

Module 1 – Rate Indication

Actuarial Data Manipulation with R – CAS Spring Meeting 2024

Denis Dreano

2024-05-05

Goal: Reproduce the main calculations in the final rate indication exhibit.

Content:

- Quick Introduction to R and RStudio
- Vector
- Data Frame and Tibbles
- Data Manipulations with dplyr:
 - `mutate()`
 - `summarize()`
 - `filter()`

- Open the Workshop folder as a new RStudio project
- Open `Workshop/part1_exercises_indication.R` and `Workshop/part1_solutions_indication.R`
- Useful shortcuts:
 - `Ctrl + Enter` to run a line or a selection
 - `F1` to get help on a function
 - `F2` to see what is in a variable
 - to restart the R session: `Ctrl + Shift + F10`
- Cheatsheets in `References/` folder:
 - RStudio Cheatsheet
 - Base R Cheatsheet

- A vector is a one-dimensional array that can hold numeric, character, or logical data.

```
# Create a numeric vector
x <- 1:5
premium <- c(100, 200, 300, 400, 500)
# Create a character vector
policy_id <- c("policy1", "policy2", "policy3", "policy4", "policy5")
# Create a logical vector
is_auto_policy <- c(TRUE, FALSE, TRUE, FALSE, TRUE)
```

- You can access elements of a vector using square brackets `[]`.

```
# Access the first element
```

```
premium[1]
```

```
[1] 100
```

```
# Access the second and third elements
```

```
premium[2:3]
```

```
[1] 200 300
```

```
# Access the last element
```

```
premium[length(premium)]
```

```
[1] 500
```

- You can assign names to the elements of a vector using the `names()` function.

```
# Create a named vector
premium <- c(100, 200, 300, 400, 500)
names(premium) <- c("policy1", "policy2", "policy3", "policy4", "policy5")
premium
```

```
policy1 policy2 policy3 policy4 policy5
    100     200     300     400     500
```

- R supports vectorized operations, which allow you to perform operations on entire vectors at once.

```
# Add two vectors element-wise
premium <- c(100, 200, 300, 400, 500)
loss <- c(50, 100, 150, 200, 250)
loss_ratio <- loss / premium
loss_ratio
```

```
[1] 0.5 0.5 0.5 0.5 0.5
```

Data Frames

- A data frame is a two-dimensional array that can hold multiple types of data.

```
# Create a data frame
library(dplyr)
df <- tibble(
  policy_id = c("policy1", "policy2", "policy3"),
  premium = c(100, 200, 300),
  loss = c(50, 100, 150)
)
```

- You can access columns of a data frame using the \$ operator.

```
# Access the name column of df
df$policy_id
```

```
[1] "policy1" "policy2" "policy3"
```

```
# Create a new column in df
df$is_auto <- c(TRUE, FALSE, TRUE)
```


Data Manipulation with dplyr: mutate()

```
# Mutate/create a new column based on existing columns
df_mutated <- mutate(df, loss_ratio = loss / premium)
df_mutated
```

```
# A tibble: 3 x 5
  policy_id premium  loss is_auto loss_ratio
  <chr>      <dbl> <dbl> <lgl>      <dbl>
1 policy1      100    50 TRUE         0.5
2 policy2      200   100 FALSE         0.5
3 policy3      300   150 TRUE         0.5
```

Data Manipulation with dplyr: select()

```
# Select specific columns
df_selected <- select(df, premium, loss)
df_selected
```

```
# A tibble: 3 x 2
```

	premium	loss
	<dbl>	<dbl>
1	100	50
2	200	100
3	300	150

```
# Select columns and rename
```

```
select(df, gross_written_premium = premium, incurred_loss = loss)
```

```
# A tibble: 3 x 2
```

	gross_written_premium	incurred_loss
	<dbl>	<dbl>
1	100	50
2	200	100
3	300	150

Data Manipulation with dplyr: summarize()

```
# Summarize data
df_summarized <- summarize(
  df,
  total_premium = sum(premium),
  total_loss = sum(loss),
  avg_loss_ratio = mean(loss / premium)
)
df_summarized
```

```
# A tibble: 1 x 3
  total_premium total_loss avg_loss_ratio
      <dbl>      <dbl>      <dbl>
1         600        300          0.5
```

- Cheat sheets in References/
 - `rstudio-ide.pdf`
 - `base-r-cheat-sheet.pdf`
 - `data-transformation.pdf`
- Wickham, H. et al. *R for Data Science* (r4ds.hadley.nz)
 - Chap. 3 Data Transformation
- Wickham, H. *Advanced R* (adv-r.hadley.nz)
 - Chap. 3 Vectors
 - Chap. 4 Subsetting