Session 1 - Introduction

Alex Mounsey

28/09/20

Exercise 1: Basic Data Structure, Summaries, and Plots

Consider the following data, collected by a farm management company, on the number of cattle and the number of sheep in nine farms in Devon:

$$Cattle = \begin{pmatrix} 348 \\ 407 \\ 1064 \\ 750 \\ 593 \\ 1867 \\ 471 \\ 935 \\ 1443 \end{pmatrix} \qquad Sheep = \begin{pmatrix} 110 \\ 179 \\ 303 \\ 173 \\ 182 \\ 458 \\ 151 \\ 140 \\ 222 \end{pmatrix}$$

Create two R objects for these data vectors, naming them `cattle` and `sheep`:

```r
cattle <- c(348, 407, 1064, 750, 593, 1867, 471, 935, 1443)
sheep <- c(110, 179, 303, 173, 182, 458, 151, 140, 222)
```

Save all observations, *except the 6th and 9th*, into two new vectors: `cattle_new` and `sheep_new`

```r
cattle_new <- cattle[-c(6, 9)]
sheep_new <- sheep[-c(6, 9)]
```

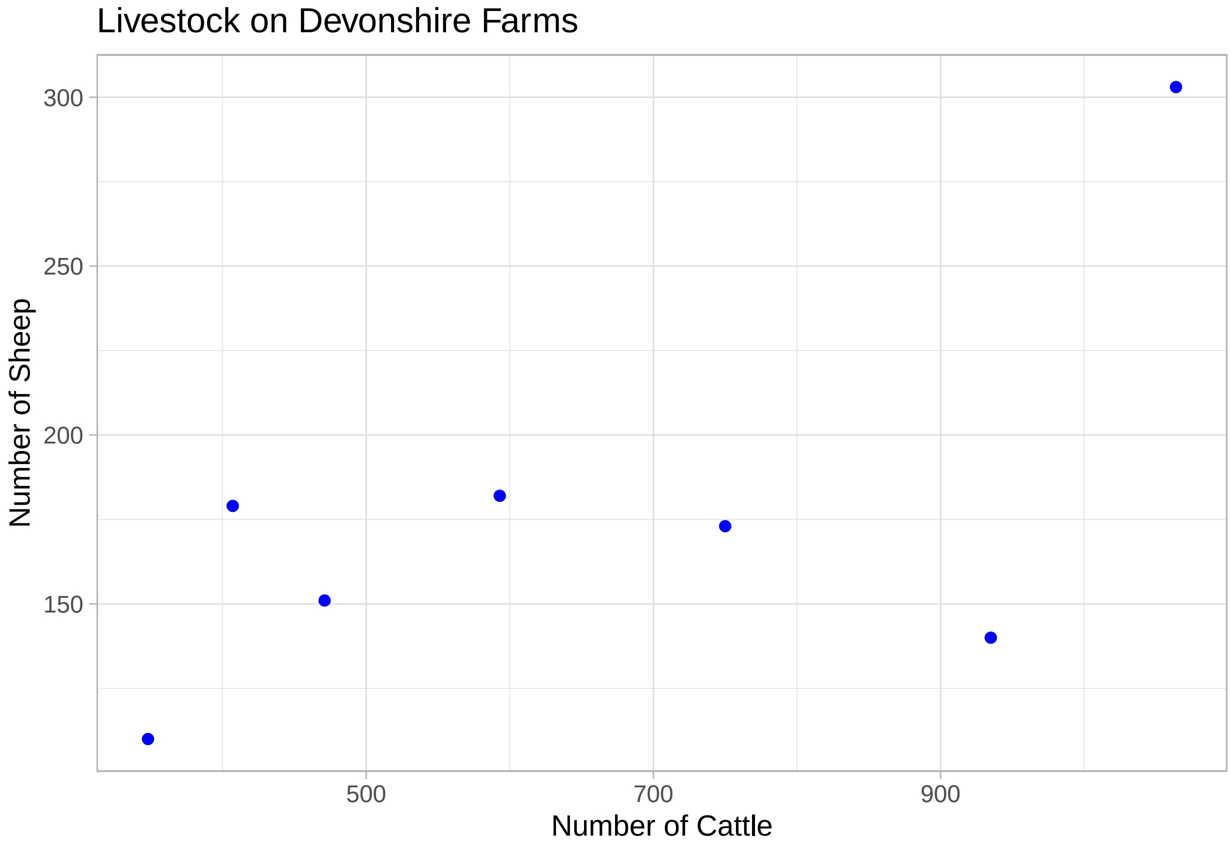
Create a data frame for `cattle_new` and `sheep_new`:

```r
df <- data.frame(cattle_new, sheep_new)
```

```
##   cattle_new sheep_new
## 1        348       110
## 2        407       179
## 3       1064       303
## 4        750       173
## 5        593       182
## 6        471       151
## 7        935       140
```

Create a scatterplot with the values in `cattle_new` on the horizontal axis and the values in `sheep_new` on the vertical axis using appropriate labels and title, colouring the dots in blue:

```
ggplot(df, aes(x = cattle_new, y = sheep_new)) +
  theme_light() + geom_point(col = 'blue') +
  labs(x = "Number of Cattle", y = "Number of Sheep",
       title = "Livestock on Devonshire Farms")
```



Calculate the sample mean and sample median of the values in both `cattle_new` and `sheep_new`:

```
df %>%
  summarise(mean_cattle_new = mean(cattle_new),
            mean_sheep_new = mean(sheep_new),
            median_cattle_new = mean(cattle_new),
            median_sheep_new = mean(sheep_new))
```

```
##   mean_cattle_new mean_sheep_new median_cattle_new median_sheep_new
## 1        652.5714       176.8571          652.5714         176.8571
```

Calculate the sum of the values in `cattle_new`, divided by the number of values in `cattle_new`:

```
sum_cattle_new <- sum(cattle_new)
len_cattle_new <- length(cattle_new)

sum_cattle_new / len_cattle_new
```

```
## [1] 652.5714
```

Exercise 2: Factors

Ten students are asked to respond to a question: **“Overall, I was satisfied with my experience of this module”** on a five point scale: *“Strongly Disagree”*, *“Disagree”*, *“Neutral”*, *“Agree”*, and *“Strongly Agree”*. The results were recorded numerically using the following encoding:

Numerical Encoding	Corresponding Response
1	Strongly Disagree
2	Disagree
3	Neutral
4	Agree
5	Strongly Agree

Here are the results from the ten students:

```
results <- c(5, 5, 2, 4, 3, 5, 5, 1, 2, 5)
```

Convert these results into a factor with the labels: *“Strongly Disagree”*, *“Disagree”*, *“Neutral”*, *“Agree”*, and *“Strongly Agree”*:

```
results_factor <- factor(results, levels = c(1, 2, 3, 4, 5),
                         labels = c(’Strongly Disagree’, ’Disagree’, ’Neutral’, ’Agree’,
                                    ’Strongly Agree’))
```

Tabulate these results:

```
results_table <- table(results_factor)
```

```
## results_factor
## Strongly Disagree          Disagree           Neutral             Agree
##                 1                 2                 1                 1
##    Strongly Agree
##                 5
```

Now, create a factor with the labels in the reverse order, and tabulate it:

```
reversed_results_factor <- factor(results, levels = c(5, 4, 3, 2, 1),
                                  labels = c(’Strongly Agree’, ’Agree’, ’Neutral’,
                                             ’Disagree’, ’Strongly Disagree’))

reversed_results_table <- table(reversed_results_factor)
```

```
## reversed_results_factor
##    Strongly Agree             Agree           Neutral          Disagree
##                 5                 1                 1                 2
## Strongly Disagree
##                 1
```

Exercise 3: Somewhat Harder

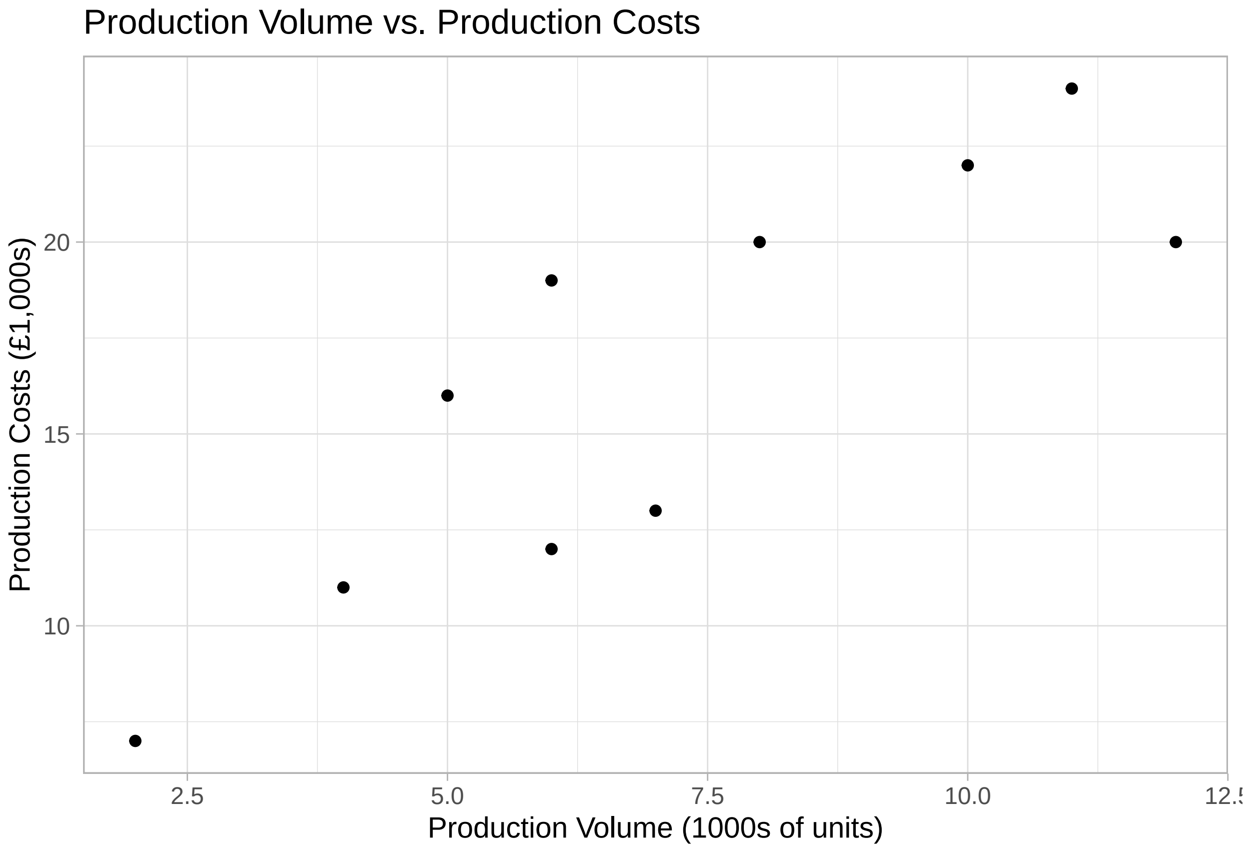
A small company has recorded the following data on production volume *(in 1,000s of units)* and production costs *(in £1,000s)* for the past ten months:

```
volume <- c(2, 4, 6, 6, 10, 8, 5, 7, 11, 12)
costs <- c(7, 11, 12, 19, 22, 20, 16, 13, 24, 20)
```

Create a data frame containing the variables `volume` and `costs`, before plotting this data. Comment on your plot:

```
# Create a data frame, using the provided vectors
df <- data.frame(volume, costs)

# Plot production volume vs. production costs
ggplot(df, aes(x = volume, y = costs)) +
  theme_light() + geom_point() +
  labs(x = "Production Volume (1000s of units)",
       y = "Production Costs (£1,000s)",
       title = "Production Volume vs. Production Costs")
```



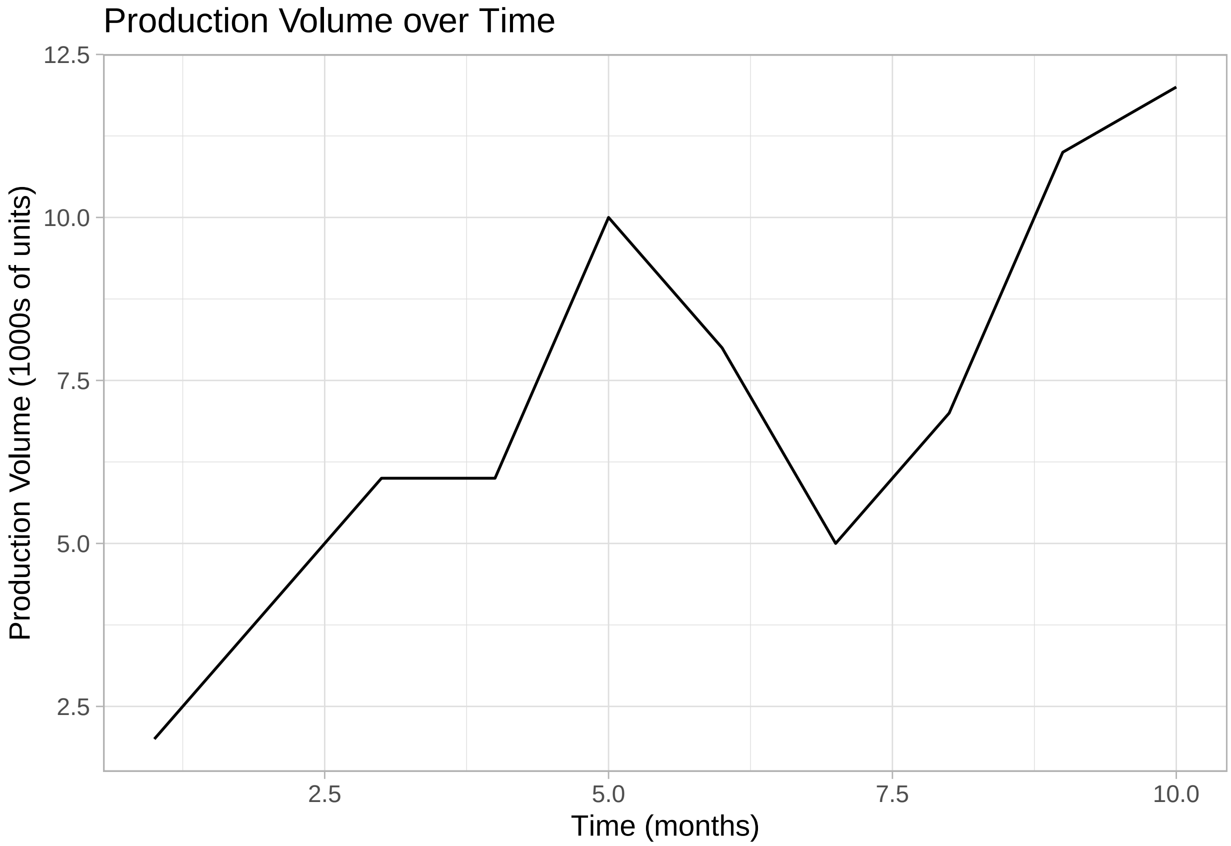
Comments:

There appears to be a positive linear relationship between the number of units produced and the production costs. This is to be expected: producing more units is going to cost more money (labor, materials, etc.).

Create a time-series plot of production volume, and comment on this plot:

```r
# Create a vector of months and add it to the data frame
month <- c(1:10)
df <- cbind(df, month)

# Plot production volume vs. time
ggplot(df, aes(x = month, y = volume)) +
  theme_light() + geom_line() +
  labs(x = "Time (months)", y = "Production Volume (1000s of units)",
       title = "Production Volume over Time")
```

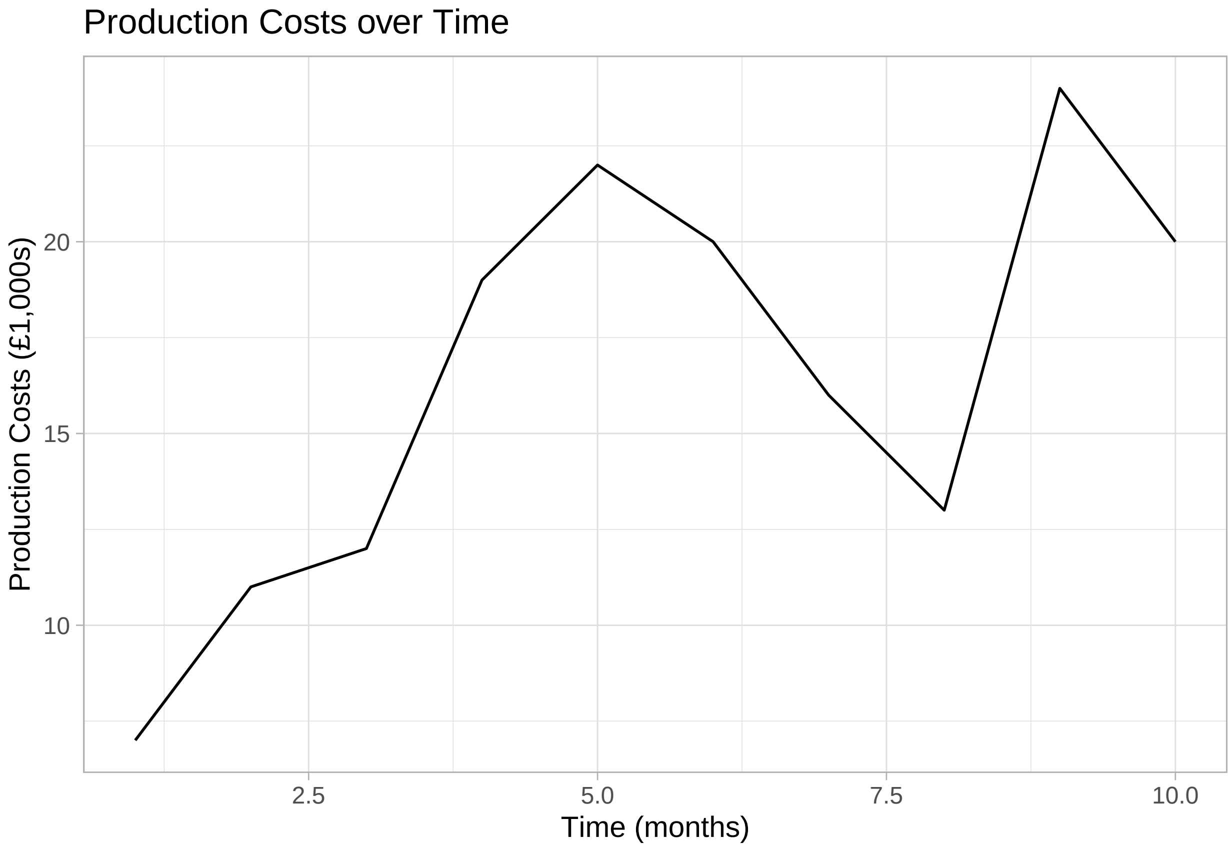


Comments:

The production volume steadily increases over time, this is likely due to the expansion of the business as they gain more clients. There appears to be dip in production volume between months `5` and `8`.

Create a time-series plot of production costs, and commend on this plot:

```r
ggplot(df, aes(x = month, y = costs)) +
  theme_light() + geom_line() +
  labs(x = "Time (months)", y = "Production Costs (£1,000s)",
       title = "Production Costs over Time")
```

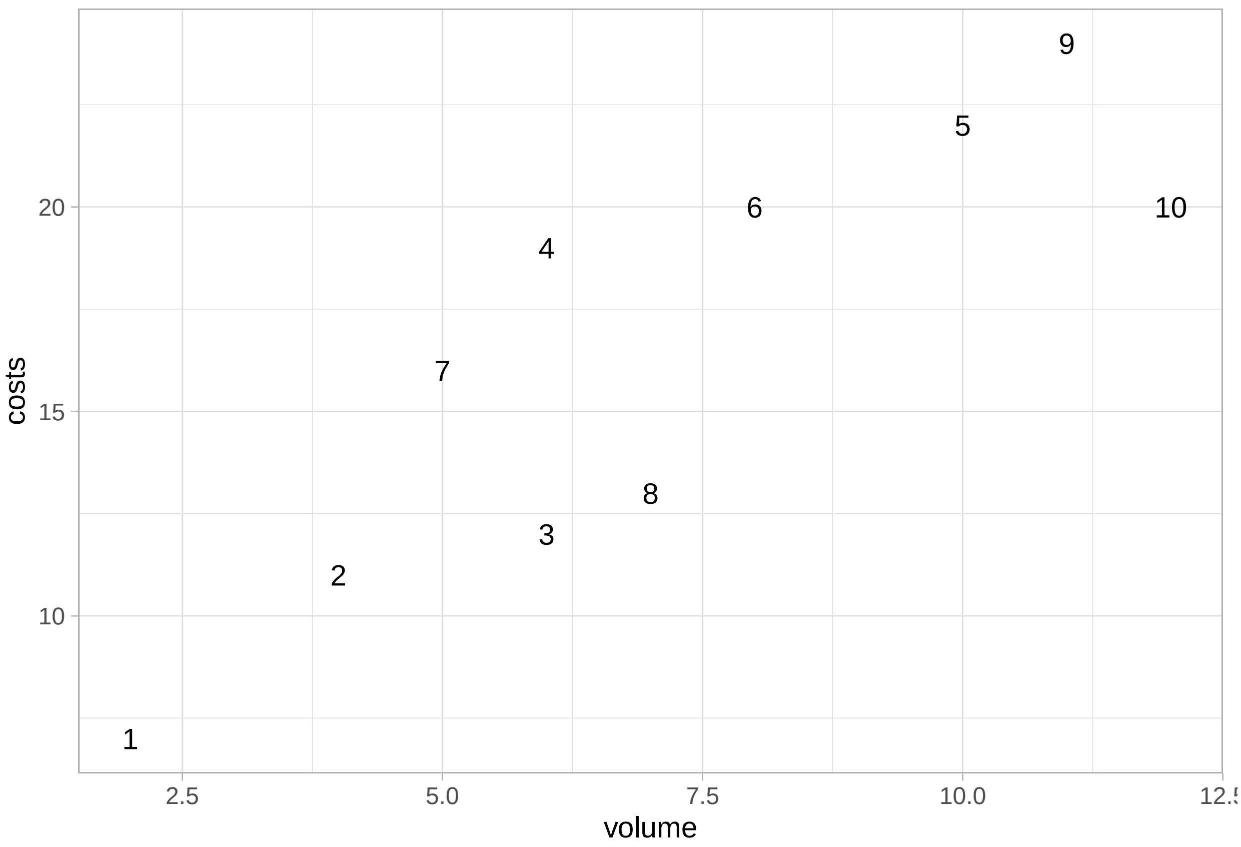


Comments:

As you might expect, the production costs increase over time in line with the production volume.

Create a plot showing production volume and production costs, with the month represented as text-based data points:

```
ggplot(df, aes(x = volume, y = costs, label = row.names(df))) +
  theme_light() + geom_text()
```



```
  labs(x = "Production Volume (1000s of units)",
       y = "Production Costs (£1,000s)",
       title = "Production Volume vs. Production Costs over Time",
       subtitle = "Months are represented as the data points’ text")
```

```
## $x
## [1] "Production Volume (1000s of units)"
##
## $y
## [1] "Production Costs (£1,000s)"
##
## $title
## [1] "Production Volume vs. Production Costs over Time"
##
## $subtitle
## [1] "Months are represented as the data points’ text"
##
## attr(,"class")
## [1] "labels"
```