BCB744: Intro R Test

Smit, A. J.

2025-03-17

ABOUT THE TEST

The Intro R Test will start at 8:30 on 17 March, 2025 and you have until 08:30 on 18 March to complete it. The Theory Test must be conducted on campus, and the Practical Test at home or anywhere you are comfortable working. The test constitutes a key component of Continuous Assessment (CA) and are designed to prepare you for the final exam.

The test consists of two parts:

Theory Test (30%)

This is a written, closed-book assessment where you will be tested on theoretical concepts. The only resource available during this test is RStudio, the R help system, your memory, and your mind.

Practical Test (70%)

In this open-book coding assessment, you will apply your theoretical knowledge to real data problems. While you may reference online materials (including ChatGPT), collaboration with peers is strictly prohibited.

ASSESSMENT POLICY

The marks indicated for each section reflect the relative weight (and hence depth expected in your response) rather than a rigid checklist of individual points. Your answer should demonstrate a comprehensive understanding of the concepts and techniques required, showing thoughtful integration of multiple R skills. Higher marks will be awarded for solutions that demonstrate not only technical correctness but also elegant code, insightful analysis, and clear communication of findings. We are assessing your ability to think systematically through complex data problems, make appropriate methodological choices, and present your findings in a coherent narrative that reveals meaningful patterns in the data. Your code should be well-structured, adequately commented, and reflect good programming practices.

Please refer to the Assessment Policy for more information on the test format and rules.

THEORY TEST

This is the closed book assessment.

Below is a set of questions to answer. You must answer all questions in the allocated time of 3-hr. Please write your answers in a neatly formatted Word document and submit it to the iKamva platform.

Clearly indicate the question number and provide detailed explanations for your answers. Use Word's headings and subheadings facility to structure your document logically.

Naming convention: `Intro_R_Test_Theory_YourSurname.docx`

Question 1

You are a research assistant who have just been given your first job. You are asked to analyse a dataset about patterns of extreme heat in the ocean and the possible role that ocean currents (specifically, eddies) might play in modulating the patterns of extreme sea surface temperature extremes in space and time.

Being naive and relatively inexperienced, and misguided by your exaggerated sense of preparedness as young people tend to do, you gladly accept the task and start by exploring the data. You notice that the dataset is quite large, and you have no idea what's happening, what you are doing, why you are doing it, or what you are looking for. Ten minutes into the job you start to question your life choices. Your feeling of bewilderment is compounded by the fact that, when you examine the data (the output of the `head()` and `tail()` commands is shown below), the entries seem confusing.

```
fpath <- "/Volumes/OceanData/spatial/processed/WBC/misc_results"
fname <- "KC-MCA-data-2013-01-01-2022-12-31-bbox-v1_ma_14day_detrended.csv"
data <- read.csv(file.path(fpath, fname))
```

```
> nrow(data)
[1] 53253434

> head(data)
           t     lon    lat      ex    ke
1 2013-01-01 121.875 34.625 -0.7141 2e-04
2 2013-01-01 121.875 34.625 -0.8027 2e-04
3 2013-01-02 121.875 34.625 -0.8916 2e-04
4 2013-01-02 121.875 34.625 -0.9751 2e-04
5 2013-01-03 121.875 34.625 -1.0589 3e-04
6 2013-01-03 121.875 34.625 -1.1406 3e-04

> tail(data)
                  t     lon    lat     ex      ke
53253429 2022-12-29 174.375 44.875 0.4742 -0.0049
53253430 2022-12-29 174.375 44.875 0.4856 -0.0049
53253431 2022-12-30 174.375 44.875 0.4969 -0.0050
53253432 2022-12-30 174.375 44.875 0.5169 -0.0050
53253433 2022-12-31 174.375 44.875 0.5367 -0.0051
53253434 2022-12-31 174.375 44.875 0.5465 -0.0051
```

You resign yourself to admitting that you don't understand much, but at the risk of sounding like a fool when you go to your professor, you decide to do as much of the preparation you can do so that you at least have something to show for your time.

a. What will you take back to your professor to show that you have prepared yourself as fully as possible? For example:

- What is in your ability to understand about the study and the nature of the data?
- What will you do for yourself to better understand the task at hand?
- What do you understand about the data?
- What will you do to aid your understanding of the data?
- What will your next steps be going forward?

b. What will you need from your professor to help you understand the data and the task at hand so that you are well equipped to tackle the problem?

[15 marks]

Answer

- I am able to understand what the concept of 'extreme heat' is, and what ocean eddies are – all I need to do is find some papers about it and do broad reading around these concepts. So, I will start by reading up on these concepts.
- I can see from the columns that there appears to be three independent variables (`lon`, `lat`, and `t`) and two dependent variables (`ex` and `ke`). I will need to understand what these variables are, and how they relate to each other. It is easy to see that `lon` and `lat` are the longitude and latitude of the data points, and that `t` is the date of the data point. I will need to understand what the `ex` and `ke` variables are, and how they relate to the `lon` and `lat` variables. Presumably `ex` and `ke` are the extreme heat and ocean eddies, respectively. I'll confirm with the professor.
- Because I have `lon` and `lat`, I can make a map of the study area. By making a map of the study area for one or a few days in the dataset, I can get a sense of the spatial distribution of the data. I can also plot the `ex` and `ke` data to see what the data look like. Because the data cover the period 2013-2022, I know that I can create a map for each day (a time-series analysis might eventually be needed?), and that is probably where the analysis will takle me later once I have confirmed my thinking with the professor. If I am really proactive and want to seriously impress the professor, I'll make an animation of the data to show the temporal evolution of revealed patterns in the data over time. This will clearly show the processes operating there. A REALLY informed mind will be able to even go as far as understanding what the analysis shoud entail, but, admittedly, this will require a deep subject matter understanding, which you might not possess at the moment, but which is nevertheless not beyond your reach to attain without guidance.
- I can conclude that the data reveal some dynamical process (I infer 'dynamical' from the fact that we have time-series data, and time-series reveal dynamics).
- Knowing what the geographical region is from the map I created and what is happening there that might be of interest to the study, I can make some guesses about what the analysis will be.
- FYI, what basic reseach would reveal include the following (not for marks):
  - you'd see that it is an ocean region south of South Africa;
  - once you know the region covered, you can read about the processes operating in the region that the data cover;
  - because the temperature spatially defines the Agulhas Current, you can infer that the study is about the Agulhas Current
  - plotting `ke` will reveal eddies in the Agulhas Current;
  - you can read about the Agulhas Current and its eddies and think about how eddies might affect the temperature in the region – both of these are dynamical processes.
- I will need to understand what the data are telling me, and what the variables mean. I will need to understand what the `ex` and `ke` variables are, and how they relate to the `lon` and `lat` variables.

- Having discovered all these things simply by doing a basic first-stab analyses, I can prepare a report of my cursory findings and draw of a list of things I know, toghether with suggested further avenues for exploration. I will take this to the professor to confirm my understanding and to get guidance on how to proceed.
- I will also add a list of the things I cannot know from the data, and what I need to know from the professor to proceed.
- There is also something strange happening with the data. It seems that there are duplicate data entries (two occurrences of each combination of `lat` x `lon` x `t` resulting in duplicated values for each spatio-temporal point of `ke` and a pair of dissimilar values for `ex`). I will need to understand why this is the case. Clearly this is incorrect, and this points to pre-processing errors somewhere. I will have to ask the professor to give me access to all pro-processing scripts and the raw data to see if I can trace the error back to its source.
- If I was this professor, I'd be immensepy mpressed by tyour proactive approach to the problem. You are showing that you are not just a passive learner, but that you are actively engaging with the data and the problem at hand. This is a very good sign of a good researcher in the making. In my mind, I'd seriously think about finding you a salary for permanent employment in my lab.

Question 2

Please translate the following code into English by providing an explanation for each line:

```
monthlyData <- dailyData %>%
    dplyr::mutate(t = asPOSIXct(t)) %>%
    dplyr::mutate(month = floor_date(t, unit = "month")) %>%
    dplyr::group_by(lon, lat, month) %>%
    dplyr::summarise(temp = mean(temp, na.rm = TRUE)) %>%
    dplyr::mutate(year = year(month)) %>%
    dplyr::group_by(lon, lat) %>%
    dplyr::mutate(num = seq(1:length(temp))) %>%
    dplyr::ungroup()
```

In your answer, simply refer to the line numbers (1-9) before each line of code and provide an explanation for each line.

[10 marks]

Answer

- Line 1: The variable `monthlyData` is created by starting with `dailyData`, which is a dataset containing daily records.
- Line 2: The `mutate()` function is used to convert the column `t` (presumably a date or timestamp) into a POSIXct datetime format. This ensures that `t` is stored in a standardised date-time format suitable for time-based operations.
- Line 3: The `mutate()` function is again used to create a new column `month`, which is derived from `t`. The `floor_date()` function rounds down the date to the first day of the corresponding month, effectively extracting the month from `t`.
- Line 4: The `group_by()` function groups the dataset by `lon` (longitude), `lat` (latitude), and `month`. This means subsequent operations will be performed separately for each unique combination of these three variables.
- Line 5: The `summarise()` function computes the mean temperature (`temp`) for each group. The

`na.rm = TRUE` argument ensures that missing values (`NA`) are ignored in the calculation.
- Line 6: The `mutate()` function creates a new column, `year`, extracting the year from the `month` column. This provides an explicit reference to the year of each data entry.
- Line 7: The `group_by()` function is applied again, but this time only by `lon` and `lat`. This modifies the grouping structure to remove the month grouping while retaining spatial grouping.
- Line 8: `The mutate()` function adds a new column, `num`, which assigns a sequence of numbers (`1:length(temp)`) to the grouped data. This effectively creates an index for each record within each longitude-latitude group.
- Line 9: The `ungroup()` function removes all grouping, ensuring that further operations on `monthlyData` are performed on the entire dataset rather than within groups.

Question 3

What is 'Occam's Razor'?

[5 marks]

Answer

Occam's Razor is sometimes attributed to the 14th-century philosopher William of Ockham, is a principle of parsimony that states: "Entities should not be multiplied beyond necessity." It is relevant to the BCB744 module because the principle of Occam's Razor is often interpreted as "the simplest explanation that sufficiently explains the data should be preferred over more complex alternatives." This is a nice guiding principle which might be useful in your research, especially when you are faced with multiple explanations for a phenomenon. The principle suggests that the simplest explanation is often the best one, and that more complex explanations should only be considered when the simpler ones fail to account for the data. But, keep in mind that biological systems tend to be complex, and oversimplifying an explanation may ignore important interactions or heterogeneities.

Question 4

Explain the difference between R and RStudio.

[5 marks]

Answer

Taken verbatim from Tangled Bank:

R is a programming language and software environment for statistical computing and graphics. It provides a wide variety of statistical (linear and non-linear modelling, classical statistical tests, time-series analysis, classification, clustering, multivariate analyses, neural networks, and so forth) and graphical techniques, and is highly extensible.

RStudio is an integrated development environment (IDE) for R. It provides a graphical user interface (GUI) for working with R, making it easier to use for those who are less familiar with command-line interfaces. Some of the features provided by RStudio include:

- a code editor with syntax highlighting and code completion;
- a console for running R code;
- a graphical interface for managing packages and libraries;
- an integrated tools for plotting and visualisation; and

- support for version control with Git and SVN.

R is the core software for statistical computing, like a car's engine, while RStudio provides a more user-friendly interface for working with R, like the car's body, the seats, steering wheel, and other bells and whistles.

Question 5

By way of example, please explain some key aspects of R code conventions. For each line of code, explain also in English what aspects of the code are being adhered to.

For example:

1. `a ← b` is not the same as `a < -b`. The former is correct because there is a space preceding and following the assignment operator (←, a less-than sign immediately followed by a dash to form an arrow); this has a different meaning from the latter, which is incorrect because there is no space between the less-than sign and the dash, reading as "a is less than negative b".

Hint: In your Word document, use a fixed-width font to indicate the code as a separate block which is distinct from the rest of the text.

[10 marks]

Answer

1. Proper use of indentation:

```
if (x > 0) {
  print("Positive number")
}
```

2. Use of meaningful variable names:

```
temperature <- 25
```

3. Use of comments to explain code:

```
# Calculate the mean temperature
mean_temp <- mean(temperature)
```

4. Consistent use of spacing around operators:

```
a <- b + c
```

5. Consistent use of compound object names:

A principles of writing clean and readable R code (or *any* code) is maintaining consistent variable naming conventions throughout a script or project. Mixing different naming styles – such as “snake_case” (words separated by underscores) and “camelCase” (capitalising the first letter of each subsequent word) – makes the code harder to read, maintain, and debug.

Examples:

```
# Example of consistent use of either convention:
my_variable <- 10 # snake case
another_variable <- 20 # camel case

# An example of inconsistent use of conventions:
myVariable <- 30 # camel case
yet_another_variable <- 40 # snake case

# This is also incorrect:
variable_one <- 13 # llowercase "one"
variable_Two <- 13 * 2 # uppercase "Two"
```

6. Avoiding the = as Assignment Operator

```
# Correct:
a <- 1

# Incorrect:
a = 1
```

7. Consistent use of spaces around # symbols in comments:

```
# This is correct:

# This is a comment
# This is another comment
# And another

# This is incorrect:

#This is a comment
# A comment?
#  Another comment
```

8. Correct use of `+` or `-` for unary operators:

```
# Correct:
a <- -b
```

9. Use of `TRUE` and `FALSE` instead of `T` and `F`:

```
# Correct:
is_positive <- TRUE
```

```
# Incorrect:
is_positive <- T
```

For more, refer to the tidyverse style guide.

Question 6

a. Explain why one typically prefers working with CSV files over Excel files in R.
b. What are the properties of a CSV file that make it more suitable for data analysis in R?
c. What are the properties of an Excel file that make it less suitable for data analysis in R?

[15 marks]

Answer

a)

CSV (Comma-Separated Values) files are preferred over Excel files due to their simplicity, compatibility, and efficiency in handling data. CSV files are stored as plain text, making them easy to read and write across different software and platforms. They do not contain proprietary formatting, formulas, or metadata, which minimises the risk of unintended data transformations.

Excel files (.xls, .xlsx) are proprietary and designed for spreadsheet applications, incorporating complex formatting, formulas, and visual formatting that can interfere with data processing in R. Unlike CSV files, which can be directly read using base R functions like `read.csv()`, Excel files require additional packages such as **readxl** for data extraction. Excel's tendency to automatically modify data types – such as converting text to dates or numbers – is annoying and introduces errors, making CSV a more reliable format for reproducible data analysis.

b)

- CSV files store data in a simple text-based format that ensures easy readability by both humans and computers.
- Each row represents a single record, and fields are separated by commas (or another delimiter) to ensure a consistent tabular format.
- CSV files can be opened and edited using a wide range of software, including text editors, spreadsheets (e.g., Excel, Google Sheets), and statistical tools (e.g., R, Python).
- R provides optimised functions like `read.csv()` (base R) and `read_csv()` (tidyverse) for quickly reading CSV files without additional dependencies.
- Unlike Excel, CSV files do not contain embedded formulas, formatting, figures, or macros and these properties reduce the risk of unintended data stuff-ups.
- Being plain text, CSV files are typically smaller in size compared to Excel files.

c)

- Excel files are stored in a format (.xls, .xlsx) that is specific to Microsoft Excel; special packages (e.g., **readxl**, **openxlsx**) are needed to read them in R.
- Excel often automatically formats data and changes numeric values to dates or rounding decimal

values. This can lead to errors in data analysis.
- Excel files support formulas, pivot tables, conditional formatting, and visual elements that may not be relevant for raw data processing in R.
- Users can store multiple sheets within a single Excel file and this makes it trickier to maintain a standardised structure when importing data into R.
- Excel files are not made for handling large datasets. Excel becomes very slow and is prone to crashing or memory limitations when dealing with 'big' data.
- Excel's binary files do not work with version control systems like Git.
- Excel files are complex and more prone to accidental modifications or corruption.

Question 7

Explain each of the following in the context of their use in R. For each, provide an example of how you would construct them in R:

a. A vector
b. A matrix
c. A dataframe
d. A list

Hint: See my hint under Question 5.

[20 marks]

Answer

(a) A vector in R is the simplest and most fundamental data structure. It is a one-dimensional collection of elements, all of the same type (e.g., numeric, character, or logical). Vectors can be created using the `c()` function. For example:

```r
# Creating a numeric vector
numbers <- c(1, 2, 3, 4, 5)

# Creating a character vector
names <- c("Acacia", "Protea", "Leucadendron")

# Creating a logical vector
logical_values <- c(TRUE, FALSE, TRUE)
```

(b) A matrix is a two-dimensional data structure where all elements must be of the same type. It is essentially an extension of a vector with a specified number of rows and columns.

```r
# Creating a matrix with 3 rows and 2 columns
my_matrix <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 3, ncol = 2)
```

(c) A dataframe is a two-dimensional data structure that can contain different data types in different columns (variables). It is the most commonly used data structure for data analysis in R and resembles a table with rows and columns.

```r
# Creating a dataframe
my_dataframe <- data.frame(
  Name = c("Acacia", "Protea", "Leucadendron"),
  Age = c(25, 30, 22),
  Height = c(85.5, 90.3, 78.0)
)
```

(d) A list is a flexible data structure that can store elements of different types, including vectors, matrices, dataframes, and even other lists. Unlike vectors and matrices, which require uniform data types, lists can contain heterogeneous elements.

```r
# Creating a list with different data types
# Uses the data created abobe, for example
my_list <- list(
  plants = my_dataframe,
  some_numbers = mu_matrix,
  other_numbers = numbers
  )
```

Question 8

a. Write a 150 to 200 word abstract about your Honours research project. In your abstract, draw attention to the types of data you will be expected to generate, and mention how these will be used to address your research question.
b. Explain which of the R data classes will be most useful in your research and why.
c. With reference to the abstract you wrote in Question 8.a, explain how you would visualise (or display your finding in tabular format) your research findings. Provide an example of how you would do this in R. Which of your research questions would be best answered using a visualisations or tables? What do you expect your visualisations or tables to show?
d. Provide an example of how you would create a plot or table in R. Generate mock code (it does not need to run) that you would use to create the plot or table.

Note 1: In the unlikely event that your research will not require visualisations or tables, please explain why this is the case and how you would communicate your findings.

Note 2: If you haven't defined your research project yet, describe a hypothetical project in your field of interest.

[30 marks]

Answer

This will have to be assessed based on the information quality produced in each abstract. Assign marks as follows:

a. Abstract: 30%
b. Data classes: 10%
c. Visualisation: 30%
d. Mock code: 30%

TOTAL MARKS: 110

PRACTICAL TEST

This is the open book assessment.

Below is a set of scripting problems to solve. You have 21 hours from the end of the Theory Test to complete this section Please write your code in an R script file and submit it to the iKamva platform by no later than 8:30 on Tuesday, 18 March 2025.

Please follow a clear structure (appropriate, clearly numbered headings and subheadings) in your code, including comments and explanations.

Ensure that all code runs without errors before submitting it – serious penalties will apply to non-functional scripts.

Naming convention: `Intro_R_Test_Practical_YourSurname.R`

Question 1

Download the `fertiliser_crop_data.csv` data.

The data represent an experiment designed to test whether or not fertiliser type and the density of planting have an effect on the yield of wheat. The dataset contains the following variables:

- Final yield (kg per acre) – make sure to convert this to the most suitable SI unit before continuing with your analysis
- Type of fertiliser (fertiliser type A, B, or C)
- Planting density (1 = low density, 2 = high density)
- Block in the field (north, east, south, west)

Undertake a full visual assessment of the dataset and establish which of the influential variables are most likely to have an effect on crop yield. Provide a detailed explanation of your findings.

[25 marks]

Answer

Question 2

The Bullfrog Occupancy and Common Reed Invasion data are here: `AICcmodavg::bullfrog` (i.e. the `bullfrogs` dataset resides within the **AICcmodavg** package, which you might have to install).

Create a tidy dataframe from the bullfrog data.

[10 marks]

Answer

Question 3

The Growth Curves for Sitka Spruce Trees in 1988 and 1989 data are here: `MASS::Sitka` and `MASS::Sitka89`.

Combine the two datasets and provide an analysis of the growth curves for Sitka spruce trees in 1988 and 1989. Give graphical support for the idea that i) ozone affects the growth of Sitka spruce trees, and ii) the growth of Sitka spruce trees is affected by the year of measurement. In addition to showing the overall response in each year x treatment, also ensure that the among tree variability is visible.

Explain your findings.

[20 marks]

Answer

Question 4

The Frog Dehydration Experiment on Three Substrate Types data can be accessed here: `AICcmodavg::dry.frog`.

a. Based on the dataset, what do you think was the purpose of this study? Provide a 200 word synopsis as your answer.
b. Create new columns in the dataframe showing:
   - the final mass;
   - the percent mass lost; and
   - the percent mass lost as a function of the initial mass of each frog.
c. Provide the R code that would have resulted in the data in the variables `cent_initial_mass` and `cent_Air`.
d. An analysis of the factors responsible for dehydration rates in frogs. In your analysis, consider the effects substrate type, initial mass, air temperature, and wind.
e. Provide a brief discussion of your findings.

[25 marks]

Answer

Question 5

Consider this script:

```
ggplot(points, aes(x = group, y = count)) +
  geom_boxplot(aes(colour = group), size = 1, outlier.colour = NA) +
  geom_point(position = position_jitter(width = 0.2), alpha = 0.3) +
  facet_grid(group ~ ., scales = "free") +
  labs(x = "", y = "Number of data points") +
  theme(legend.position = "none",
    strip.background = element_blank(),
    strip.text = element_blank())
```

a. Generate fictitious (random, normal) data that can be plotted using the code, above. Make sure to assemble these data into a dataframe suitable for plotting, complete with correct column titles.
b. Apply the script *exactly as stated* to the data to demonstate your understanding of the code and convince the examiner of your understanding of the correct data structure.

[10 marks]

Answer

Question 6

For this assessment, you will analyse the built-in R dataset `datasets::UKDriverDeaths`, which contains monthly totals of car drivers killed or seriously injured in road accidents in Great Britain from January 1969 to December 1984. This time series data allows for examination of long-term trends, seasonal patterns, and potential correlations with societal factors.

a. **Data Exploration and Preparation**

   i. Load the `UKDriverDeaths` dataset and examine its structure. Convert the time series data into a standard data frame format with separate columns for:

      - Year
      - Month (both as a number and as a factor with proper names)
      - Number of deaths/injuries

   ii. Create a new variable that classifies each month into seasons (Winter: Dec-Feb, Spring: Mar-May, Summer: Jun-Aug, Autumn: Sep-Nov).
   iii. Create another variable identifying whether each observation falls during a major energy crisis period (e.g., the oil crises of 1973-1974 and 1979-1980).
   iv. Identify and handle any potential inconsistencies or issues in the dataset that might affect subsequent analyses.

[20 marks]

Answer

b. **Temporal Trend Analysis**

   i. Create a comprehensive visualisation showing the full time series with:

      - Clear temporal trends
      - A smoothed trend line
      - Vertical lines or shading indicating major UK policy changes related to road safety (e.g., 1983 seat belt law)
      - Annotations for key events

   ii. Develop a visualisation examining monthly fatality averages across the entire period, ordered appropriately to show seasonal patterns.
   iii. Create a visualisation that compares annual patterns between the first half of the dataset (1969-1976) and the second half (1977-1984).
   iv. Using *tidy* data manipulation techniques, calculate and visualise the year-over-year percent change in fatalities for each month throughout the dataset.

[20 marks]

Answer

c. **Pattern Analysis and Decomposition**

   i. Calculate and visualise the average number of fatalities by season across all years.
   ii. Create a heatmap showing fatalities by month and year, with appropriate color scaling to highlight temporal clusters.
   iii. Implement a decomposition of the time series to separate: - The overall trend - Seasonal patterns - Remaining variation
   iv. Visualise each component and discuss what factors might contribute to the patterns observed.

Note: Some of this will be new to you. But don't worry, use any means available to you to solve the problem.

[25 marks]

Answer

d. **Data manipulation**

Starting with the data as presented in the `UKDriverDeaths` dataset, create a new dataframe identical to the `Seatbelts` dataset.

[5 marks]

Answer

TOTAL MARKS: 160

– THE END –