Deliverables

Tyler Yellott and Ashley Hutchings

2025-04-19

1. Concept and Purpose

The goal of the Alzheimer's Data Explorer is an interactive Shiny application designed to help users explore patterns and potential risk factors related to Alzheimer's Disease. Its core purpose is to make complex health data more accessible, especially for users without programming experience. By providing a visual, no-code interface, the app empowers users to investigate relationships between variables and Alzheimer's diagnosis using dynamic charts, summary statistics, and customizable data tables.

2. Initial Plan and Motivation

The motivation behind this app was to create a simple platform for visualizing the Alzheimer's Disease dataset in a way that was intuitive and accessible for individuals without coding experience. The goals was to provide a tool for any user to interact with real-world Alzheimer's data and uncover patterns related to diagnosis. The focus on Alzheimer's disease seeks to shed light on the complexities of diagnosing it based on multiple contributing factors. (See final page for design sketch.)

3. Dataset and Relevance to the Course

The dataset used is a publicly available Alzheimer's Disease dataset from Kaggle. It contains 35 variables related to demographics, cognitive test scores, physical health measures, and behavioral symptoms. The dataset provides real-world data with which to practice and apply techniques learned in this class such as data wrangling, visualizations and data summaries. The data contains both categorical and continuous variables, leveraging conditional logic. Moreover, this was an interesting, relevant but not overly complex dataset that was ideal for an initial project with R Shiny.

4. Key Features and User Interaction

Inputs

Users can provide 3 inputs: variable selector, chart type, and an option to facet by Diagnosis. Once a variable is selected, the options provided for chart type will dynamically change depending on the type of variable. If the variable is numeric then users will have the option to view a box plot or a histogram. If the variable is not numeric, then the chart type will be bar chart. Users also have a checkbox option to facet by Diagnosis which breaks down charts and summaries by diagnosis status. This option is selected by default.

Outputs

The main plot is a reactive chart rendered with ggplot2 and styled with a modern, dark theme. Dynamic text provides a description of the selected variable. A table with summary statistics shows mean, median, standard deviation, minimum, maximum, and count, split by Diagnosis if facet option is selected. Next there is an interactive data table that summarizes counts by the selected variable, with each column having the ability to sort and filter. Above the data table is a download button to allow users to download the data table in csv format.

5. Define Server Logic

The server handles everything behind the scenes that makes the app interactive. It reacts to user input and updates the visuals or tables in real time. We used reactive() to manage filtered data and make sure it changes automatically based on selections. Outputs like plots and tables are displayed using renderPlot() and renderTable(), so anytime the user adjusts a variable or filter, the app shows the updated results right away. For example, when users select different values for age or gender, the corresponding bar charts or histograms update instantly to reflect trends within that group. This setup helps users explore how different factors might relate to Alzheimer's outcomes in a simple and interactive way.

6. Examples of insights or analyses that can be obtained through your app

This app helps users explore the Alzheimer's data set in a simple, visual way. They can compare variables like age, gender, diet quality, etc. and see how those might relate to diagnosis or cognitive scores. For example, we saw that older age groups tended to have a higher proportion of Alzheimer's diagnoses, and that people with lower diet quality scores often had lower cognitive function scores. Users can also observe how education level might play a role in cognitive outcomes, or how diagnosis rates vary between males and females. By applying filters and changing the inputs, users can break the data down into different groups and spot trends without needing to do manual calculations. It's a helpful tool for getting quick insights from a pretty big data set.Even just exploring a few variables can uncover interesting relationships that would be easy to miss in raw data alone.

7. Link to GitHub

Here's the GitHub repo with all the files and code used to build the app:
https://github.com/ashley-hutchings/Shiny-App-R-HW