Individual Assessment 2

Changelog

Date	Changes
19 June 2024	Initial version
26 June 2024	Change due date
3 July 2024	Add that dashboard layouts may be used but not interactive Shiny components

Data for this individual assessment (and assessment 3)

You can read more about PhysioNet here. It is most famous for the MIMIC series of clinical data sets which have been used extremely widely, especially in health machine learning research. The full range of PhysioNet data sets can be found here.

Both assessment 2 and assessment 3 are based on the same clinical dataset provided by PhysioNet. PhysioNet provides access to three classes of datasets: open datasets, restricted datasets and credentialled datasets. Open datasets are immediately available to everyone. Restricted datasets require you to first register an account with PhysioNet (free, only name and email address required), and then agree to a data usage agreement (a single click) to get access to each dataset. Credentialed datasets require each user to undertakes reasonably extensive, US-centric training and pass an online exam before providing access.

The data set to be used for Individual assessment 2 & 3 is a restricted dataset, so you will need to create a PhysioNet account and agree to the terms of use for the data set. This will only take a few minutes.

The data set to be used is:

- Hospitalized patients with heart failure: integrating electronic healthcare records and external outcome data

Here is the abstract from the paper describing this data set:

> Heart failure is a common reason for hospitalization in the elderly and it is associated with significant mortality and morbidity. To facilitate epidemiological studies of heart failure, there is a need for high quality datasets to be made available to researchers. While several heart failure datasets have been established in Western countries, there is a paucity of data available from China. Understanding differences in patient populations and healthcare systems between China and other countries is important in providing optimal care. To help address this issue, we created a retrospective heart failure dataset using electronic health data collected from patients who were admitted to a hospital in Sichuan, China between 2016 and 2019. The dataset includes 168 variables for 2,008 patients with heart failure.

From this, it is clear that this is a serious, real-life clinical dataset containing considerable detail about a set of heart failure patients – it is in no way a "toy" data set. But don't panic! You will not be required to examine all 168 variable in the entire dataset, nor will you need to become a cardiologist. However, some understanding of the clinical condition of *heart failure* (which is quite different from "*heart attack*", the common name for a myocardial infarction, or from cardiac arrhythmias (disorders of the heartbeat) is useful. This resource published by the US National Institutes for Health is a useful starting point, as is this resource from the Australian Heart Foundation. The wikipedia page on heart failure is also very good and goes into a lot more technical and clinical detail.

Overview

We will refer to the data set as the *Zigong heart failure data set* henceforth, or just the *Zigong data* for short.

As noted on the *Assessments Overview* page, Individual Assessment 2 will involve visualisation of some aspects of the Zigong data, to be presented in the form of a short report or a brief presentation slide deck, compiled using the literate programming tools covered in the first few chapters of the course. You can use R Markdown and `knitr` for this, or you can use the newer `Quarto` framework, which works very similarly.

Note that because this is the first time this assessment has used the Zigong data in the HDAT9800 course, some minor clarification of the steps and aspects detailed below may be required in response to student feedback. Please do not hesitate to seek clarification, via the course Teams space. All changes will be listed in the Change log at the top of this page and such changes will be announced on the Teams space.

Due date

This assessment must be submitted by beginning of Week 8, 8:00am Monday 15 July 2024 .

Please see the *Late assessments* and *Special Consideration* sections on the course web site home page for policy on those issues.

Policy on the use of AI assistants in HDAT9800 assessments

Please see the *Generative AI, Large Languages Models and Assessments* page for details.

Setting up the assignment repository via GitHub Classroom

We will be using GitHub Classroom for this and all the other HDAT9800 assessment tasks (and for the optional unmarked exercises).

First, make sure you have installed `git` on your computer and that you have a `GitHub` account, as explained in the pre-recorded lectures and tutorials on `git` (see above), and that `git` is configured in `RStudio` (see also the *Setting up git & GitHub* page).

Then you can click on this link to the Individual Assessment 2 GitHub Classroom repository:
https://classroom.github.com/a/kZZmenoe

You may need to log into to GitHub first. You may be asked to choose your name from the roster (list) of students in the course (if you have done some of the optional exercises already, it may not ask you again to identify yourself). Be careful to choose your name, not someone else's. You will then be asked if you wish to accept the assessment. A new GitHub repository for the assessment will be created for you, just for this assessment. Only you and the course instructors can access it, it is not public.

You should then clone this assessment repository to your local computer, using RStudio (or other git interfaces such as GitHub Desktop if you prefer). Complete the assessment according to the instructions given below on this page, save and commit your work to your local git repository, and then push that work back to the remote `GitHub` repository for the exercise (this should all be automatically set up when you clone the repository in RStudio).

Be sure that you can successfully knit the `Rmd` (R Markdown) (or render the `qmd`) document that you will be working on in this assessment, before you commit and push your final version. The instructors will be marking your `Rmd` (or `qmd`) file by re-rendering it (re-knitting it).

You may revise your assessment task(s) as many times as you wish. We will mark the last committed version that you push to the GitHub repository for your assessment by the due date.

Task details

As noted above, the Zigong heart failure data set is a reasonably detailed collection of real clinical data on over 2,000 heart failure patients.

The broad task in this assessment is to create a report or a presentation slide deck which presents a brief introduction to this data set, in the form of an brief EDA (exploratory data analysis), and then five slides containing charts which use the concept of small multiples (facetting in `ggplot2` terminology) to illustrate significant (not necessarily statistically significant) or interesting differences in various

characteristics between defined subsets of the dataset. These visualisations should be presented as an HTML report or an HTML slide deck for a presentation, using the R Markdown and knitr (or Quarto) tools used in Assessment 1. You may use a dashboard arrangement for the report if you wish, but please do not use Shiny components in this assessment – they will be used in assessment 3.

At this point this task description may sound rather vague. Don't worry, more details are provided below, together with marking rubrics for each part of the assessment. However, unlike Individual Assessment 1, you have a great deal more freedom to design your response to the assessment tasks, and thus detailed instructions and prescriptive (detailed) marking rubrics are not provided. However, you are strongly encouraged to discuss your assessment response with your fellow students (in person or via Teams), and/or approach the course instructors for advice or tips on how to proceed (via Teams). The code for your assessment should be your own work (and/or any AI coding assistance declared), but you are free to discuss, workshop, brainstorm and share conceptual details with your fellow students.

This assignment is worth 23 marks (out of 100 marks for the overall course).

You will first enrol in the assignment via GitHub Classrooms at the link above. That will create a git repository for your assignment on GitHub, You should then clone that repository to your laptop (or local computer), then open the cloned project (the `.Rproj` file) in RStudio so that you can add your assignment code and other documentation to it. You should periodically, and regularly commit your code to git, and then push those commits to GitHub, even if the assignment is incomplete — in other words, save and push your work-in-progress, as often as you wish.

All R code you include should contain brief comments to indicate what it is doing, or if you prefer, you can echo the code in the document and include text explaining what the code does in the body of the document, or slide deck, or both. We want to see evidence of your thinking process in the code.

Task steps and aspects

1. Initial steps – **note**: you may re-use code you wrote for Individual Assessment 1 for these steps (or improved versions of it if you wish) [4 marks (out of 23) overall for all of the following initial steps].

a. The R Markdown document for the assignment should be named "assignment_2.Rmd" (or "assignment_2.qmd" if you wish to use Quarto, please see below). This file should be in the root of the project directory, not in a subdirectory.
b. The YAML information at the top of the document should specify the title of the document (choose an appropriate title, the exact wording does not matter), your name as the author of the document, and the approximate date you completed the assignment.
c. The YAML information should also explicitly specify that the output format is an HTML document or as an HTML presentation (slide deck) format. You may have to consult the R Markdown (and/or Quarto) documentation on the web to determine how to do this (it is well documented).
d. You should include a set-up code chunk at the top of your document (after the YAML header) to load the R packages required in the document and to set defaults for your document.
e. Download the main Zigong data set (called `dat.csv`) and place it in a directory (folder) called *zigong* at the same level as your assessment 2 project directory, so that you can read the data into R using this code: `zigong <- readr::read_csv("../zigong/dat.csv")` The reason for doing this is we do not want you to commit the actual data files into your assignment `git` repository, so they need to be stored outside the project directory, but we also need the data file(s) to be located in a standard place so that the markers can easily re-run your code. You may add options to the `read_csv()` function, but **please** leave the path to the data file exactly as it appears above, and place your local copy of the data file accordingly. Ask your fellow students or instructors for help if required.[Marks will be deducted if the data is not read directly from the directory path given above or if the data file is committed to the assignment repository.]
f. A separate code chunk should then be included to clean and transform that data frame (tibble) into a form that is easier to work with. This should include, if necessary, a) renaming columns so that the names do not include any spaces or dots — use the `clean_names()` function in the `janitor` package to do this efficiently; b) converting categorical columns to factors, with appropriate labels for each factor value. The text in the categorical columns is fine to use as the value labels, you do not need to transform it further. There are not that many categorical columns. However, there is an age group column which does need better labels for each value. You can combine some of these steps with step 1.e. if you wish – that is, you might wish to set column data types at the time you read in the data (we strongly suggest that your use the `read_csv()` function from the `readr` package to do that).

2. Create a set of no more than 10 slides containing charts which visualise the range and/or distribution of values in no more than 10 variables in the dataset, for the dataset overall. These variables should include age (or age group), gender, height and weight or BMI, and type of heart failure. You may choose which other variables to visualise, but at least three of them should be the numeric variables (columns). Give each chart a title, and axis labels and value legends as appropriate. You may wish to use a `ggplot2` theme to make the charts look more attractive and/or readable. [8 marks]

3. Use “small multiples” (using `ggplot2` facetting or the `patchwork` library of similar means) to display some interesting or significant differences between subsets of the overall patient population in the data set (in statistical parlance, *contrasts*). These contrasts should be created on the basis of at least one of the categorical (or logical) variables in the dataset. That is, you need to *condition* each chart in a set of small multiples on the value of one or more other variables in the dataset. Good choices for the conditioning variable(s) are the categorical variables. The visualisations needs to display differences (or similarities) clearly, and thus including too many subset groups and too many response (continuous) variables is likely to detract from the visualisation (and thus lose marks). You do not need to provide a comprehensive or exhaustive analysis or visualisation of every aspect or contrast in the dataset. You should include no more than five (5) slides in your slide deck containing such visualisations, but you may add slides in between them explaining each one or commenting on what they show. Include some brief explanation of the construction of your visualisation (what columns are used, what type of chart is employed, what data transformations were needed and how they were done). There is no need provide clinical explanations of why they may show what they do (in other words, you do not need to explain the medical or biological reasons behind the responses shown). Include a brief discussion of potential conceptual problems or implementation issues with your chart(s). [11 marks]

Assessment responses which fulfil the brief as described above will be awarded at least 60% of the available marks for each task. Beyond this, additional marks will be awarded at the discretion of the markers for excellence in implementation of your solution (judged by the markers against the benchmark of your fellow students), including neatness and documentation of your code, the effectiveness and quality of your visualisations and the completeness of your explanatory text.

Please ask for clarification of any aspect of this assignment. Responses will be recorded on this page in the Question and Answer section below, and/or as additions or changes to the task descriptions above.

Please do not hesitate to ask for help or suggestions from the course instructors about how to approach any of the tasks in this assessment.

Questions and Answers

> Q: I’m really struggling to come up with a good visualisation!

> A: Don’t panic, this is as much of a formative exercise (a learning experience) as it is a summative exercise (a test of your skills), and thus most of the marks to be awarded are **not** based on how effective your visualsiation is, but rather on how well you have documented your efforts to visualise the data in the manner requested. It isn’t easy! Nor are there right or wrong answers. Obviously some marks well be allocated to how effective your visualsiations are at showing some aspect of the data, but the majority of the marks will be awarded for showing how you attempted to visualise the data as requested. Failure to create a perfectly clear or insightful visualisation will not result in a low mark. Marks are awarded for showing your thought processes and the journey you went on.