The **Tidyverse** is a collection of R packages that work in harmony to provide tools for data manipulation, visualization, and analysis. The key packages within the Tidyverse include **dplyr** (for data manipulation), **ggplot2** (for visualization), and **tidyr** (for data tidying). Here’s a breakdown of how each of these libraries works, along with examples.

**1. dplyr (Data Manipulation)**

dplyr is designed to simplify data manipulation by providing a range of functions that make it easy to select, filter, mutate, and summarize data. It works particularly well with **tibbles** (a modern form of data frames).

**Key Functions:**

* filter() – Subset rows based on conditions.
* select() – Pick specific columns.
* mutate() – Create or transform variables.
* summarise() – Aggregate data to compute summary statistics.
* group\_by() – Group data for further operations.

**Example 1: Filter and Summarize Data**

Suppose you have a dataset mtcars and you want to find the average miles per gallon (mpg) of cars with more than 4 cylinders.

r

library(dplyr)

mtcars %>%

filter(cyl > 4) %>%

summarise(mean\_mpg = mean(mpg))

This filters the cars with more than 4 cylinders and calculates their average mpg.

**Example 2: Adding a New Column with mutate()**

To convert the weight of the cars (wt) from 1000 lbs to kilograms:

r

mtcars %>%

mutate(wt\_kg = wt \* 0.453592)

This adds a new column wt\_kg which contains the car weights in kilograms.

**2. ggplot2 (Data Visualization)**

ggplot2 is a powerful and flexible package for creating a wide variety of plots. It follows a "grammar of graphics" philosophy, where plots are built layer by layer.

**Key Components:**

* **Data**: The dataset to visualize.
* **Aesthetics** (aes()) define how variables are mapped to visual properties (e.g., color, size, x and y positions).
* **Geoms** (geom\_ functions) are the specific type of plot (e.g., geom\_point() for scatter plots, geom\_bar() for bar charts).
* **Facets** to split plots by categories.

**Example 1: Scatter Plot with ggplot2**

Plotting the relationship between mpg and wt in the mtcars dataset:

r

library(ggplot2)

ggplot(mtcars, aes(x = wt, y = mpg)) +

geom\_point() +

labs(title = "Scatter plot of Weight vs MPG", x = "Weight (1000 lbs)", y = "Miles per Gallon")

This produces a scatter plot showing how weight and fuel efficiency are related.

**Example 2: Bar Plot with fill Aesthetic**

Creating a bar plot to show the count of cars by number of cylinders, with bars filled by gear:

r

ggplot(mtcars, aes(x = factor(cyl), fill = factor(gear))) +

geom\_bar(position = "dodge") +

labs(title = "Car Count by Cylinders and Gear Type", x = "Number of Cylinders", y = "Count")

This bar plot shows how many cars there are for each cylinder category, broken down by the type of gear.

**3. tidyr (Tidying Data)**

tidyr is used for reshaping and tidying data. It is especially useful for converting data between "wide" and "long" formats, or filling in missing values.

**Key Functions:**

* pivot\_longer() – Converts wide data into long format.
* pivot\_wider() – Converts long data into wide format.
* separate() – Splits a column into multiple columns.
* unite() – Combines multiple columns into one.

**Example 1: Pivoting from Wide to Long Format**

Suppose you have the following wide-format dataset on student scores:

r

data <- tibble(

student = c("Alice", "Bob", "Charlie"),

math = c(90, 85, 88),

science = c(85, 80, 90)

)

# Convert to long format

data\_long <- data %>%

pivot\_longer(cols = math:science, names\_to = "subject", values\_to = "score")

This converts the data from a wide format to a long format, where each student has multiple rows, one for each subject.

**Example 2: Separating a Column into Multiple Columns**

Imagine you have a column with date and time combined, and you want to separate it into two columns.

r

data <- tibble(datetime = c("2024-10-01 12:30", "2024-10-02 15:00"))

data\_separated <- data %>%

separate(datetime, into = c("date", "time"), sep = " ")

This splits the datetime column into two new columns: date and time.

**Summary:**

* **dplyr**: For data manipulation (filtering, selecting, mutating, summarizing).
  + Example: filter(cyl > 4) filters rows where the number of cylinders is greater than 4.
* **ggplot2**: For data visualization (scatter plots, bar charts, etc.).
  + Example: geom\_point() creates scatter plots, and geom\_bar() creates bar charts.
* **tidyr**: For data tidying (reshaping between wide and long formats).
  + Example: pivot\_longer() reshapes wide data to long format, and separate() splits columns.

These packages, when used together, offer a powerful and flexible framework for data science workflows in R.