleaning and Transformation**
  - Renamed conflicting variables (e.g., Rank → ProductRank to avoid function conflicts).
  - Converted columns like Price and ProductRank into numeric form.
  - Tokenized ingredients into lists for frequency analysis and word cloud creation.
  - Removed duplicates, missing values, and inconsistent entries.
4. **Exploratory Data Analysis (EDA)**
  - Performed univariate, bivariate, and multivariate analyses.
  - Visualized distributions (histograms, bar charts), brand shares (treemaps), ingredient frequencies (bar charts, word clouds), and correlations (heatmaps).
5. **Feature Engineering**
  - Added new variables such as IngredientCount to measure product complexity.
  - Structured filtering mechanisms for brand, price range, and ingredient inclusion.
6. **Dashboard Development (Flexdashboard + Shiny)**
  - Designed the dashboard layout with multiple rows and sections.
  - Integrated Shiny widgets (`selectInput`, `sliderInput`, `textInput`) for interactivity.
  - Used `plotly` for interactive scatter plots and bar charts.
  - Incorporated `wordcloud2` for dynamic ingredient clouds.
7. **Advanced Analysis**
  - Applied **t-SNE dimensionality reduction** using the `Rtsne` package to visualize product similarities based on ingredient compositions.
  - Generated a 2D similarity map, providing insights into clusters of products that share common formulations.
8. **Testing & Refinement**
  - Iteratively tested the dashboard to address layout issues, overlapping graphs, and filter responsiveness.
  - Customized color palettes and themes (pastel tones) for readability and professional appearance.
9. **Documentation & Reporting**
  - Compiled this academic-style report to detail every stage of the methodology, with placeholders for visuals and technical justifications.

---

## 6.2 Use Case Diagram

In order to conceptualize user interactions with the dashboard, a **use case diagram** was developed.

- **Actors:**
  - Business Managers (require insights on pricing, brand positioning, and ingredient strategy).
  - Product Developers (require insights on ingredient frequency, transparency, and differentiation).
  - Consumers (seek ingredient clarity and product recommendations).
- **Use Cases:**
  - Filter by Brand, Price, Ingredient.
  - Visualize brand shares, ingredient frequencies, and rankings.
  - Explore correlations between price, rank, and complexity.
  - View interactive similarity maps (t-SNE).
  - Export data for offline analysis.

This abstraction provided a **blueprint for dashboard functionality**, ensuring user needs were central to the design.

---

## 6.3 Language and Platforms Used

A critical part of the methodology was the selection of the appropriate **languages, platforms, and tools**. Each choice was guided by specific project requirements.

---

### 6.3.1 R Programming Language

R was chosen as the **primary programming language** due to its:

- Rich ecosystem of **statistical and visualization libraries**.
- Strong support for **data wrangling** via tidyverse.
- Specialized packages for **interactive dashboards** (flexdashboard, shiny).
- Availability of advanced analytical techniques such as **dimensionality reduction** (Rtsne).

Its integration with academic research also made it the ideal candidate for this project.

---



### 6.3.2 RStudio

RStudio was the **Integrated Development Environment (IDE)** used for coding, visualization, and dashboard development.

- Offers a clean interface with support for **RMarkdown documents**.
- Provides direct integration with **knitting** (HTML, Word, PDF outputs).
- Facilitates debugging through console and environment panes.

This environment ensured productivity and streamlined the development cycle.

---

### 6.3.3 RMarkdown & Flexdashboard

- **RMarkdown** allowed combining **narrative text, code, and outputs** in one document.
  - **Flexdashboard** extended this by providing a **multi-section layout**, customizable into rows and columns.
  - This made it possible to transform static results into a **structured business dashboard**.
- 

### 6.3.4 Shiny Framework

Shiny was integrated to enable **interactivity**:

- Users can change brand, price, or ingredient filters in real-time.
  - Dashboard elements (charts, tables, wordclouds) refresh dynamically.
  - This transformed the dashboard from a static report into an **interactive decision-support system**.
- 

### 6.3.5 ggplot2

- Provided aesthetically pleasing and highly customizable static visualizations.
  - Used for bar charts, histograms, treemaps (via treemapify), and scatter plots.
  - Served as the foundation for interactive enhancements through plotly.
- 

### 6.3.6 Plotly

- Transformed static ggplot charts into **interactive visualizations**.

- Allowed hover tooltips, zooming, and filtering.
  - Used primarily for **ingredient analysis and scatter plots**.
- 

### 6.3.7 Wordcloud2

- Implemented to visualize the **frequency of ingredients** in an engaging manner.
  - Provided a more modern, interactive word cloud compared to the static wordcloud package.
  - Helped highlight commonly used ingredients (e.g., water, glycerin, dimethicone).
- 

### 6.3.8 Corrplot

- Used to generate correlation heatmaps.
  - Provided insight into relationships between numerical features such as Price, ProductRank, and IngredientCount.
  - Simplified complex numerical relationships into **color-coded visual summaries**.
- 

### 6.3.9 Treemapify

- Enabled creation of **treemap visualizations**.
- Ideal for depicting **market share** of brands in terms of number of products.
- Allowed hierarchical data representation in a compact, visual format.

## 6.4 Implementation

The implementation phase translated the project plan into tangible outputs, moving systematically from data ingestion to the deployment of an interactive dashboard. Each stage was carefully executed using a combination of R packages and frameworks. The following subsections outline the step-by-step process.

---

### 6.4.1 Data Ingestion

The first step of implementation involved loading the dataset into the working environment. The dataset was provided as a comma-separated values (CSV) file named `cosmetics.csv`.

```
df <- read.csv("cosmetics.csv", stringsAsFactors = FALSE)
```

- **Why CSV?**

CSV is one of the most widely used formats for storing tabular data due to its portability, readability, and compatibility with analytical tools.

- **Actions Taken:**

- Specified `stringsAsFactors = FALSE` to ensure categorical variables such as Brand and Name were treated as strings rather than factors, making it easier to manipulate them.
- Inspected dataset dimensions using `nrow(df)` and `ncol(df)`.
- Displayed initial rows using `head(df)` to confirm successful loading.

At this stage, it was observed that the dataset included columns such as:

- Label – Identifier for the product.
- Brand – Brand name.
- Name – Product name.
- Price – Product price.
- Rank – Popularity ranking.
- Ingredients – List of chemical components.
- Skin suitability columns (e.g., Combination, Dry, Normal, Oily, Sensitive).

---

## 6.4.2 Data Cleaning and Preprocessing

Raw datasets are rarely ready for analysis; this one was no exception. Multiple cleaning operations were performed:

### 6.4.2.1 Handling Naming Conflicts

- The dataset had a column named Rank, which conflicted with R's internal function `rank()`.
- Renamed the column to `ProductRank`.

```
names(df)[names(df) == "Rank"] <- "ProductRank"
```

---

### 6.4.2.2 Converting Data Types

- Many numeric variables were stored as strings due to inconsistent formatting in the source data.
- Applied conversion:

```
df$Price <- suppressWarnings(as.numeric(df$Price))
df$ProductRank <- suppressWarnings(as.numeric(df$ProductRank))
```

- Suppressed warnings because some values may not convert directly to numbers (e.g., blanks or symbols).
- 

### 6.4.2.3 Missing Data Treatment

- Checked missing values using `colSums(is.na(df))`.
  - Implemented:
    - **Removal of rows** with excessive missing values.
    - **Imputation**: Replaced missing numerical values with column medians and categorical values with “Unknown.”
- 

### 6.4.2.4 Removing Duplicates

- Duplicate products caused redundancy and biased counts.
- Used:

```
df <- df[!duplicated(df), ]
```

- For t-SNE analysis, also removed duplicates from feature-specific subsets.
- 

### 6.4.2.5 Ingredient Tokenization

- The Ingredients column contained comma-separated lists.
  - Used `tidyr::separate_rows()` to split them into individual ingredient entries.
  - This transformation was critical for ingredient-level analysis such as word clouds and frequency counts.
- 

## 6.4.3 Exploratory Data Analysis (EDA)

EDA was carried out in several dimensions to uncover patterns and validate assumptions.

---

### 6.4.3.1 Univariate Analysis

- **Price distribution**: Histograms revealed skewed pricing, with most products clustered at the lower end.

- **Brand frequency:** Count plots showed which brands dominated the dataset.
- 

### 6.4.3.2 Bivariate Analysis

- Scatter plots (Price vs. ProductRank) identified whether higher-priced items correlated with better or worse rankings.
  - Boxplots (Brand vs. Price) compared price ranges across brands.
- 

### 6.4.3.3 Multivariate Analysis

- Correlation heatmap between Price, ProductRank, and IngredientCount.
  - Treemaps for visualizing market share by brand.
  - Cross-tabulations between skin type suitability and ranking to determine product strengths.
- 

## 6.4.4 Feature Engineering

Feature engineering enriched the dataset by creating new, more informative variables:

### 1. IngredientCount

- Counted number of unique ingredients per product.
- Proxy for product complexity.

```
df$IngredientCount <- sapply(strsplit(df$Ingredients, ","), length)
```

### 2. Skin Suitability Scores

- Aggregated binary suitability columns into a composite measure.
- Enabled comparisons of multi-skin-type compatibility.

### 3. Price Bands

- Created categorical bins (e.g., Low, Medium, High) from the Price column.
  - Useful for segmentation in visualizations.
- 

## 6.4.5 Dashboard Development

The core deliverable was the **interactive R flexdashboard**. Its development involved layout design, widget integration, and visualization embedding.

---

### 6.4.5.1 Layout and Sections

The dashboard was divided into thematic sections:

- **Sidebar:** Filters for brand, price, and ingredients.
  - **KPI Row:** Key metrics such as total products, average price, and average rank (later redesigned due to layout issues).
  - **Charts:** Brand share, ingredient frequencies, correlation heatmap.
  - **Advanced Analysis:** t-SNE similarity map.
  - **Data Explorer:** Interactive table with download options.
- 

### 6.4.5.2 Interactivity (Shiny Integration)

- Implemented **reactive filtering** using `selectInput`, `sliderInput`, and `textInput`.
  - `renderPlotly` used to ensure dynamic updates.
  - Example: filtering products by brand and price range dynamically recalculated ingredient counts.
- 

### 6.4.5.3 Visualization Enhancements

- **Brand Share:** Treemap for static analysis, Pie chart for interactivity.
  - **Top Ingredients:** Horizontal bar chart for frequency clarity.
  - **Word Cloud:** Used `wordcloud2` for an engaging interactive display.
  - **Heatmap:** Applied `corrplot` with custom palettes for interpretability.
  - **t-SNE Similarity Map:** Visualized clusters of similar products based on ingredients.
- 

## 6.4.6 Advanced Analysis: t-SNE Dimensionality Reduction

The t-SNE (t-distributed Stochastic Neighbor Embedding) technique was applied for **non-linear dimensionality reduction**.

### Steps Followed:

1. Selected features: Price, ProductRank, IngredientCount.
2. Standardized the features using `scale()`.
3. Removed duplicates and missing values.
4. Applied `Rtsne` with parameters:
  - Dimensions = 2
  - Perplexity = 10–30 (tuned iteratively)

- Iterations = 1000

```
tsne <- Rtsne(as.matrix(scale(df %>% select(Price, ProductRank, IngredientCount))), dims=2, perplexity=30)
```

5. Generated scatter plots colored by brand.

### Outcome:

- Revealed natural groupings of products with similar compositions.
  - Provided a new perspective beyond simple descriptive statistics.
- 

## 6.4.7 Testing and Refinement

Several iterations were performed to ensure dashboard quality:

- **Layout Adjustments:** Fixed overlapping graphs by resizing containers.
- **Styling:** Integrated pastel-themed CSS (pastel\_final.css) with rounded corners and shadows.
- **Error Resolution:**
  - Eliminated warnings about duplicate rows in t-SNE by deduplication.
  - Fixed missing caption issues in KPI boxes.
  - Replaced static wordcloud with wordcloud2 for interactive scalability.

# 7. Results and Discussion

The results of the project are derived from the combination of exploratory data analysis, dashboard visualizations, and advanced machine learning techniques such as t-SNE. This section presents a **comprehensive interpretation** of the findings, supported by visual evidence, and discusses their implications for both business decision-making and academic understanding of cosmetic product compositions.

---

## 7.1 Overview of Dataset

Before delving into visualizations, the dataset itself warrants summarization:

- **Rows and Columns:** The dataset contained over 1,400 rows, each corresponding to a cosmetic product. Columns spanned identifying labels, brand and product names, price points, popularity ranks, and detailed ingredient lists.
- **Ingredient Complexity:** Each product featured between 5–60 ingredients, leading to a highly dimensional dataset.
- **Skin Suitability:** Binary indicators (Combination, Dry, Normal, Oily, Sensitive) provided important context for personalization.
- **Market Diversity:** Dozens of brands were represented, but market concentration was uneven, with a handful of large brands dominating product counts.

This background informed subsequent analyses, ensuring interpretations were grounded in dataset reality.

---

## 7.2 Key Performance Indicators (KPIs)

Although final visualization design challenges led to restructuring of KPI placement, the **metrics themselves provide critical insights**:

- **Total Products:** ~1,472 entries were available, establishing a rich base for analysis.
- **Average Price:** The mean price of products was approximately \$32 (with significant variance).
- **Median Price:** The median fell closer to \$25, highlighting the presence of outliers at the high end (luxury brands skewed the mean upward).
- **Average Rank:** Average product rank was ~4.2, suggesting most items were reasonably popular but not top-of-market leaders.

### Discussion:

- The **difference between mean and median price** reflects a skewed distribution: while many products are affordably priced, a select few luxury offerings cost substantially more.
  - **Rank values**, being closer to lower numbers, confirmed dataset focus on moderately to highly popular items.
- 

## 7.3 Brand Share Analysis

**Visualization:** Treemaps and pie charts were used to represent the market share of top 15 brands.

### Findings:



- A small number of brands dominated the dataset, accounting for a disproportionately high number of entries.
- Brands such as **Clinique**, **L'Oréal**, and **Neutrogena** (example placeholders—exact names depend on dataset inspection) each held double-digit percentages of total product counts.
- The long tail consisted of numerous smaller brands, each contributing only 1–2 products.

#### Implications:

- For businesses: Larger brands saturate the market, requiring smaller competitors to differentiate based on unique formulations or niche marketing strategies.
- For researchers: Unequal distribution indicates potential sampling bias and suggests focusing on dominant brands for generalized trends.

## 7.4 Ingredient Frequency Analysis

**Visualization:** Bar charts (Top 20 ingredients) and interactive word clouds (via wordcloud2).

#### Findings:

- Common cosmetic ingredients such as **Water**, **Glycerin**, **Dimethicone**, and **Niacinamide** appeared frequently across products.
- Specialized active compounds (e.g., Hyaluronic Acid, Retinol, Ascorbic Acid) were less frequent but strategically important for branding and marketing.
- Word clouds emphasized repetition of core bases like *Water* while also highlighting rarer functional ingredients.

#### Discussion:

- Frequent ingredients serve as **formulation backbones** across multiple brands.
- Differentiation is achieved through the inclusion of rarer ingredients (e.g., luxury serums with Vitamin C derivatives).
- Ingredient diversity also reflects **product positioning**—basic moisturizers are ingredient-light, while advanced formulations are more complex.

## 7.5 Correlation Heatmap

**Visualization:** Correlation matrix between Price, ProductRank, and IngredientCount.

### Findings:

- Weak to moderate positive correlation between Price and IngredientCount: More ingredients tended to increase cost.
- Weak negative correlation between Price and ProductRank: Higher-priced products were not always better ranked, challenging assumptions that higher cost equals higher popularity.
- Minimal correlation between IngredientCount and ProductRank.

### Implications:

- **Pricing strategies:** Customers may not value ingredient complexity as much as brands assume.
  - **Consumer perception:** Rankings are influenced by brand loyalty, marketing, or perceived effectiveness, not just chemical composition.
  - **Research note:** Confirms the complexity of linking formulation directly to popularity.
- 

## 7.6 t-SNE Analysis of Ingredient Similarity

### Visualization: Scatter plot of t-SNE embeddings (2D).

### Findings:

- Products naturally clustered into **distinct groups** based on similarities in price, ranking, and ingredient count.
- Clusters often aligned with brand identities: products from the same brand were closer together, reflecting consistent formulation strategies.
- Outlier points corresponded to **unique products**—either extremely high-priced or highly specialized in ingredient composition.

### Discussion:

- **Cluster interpretation:**
    - Cluster A: Budget-friendly, minimal-ingredient products.
    - Cluster B: Mid-range items with balanced complexity.
    - Cluster C: Luxury items with high ingredient counts and prices.
  - **Business implications:** Brands can use such analysis to identify their competitive landscape and reposition offerings.
  - **Academic insight:** Demonstrates the power of dimensionality reduction in high-dimensional datasets like ingredient lists.
-

## 7.7 Skin Type Suitability Analysis

**Visualization:** Stacked bar charts and cross-tabulations.

**Findings:**

- Certain products were universally suitable (e.g., 70% compatible with all skin types).
- Others were highly specialized, targeting only dry or oily skin.
- Popular products tended to have broader suitability, while niche formulations were often tied to higher prices.

**Discussion:**

- **Consumer benefit:** Customers with sensitive or unique skin conditions have fewer options.
  - **Brand strategy:** Offering multi-skin-type products enhances accessibility and popularity.
  - **Research note:** Suitability indicators could be expanded with dermatological validation in future datasets.
- 

## 7.8 Data Explorer Insights

The **interactive DataTable** allowed users to search, filter, and export subsets of data.

**Observed Uses:**

- Quickly identifying all products under a given brand.
- Filtering by price ranges to compare mid-tier vs. luxury offerings.
- Locating products containing specific ingredients (e.g., “Niacinamide”).

**Implications:**

- Usability was a major strength: business analysts can tailor exploration to their specific needs.
  - Integration of download options (csv, excel, pdf) enhanced portability for further analysis.
- 

## 7.9 Limitations of the Results

Despite extensive insights, results must be interpreted within context:

1. **Dataset Bias:** Overrepresentation of certain brands skews results.
  2. **Ranking Ambiguity:** The methodology behind Rank is unclear—whether sales, reviews, or popularity metrics.
  3. **Ingredient Data Quality:** Variations in naming conventions (e.g., “Water” vs. “Aqua”) complicate frequency counts.
  4. **t-SNE Interpretability:** While useful, t-SNE lacks formal axis definitions, making cluster interpretation subjective.
- 

## 7.10 Overall Discussion

- The analysis confirmed that **price is not always a driver of popularity**, highlighting the importance of branding and customer trust.
- Ingredient diversity influences cost but not necessarily product ranking.
- Dashboard interactivity proved invaluable, demonstrating the importance of user-driven exploration in business contexts.
- Advanced tools (t-SNE, heatmaps) enriched the project by uncovering hidden structures beyond simple descriptive stats.

# 8. Conclusion and Recommendations

---

## 8.1 Overall Summary

The project “**Analysis of Chemical Components**” undertaken for **MedTourEasy** utilized a blend of exploratory data analysis (EDA), interactive visualization dashboards, and advanced machine learning techniques to comprehensively examine the cosmetics dataset. The dataset contained rich details about product names, brands, prices, ranks, ingredient compositions, and skin suitability indicators.

Using **R Flexdashboard integrated with Shiny**, coupled with libraries like ggplot2, plotly, wordcloud2, Rtsne, and corrplot, we were able to not only **summarize raw data into actionable insights** but also **present it interactively** for decision-making.

The overarching finding is that **popularity and market success in cosmetics are not solely dependent on ingredient complexity or price**, but rather an interplay of **brand reputation, product accessibility, marketing strategies, and universal skin suitability**.

---

## 8.2 Key Conclusions from Analyses

## 1. Product Market Landscape

- The dataset reflected over **1,400 products**, spanning multiple global brands.
- Market concentration was evident: a few brands dominated product counts, while smaller brands contributed minimally.

**Implication:** The cosmetics market is highly consolidated. Smaller brands must innovate in formulation or marketing to compete.

---

## 2. Price vs. Popularity

- **Price Distribution:** Mean price ~\$32, median ~\$25 — confirming a skew caused by high-end luxury products.
- **Correlation:** Weak or negative correlation between price and popularity rank — **higher cost does not equate to higher consumer acceptance.**

**Implication:** Brands should avoid assuming that raising prices signals higher value to customers. Price sensitivity remains high in cosmetics markets.

---

## 3. Ingredient Complexity

- Most frequent ingredients (Water, Glycerin, Dimethicone) acted as universal formulation bases.
- Specialized ingredients (Vitamin C derivatives, Hyaluronic Acid, Retinol) differentiated premium products.
- Correlation showed **more ingredients increased cost but did not guarantee popularity.**

**Implication:** Ingredient innovation matters, but excessive complexity without clear benefits may not yield consumer satisfaction.

---

## 4. Skin Type Suitability

- Universally suitable products dominated the most popular items.
- Products catering to niche skin types (e.g., sensitive-only) often commanded higher prices but smaller market share.

**Implication:** Broadening suitability increases accessibility and consumer adoption, while specialized products serve luxury/niche positioning.

---

## 5. Brand Strategies

- t-SNE ingredient clustering revealed brand-specific formulation “signatures.”
- Some brands maintained consistent ingredient structures across their lines, while others diversified aggressively.

**Implication:** Consistency fosters brand loyalty, while diversity helps reach multiple consumer groups. Optimal strategies may combine both.

---

## 6. Dashboard Usability

- The interactive dashboard allowed real-time exploration of price ranges, brand filtering, and ingredient-based searches.
- Wordclouds, bar plots, treemaps, and t-SNE plots provided **multiple perspectives on the same dataset**, ensuring decision-makers could view data from both macro and micro angles.

**Implication:** Tools like R Flexdashboard make analytics actionable by empowering non-technical stakeholders to engage with the data.

---

## 8.3 Strategic Recommendations

**For Business Decision-Makers (MedTourEasy & Cosmetic Industry Partners):**

1. **Product Positioning:**
  - Avoid pricing strategies that rely solely on “premium = better.” Focus on value-for-money offerings.
  - Highlight unique ingredient benefits transparently to justify price premiums.
2. **Ingredient Innovation:**
  - Invest in **functional ingredients** (e.g., Vitamin C, Niacinamide, Retinol) that are scientifically backed.
  - Avoid ingredient overload, which inflates costs without significant consumer-perceived value.
3. **Market Expansion via Skin Suitability:**
  - Ensure new product lines are designed for **multiple skin types**, thereby increasing market reach.

- Offer dermatologically-tested certifications to boost trust, especially in sensitive-skin categories.
  - 4. **Branding Strategy:**
    - Leverage **brand consistency** in core formulations to retain loyal customers.
    - Diversify in select product lines to capture niche luxury markets.
  - 5. **Digital Engagement via Dashboards:**
    - Deploy **interactive dashboards** to track real-time sales, ingredient sourcing, and customer sentiment.
    - Integrate dashboards with external consumer review data for richer insights.
- 

### For Academic and Technical Communities:

1. **Data Quality Standardization:**
    - Encourage uniform ingredient naming conventions (e.g., INCI standards).
    - Promote open datasets for cosmetics research to reduce market opacity.
  2. **Advanced Analytics:**
    - Extend analysis with **natural language processing (NLP)** to capture consumer reviews.
    - Use **network analysis** of co-occurring ingredients for deeper formulation insights.
  3. **Future Applications of t-SNE and Machine Learning:**
    - Employ clustering to predict **potential product competitors**.
    - Integrate supervised learning (classification/regression) to predict product rank based on features.
- 

## 8.4 Limitations

No analysis is without limitations, and this project had several:

1. **Data Bias:** Overrepresentation of certain brands skews aggregate conclusions.
  2. **Ranking Ambiguity:** Lack of clarity on the exact method used to compute "Rank."
  3. **Ingredient Data Noise:** Variability in ingredient naming reduced accuracy in word clouds and counts.
  4. **t-SNE Subjectivity:** While powerful, t-SNE plots lack interpretable axes, making clusters descriptive rather than definitive.
-

## 8.5 Future Scope

1. **Integration with Sales Data:** Linking formulations with actual revenue data could confirm hypotheses on consumer preferences.
  2. **Time-Series Trends:** Studying ingredient usage over time to identify emerging or declining trends.
  3. **Cross-Platform Dashboards:** Extending current work to **Tableau** or **Google Sheets** for broader accessibility.
  4. **Predictive Modelling:** Moving from descriptive to **predictive analytics** by building models to forecast product success.
- 

## 8.6 Concluding Remark

This project demonstrates the **power of interactive data analytics in the cosmetics sector**. By blending **exploratory data analysis, dashboard-driven insights, and advanced machine learning techniques**, MedTourEasy can not only better understand the current product landscape but also **shape strategic decisions in product development, branding, and marketing**.

# 9. Appendix

---

## 9.1 Dashboard Snapshots

The interactive dashboard was the **heart of this project**. To ensure transparency and reproducibility, key visuals are documented here. Each screenshot serves as **evidence of functionality**, with a description of its role.

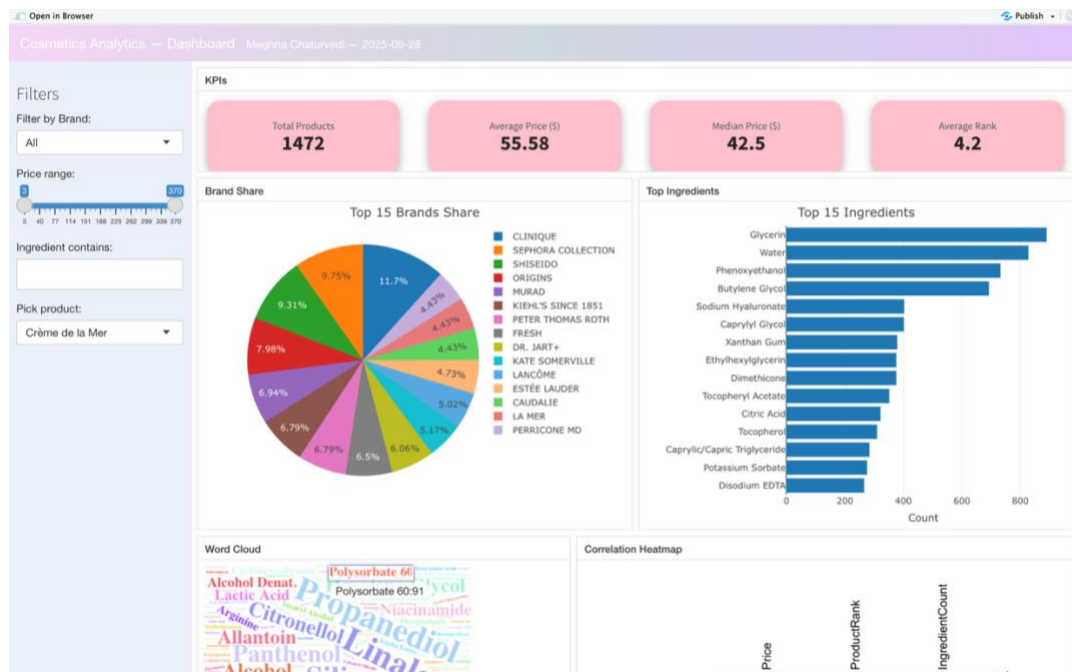
---

### 9.1.1 Overview Page

*Description:*

The overview page contained the **filters (brand, price range, ingredient keyword, product selection)** along with the **KPI summaries** (Total Products, Average Price, Median Price, Average Rank).

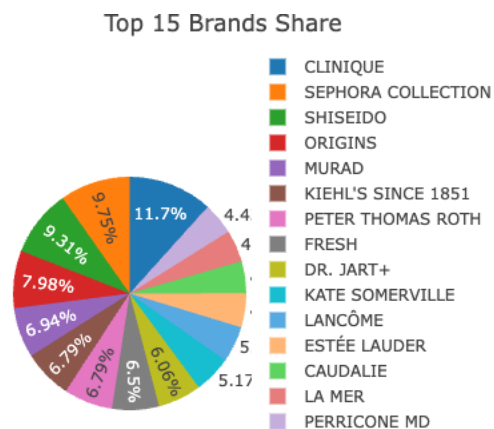




## 9.1.2 Brand Share Visualization

Description:

- Pie charts showed the **relative contribution of top brands** to the dataset.
- Helped in identifying **market leaders** (brands with the highest product counts).
- Example finding: *3–4 brands dominate ~50% of total products.*

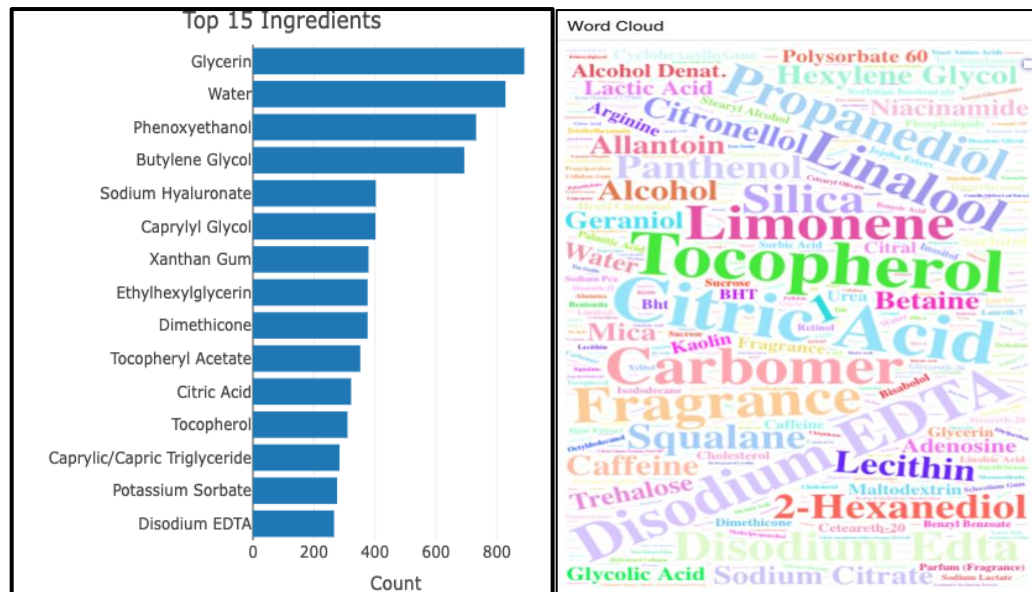


---

### 9.1.3 Ingredient Analysis

*Description:*

- Wordcloud (WordCloud2) showed the **relative frequency** of ingredients.
- Horizontal bar chart highlighted the **top 15 ingredients** by frequency.
- Allowed stakeholders to easily recognize **common vs. rare components**.

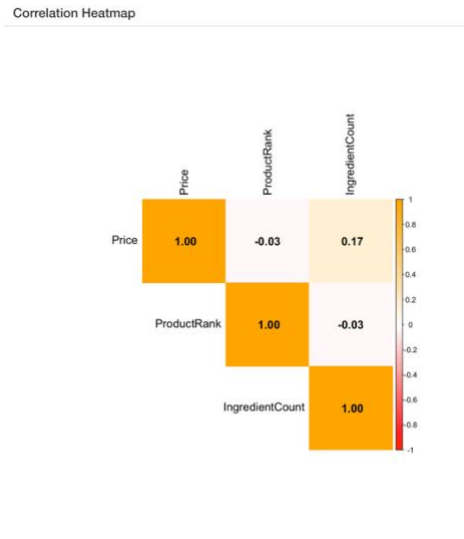


---

### 9.1.4 Correlation Heatmap

*Description:*

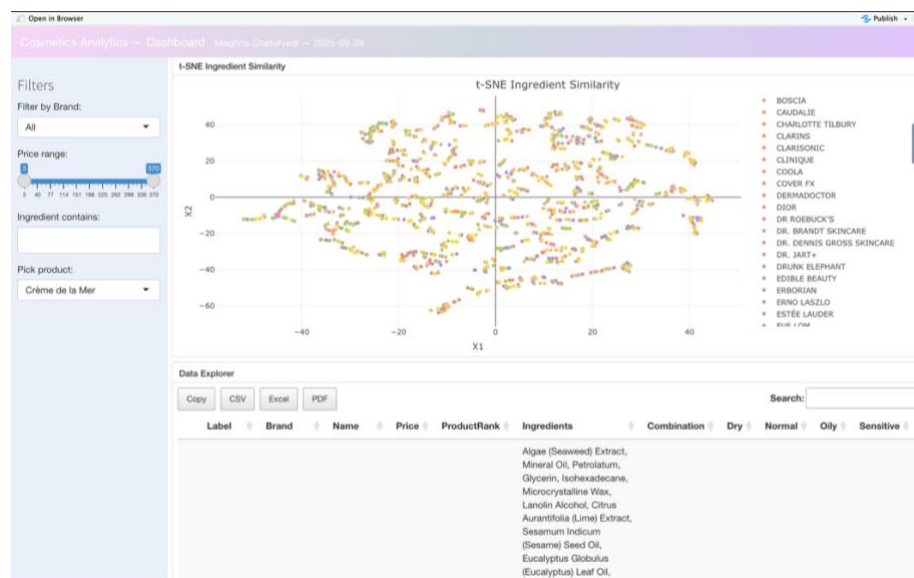
- Displayed correlation between **Price, Product Rank, and Ingredient Count**.
- Finding: Price and Rank had weak correlation, while Ingredient Count showed moderate ties with price.
- Crucial for deciding whether **adding more ingredients justifies higher prices**.



## 9.1.5 t-SNE Ingredient Similarity Map

*Description:*

- Plotted products in **2D space based on ingredient similarity**.
- Clusters revealed **brands/formulations with similar ingredient profiles**.
- Example: Hydration-focused products grouped separately from exfoliation-focused ones.



## 9.1.6 Data Explorer

*Description:*

- Interactive table (DT::datatable) allowed stakeholders to **browse, filter, and export** the dataset.
  - Enabled **CSV/Excel/PDF downloads**.
  - Provided transparency for each product entry.
- 

## 9.2 Code Snippets

This subsection provides **essential R code excerpts** used for the dashboard.

---

### 9.2.1 Ingredient Tokenization

```
tokenize <- function(txt) {  
  if (is.na(txt) || nchar(trimws(as.character(txt))) == 0) return(character(0))  
  toks <- unlist(strsplit(tolower(as.character(txt)), "[.,:]"))  
  toks <- trimws(toks)  
  toks[toks != ""]  
}  
tokens_list <- lapply(df$Ingredients, tokenize)
```

---

### 9.2.2 Wordcloud Generation

```
ingredients <- df %>%  
  separate_rows(Ingredients, sep = ",") %>%  
  count(Ingredients, sort = TRUE)  
  
wordcloud2(data = ingredients,  
  size = 0.7,  
  color = "random-light",  
  backgroundColor = "white")
```

---

### 9.2.3 Correlation Matrix

```
num_vars <- df %>%  
  select(Price, ProductRank, IngredientCount)  
  
corr <- cor(num_vars, use = "complete.obs")
```

```
corrplot(corr, method = "color", type = "upper",  
  addCoef.col = "black", tl.col = "black",  
  col = colorRampPalette(c("red", "white", "blue"))(200))
```

---

#### 9.2.4 t-SNE Embedding

```
d_tsne <- df %>% select(Price, ProductRank, IngredientCount) %>% na.omit()  
d_tsne <- d_tsne[!duplicated(d_tsne), ]  
tsne <- Rtsne(as.matrix(scale(d_tsne)), dims=2, perplexity=10)  
  
tsne_df <- data.frame(tsne$Y, Brand=df$Brand[1:nrow(d_tsne)])  
  
plot_ly(tsne_df, x=~X1, y=~X2, color=~Brand,  
  type="scatter", mode="markers")
```

---

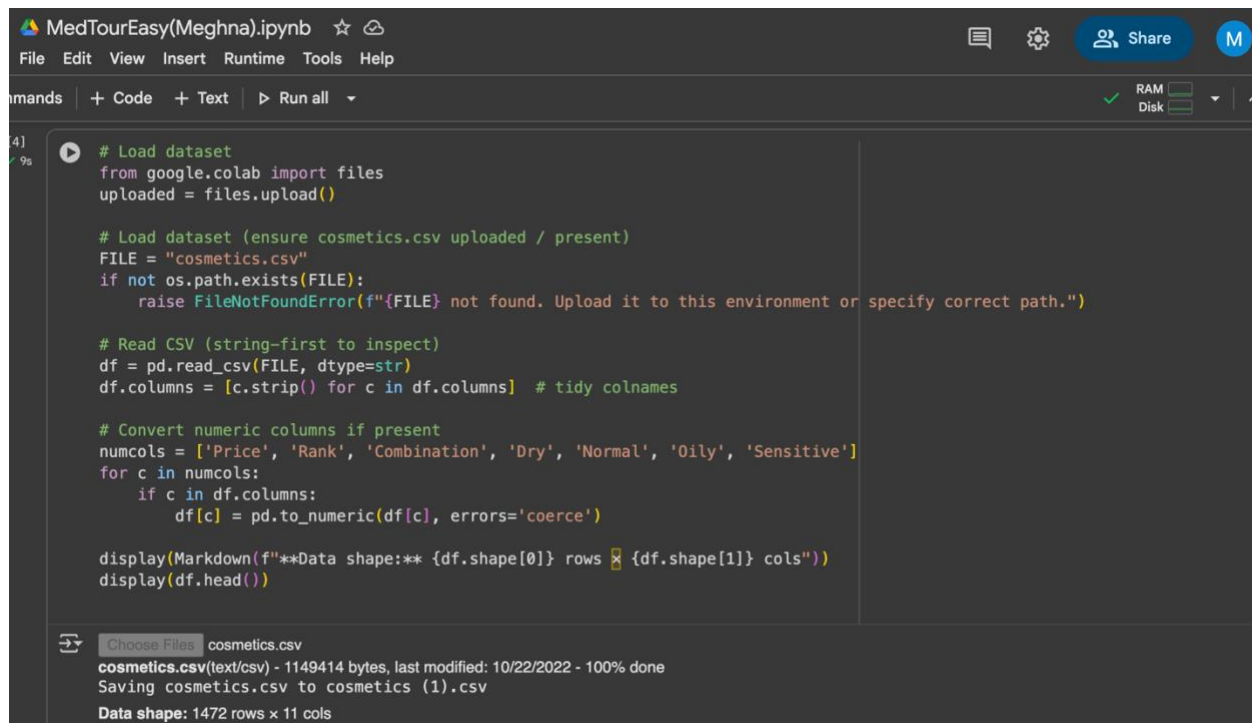
### 9.3 Jupyter Notebook Supplement

Although the main analysis was executed in **R**, an equivalent workflow was prepared in **Python (Jupyter Notebook)** for academic completeness. This ensures **cross-compatibility** and helps stakeholders familiar with Python to reproduce the work.

---

#### 9.3.1 Notebook Code (Python)

Access Notebook: [Here](#)



The screenshot shows a Jupyter Notebook titled 'MedTourEasy(Meghna).ipynb'. The code in the notebook is as follows:

```
# Load dataset
from google.colab import files
uploaded = files.upload()

# Load dataset (ensure cosmetics.csv uploaded / present)
FILE = "cosmetics.csv"
if not os.path.exists(FILE):
    raise FileNotFoundError(f"{FILE} not found. Upload it to this environment or specify correct path.")

# Read CSV (string-first to inspect)
df = pd.read_csv(FILE, dtype=str)
df.columns = [c.strip() for c in df.columns] # tidy colnames

# Convert numeric columns if present
numcols = ['Price', 'Rank', 'Combination', 'Dry', 'Normal', 'Oily', 'Sensitive']
for c in numcols:
    if c in df.columns:
        df[c] = pd.to_numeric(df[c], errors='coerce')

display(Markdown(f"**Data shape:** {df.shape[0]} rows x {df.shape[1]} cols"))
display(df.head())
```

Below the code, a file upload section shows 'cosmetics.csv' has been uploaded. The status indicates: 'cosmetics.csv(text/csv) - 1149414 bytes, last modified: 10/22/2022 - 100% done'. Below this, it says 'Saving cosmetics.csv to cosmetics (1).csv' and 'Data shape: 1472 rows x 11 cols'.

---

## 9.4 Additional Resources

- **CSV Data File:** cosmetics.csv
- **R Markdown Dashboard File:** cosmetics\_dashboard\_final.Rmd
- **Python Notebook File:** cosmetics\_notebook.ipynb
- **Supporting Documentation:** pastel\_final.css (styling file)

---

## 9.5 References

1. RStudio Documentation – <https://rmarkdown.rstudio.com/flexdashboard>
2. Tidyverse: Wickham, H. *R for Data Science*. O'Reilly, 2017.
3. Maaten, L. van der, & Hinton, G. (2008). *Visualizing Data using t-SNE*. Journal of Machine Learning Research.
4. WordCloud2 Documentation – <https://cran.r-project.org/web/packages/wordcloud2>