Interpretation of Data II

A00275664

2023-11-19

RMarkdown Guide

RMarkdown GP Applications

RMarkdown can be used by a GP in a variety of ways to optimise and enhance a GP's efficiency in day-to-day tasks Some ways that RMarkdown can be applied are;

- create reports for team-members
- provide calculations for tax returns
- provide an audit trail for reviews and audits
- communicate knowledge easily with non-technical staff members

Creating Reports with RStudio

RMarkdown files are clear and straightforward to read. For example;

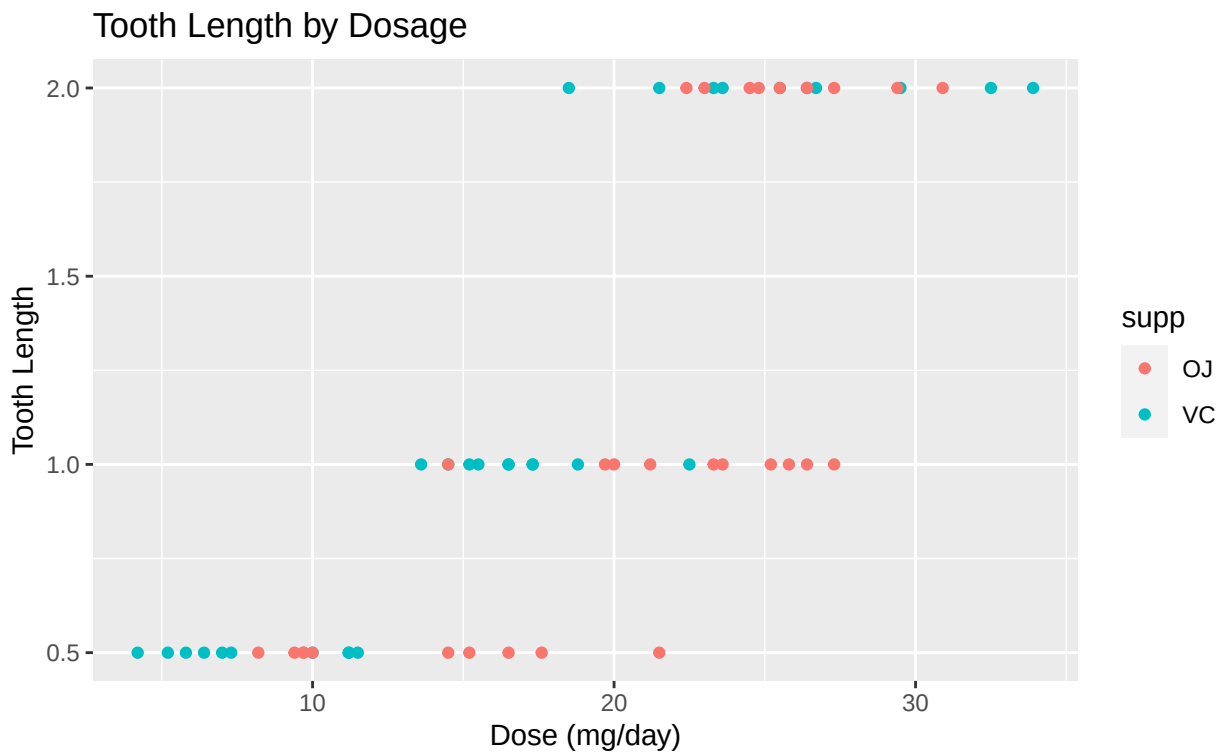
The average speed of a car (miles per hour) from the 1920s was

```
## [1] 15.4
```

Visualisation Features

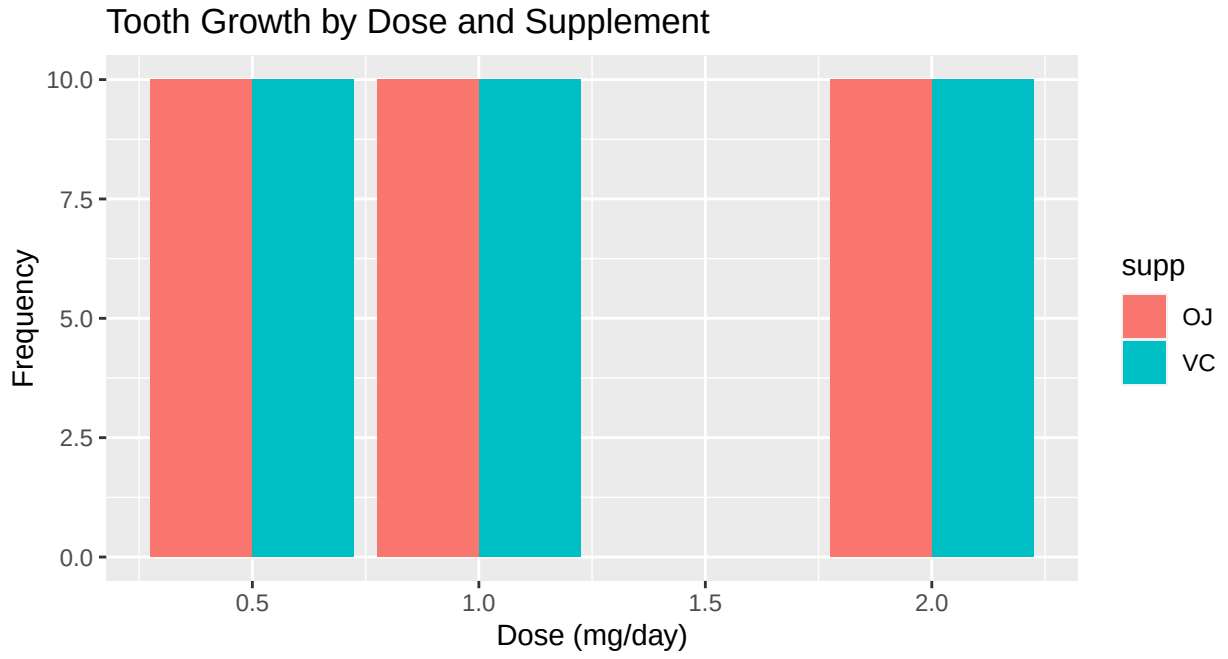
Charts and graphs can be produced from data to communicate a clearer picture of company performance. RMarkdown utilises the package **'ggplot2'** for data visualisation.

```
ggplot(ToothGrowth, aes(x=len,
                        y=dose,
                        color= supp))+
  geom_point() + labs(title="Tooth Length by Dosage",
                      x = "Dose (mg/day)",
                      y = "Tooth Length")
```



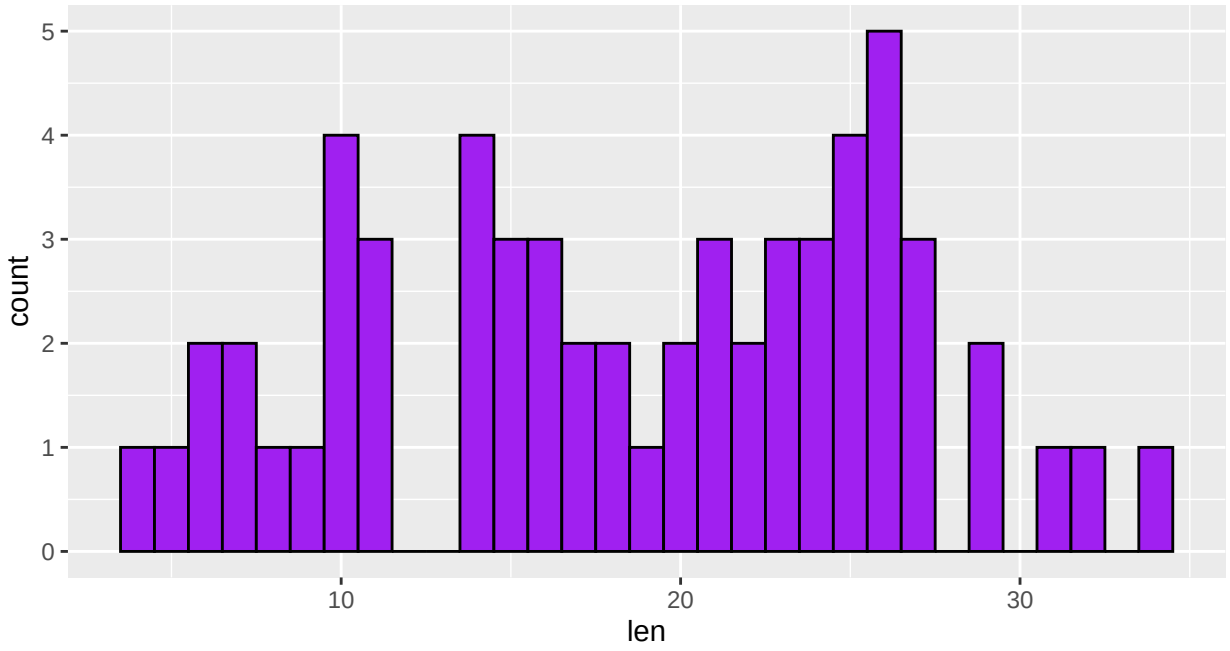
This can be shown a variety of ways useful for a business, such as a *bar chart*

```
ggplot(ToothGrowth, aes(x = dose, fill = supp)) +
  geom_bar(position = "dodge") +
  labs(title = "Tooth Growth by Dose and Supplement",
       x = "Dose (mg/day)",
       y = "Frequency")
```



Or a *histogram*

```
ggplot(ToothGrowth, aes(len))+
  geom_histogram(fill="purple", colour = "black", binwidth = 1)
```



Summary Statistics

RMarkdown and Rstudio can provide a quick analysis of your data

```
summary(airquality)
```

```
##      Ozone           Solar.R          Wind             Temp
##  Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :56.00
##  1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:72.00
##  Median : 31.50   Median :205.0   Median : 9.700   Median :79.00
##  Mean   : 42.13   Mean   :185.9   Mean   : 9.958   Mean   :77.88
##  3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00
##  Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00
##  NA's   :37       NA's   :7
##      Month            Day
##  Min.   :5.000   Min.   : 1.0
##  1st Qu.:6.000   1st Qu.: 8.0
##  Median :7.000   Median :16.0
##  Mean   :6.993   Mean   :15.8
##  3rd Qu.:8.000   3rd Qu.:23.0
##  Max.   :9.000   Max.   :31.0
##
```

This can also be narrowed down to a single variable

```
summary(airquality$Ozone)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    1.00   18.00   31.50   42.13   63.25  168.00      37
```

Creating an example dataframe

```
Patients<-data.frame(
  Pat_Id = 1:10,
  Age = c(18,19,20,22,30,40,50,65,70,71),
  Weight = c(65, 70, 80, 75, 85, 90, 120, 80, 95, 100)
  )
```

The packages **'dplyr'** can provide functions such as

- mutate() - create new columns based on a function using variables within the dataframe
- select() - select variables and create new datasets containing them
- filter() - select certain rows that meet certain criteria
- summarise() - provides summary statistics for a dataframe or variable
- arrange() - organise rows in a dataframe

Using the *filter()* function to find all patients weighing above 85 kilograms from the Patients dataframe

```
Patients1<-filter(Patients, Weight > 85)
```

The *mutate()* function can create a new variable from existing information in the dataframe

```
#Asssume height is equal for all
Height<-1.70
Patients2<- Patients %>%
  mutate(BMI = Weight / (Height*2))
```

RStudio allows you to return your output with the *print()* function

```
print(Patients2)
```

```
##    Pat_Id Age Weight      BMI
## 1       1  18     65 19.11765
## 2       2  19     70 20.58824
## 3       3  20     80 23.52941
## 4       4  22     75 22.05882
## 5       5  30     85 25.00000
## 6       6  40     90 26.47059
## 7       7  50    120 35.29412
## 8       8  65     80 23.52941
## 9       9  70     95 27.94118
## 10     10  71    100 29.41176
```

It is also possible to load your own data into RMarkdown using *read.csv()* and *read.excel()* functions, depending on the data file type

```
data <- read.csv("C:/Users/erink/Downloads/inpatientCharges.csv")
names(data)
```

```
##  [1] "DRG.Definition"
##  [2] "Provider.Id"
##  [3] "Provider.Name"
##  [4] "Provider.Street.Address"
##  [5] "Provider.City"
##  [6] "Provider.State"
##  [7] "Provider.Zip.Code"
##  [8] "Hospital.Referral.Region.Description"
##  [9] "Total.Discharges"
## [10] "Average.Covered.Charges"
## [11] "Average.Total.Payments"
## [12] "Average.Medicare.Payments"
```

Here is an example of the *summarise()* function finding the average patient discharges in the USA

```
average_Total.Discharges <- data %>%
                summarise(avg_Total.Discharges = mean(Total.Discharges))
print(average_Total.Discharges)
```

```
##   avg_Total.Discharges
## 1              42.7763
```

Lastly using the **Tidyr** package, we can create tidy data using

- gather() - removes specified columns and adds named columns
- pivot_longer() - reshapes data from a wider format to a longer format.
- spreading() - lengthens the observations across multiple rows.
- pivot_wider() - reshapes data from a longer format to a wider format.
- seperating() - splits columns into multiples columns.
- uniting() - combines multiple columns into a single column.

An example using *unite()* to combine the insurance provider's two address columns in the inpatientCharges dataset

```
data1<- unite(data, "Provider_Address",
              Provider.Street.Address, Provider.City,
              sep =", ")
```

References

E. Green, ‘Answer to "Change R Markdown plot width"’, Stack Overflow.
Accessed: Nov. 17, 2023. [Online]. Available: https://stackoverflow.com/a/23755105/22916988

Ssayols, ‘Answer to "Line breaks in R Markdown text (not code blocks)"’,
Stack Overflow. Accessed: Nov. 17, 2023. [Online].
Available: https://stackoverflow.com/a/43113246/22916988

‘Hospital Charges for Inpatients’. Accessed: Nov. 17, 2023. [Online].
Available: https://www.kaggle.com/datasets/speedoheck/inpatient-hospital-charges

file:///C:/Users/erink/Downloads/rmarkdown%20(1).pdf [R Markdown Cheat Sheet]