Group Dashboard Project Feedback - Jonny, Louise, Malcolm, Pui, Ricardo

**Presentation**

A well structured and delivered presentation. I was left with a much better understanding of the NHS demands in summer vs winter. Demo of the app was good - I’d maybe begin on the landing page of the app, rather than a separate tab. It was clear you all contributed to the project and having each person discuss their own main area of inquiry/ part in the project was a good way to structure the presentation. Having the same person introduce and conclude the presentation was a good idea as well. I would maybe have liked to see more of linking the dashboard to specific questions you were answering. I believe you introduced the questions and spoke generally about how areas of the dashboard answered them but a tad more specificity might have clarified my understanding. Overall, you did an excellent job of presenting statistical information and answering the questions we put to you.

**Positives**

Overall

* File structure was exactly what I’d expect from an industry app: ui, server, global, data, and a www folder.
* App itself had a nice layout, intuitive UI and gave the user the ability to drill down into particular data as well as view summary stats at a glance.
* Good consistent layout, colours and themes. I really liked that you set up a global theme for all of your plots.
* Good exploration of temporal, spatial, and demographic data.
* Dashboard appearance is nice and clean with every plot having a clear purpose.
* Good use of global.R
* Good use of git; it looks like you were able to work well collaboratively: each of you making many and substantial contributions.

Overall: excellent dashboard

Data Preparation

* Simplifying the polygons: excellent.
* Cleaning was good and made good use of taught materials from module 1.

ui.R

* Really nice code structure. Good names, indentation, white space, and flow. It was easy to follow and understand the construction of your user interface. Your comments and sectioning were excellent also.
* I thought all of the input options were good and appropriate widgets for each.
* Splitting up the topics into tabs meant I never had to scroll down - excellent.
* Plots were labelled well, and context was provided where necessary.
* Good plot choices.

server.R

* Laid out clearly - section headings helped. And the sections were in the same order as in other scripts and files (it was easy to find sections).
* Validation was really nice.
* Great use of different kinds of plotting techniques: leaflet for spatial and ggplot for bar plots and line plots. Every visualisation felt appropriate.

**Potential Improvements**

Overall

* I thought the app was excellent. A quick extension might be to add some infoBoxes / cards to summarise KPIs, such as: total patients, total beds, total in summer, total in winter, total nhs staff working hours - that kind of thing. An element of dashboards is that users can discern important summary statistics quickly.
* In general, as this was exploring the strain on the NHS during winter, then staff numbers/NHS resources might have been an interesting source of inquiry (data permitting of course).
* Sourcing a plotting theme: cool. I’d put any scripts containing functions required for your app to run in a folder R/. And keep scripts/ for cleaning data and preparing data (like simplifying shapefiles).
* You can remove library calls you don’t end up using in your app. I know you did some statistical analysis (excellent) but don’t use any infer functions in the app, so can take that out.
* I’d combine all your simplified shapefiles into a separate folder within clean data for clarity.

Data Preparation

* Some repetition in the cleaning script (hb\_name = case\_when(...)) - I’d change this to a function add\_hb\_name(data) {} that you can call on the necessary dfs.
* case\_when() could use a TRUE ~ NA\_character (allows evaluation if there exists any values that don’t meet previous conditions.)
* Good structure, could have used a few more comments.
* Are all of the variables you save in your clean datasets necessary for exploration in the app? Perhaps a bit more time planning, reviewing what information you had access to in each dataset would make it easier to integrate into the dashboard in a standard way, rather than a specific way for each dataset. It is clear that a lot of planning already went into the app, data, and final output of the project.
* Is your clean data tidy? E.g. Could episodes and spells be converted to a longer format?

ui.R

* (super nitpick): I’d take a new line after every function call e.g
  + column(width = 3,
    - …

would become

* column(
  + width = 3,
  + …
* There were occasions where you exceeded the standard practice line length of 80 characters - these were, however, few.
* I think that defining the input choices within the input widget is fine, - especially when it’s a fiddly date/time. If you’re finding you’re writing a lot of these and repeating yourself, then there would maybe be a case for standardising your inputs - either with a function that creates them based on the data, or by setting some defaults.
* I think your layout was fine, and I’m very happy with the consistency. I’m not overly fond of having a mix of vertical and horizontal reading. The inputs read left to right, but then within each widget, they might read from top to bottom. An alternative ui option: use the collapsible sidebar for the inputs, and across the top of your app you could have the dashboard navigation. This might involve some dynamic UI though, since the input options are dependent on the tab. This would let you keep having the plots side by side - which I liked.
* I thought the plots were good, I might have opted for a dodged bar chart for A+E attendances each year.

server.R

* You can restrict the view of leaflet options to only scotland.
* For certain date inputs (a+e admissions), the scale on the x axis can have text labels for .5 of a year: 2015.5, you can clean this up before plotting, for the other date plots this was handled automatically by having it be a time data, rather than a double.
* Could have been commented a bit more.
* Some variables aren’t named uniquely or clearly. actionButton, actionBut, actionBut2 … I’d change these to filtered\_admissions, filtered\_length\_of\_stay etc.
* Treat times as times, rather than doubles. Instead of filtering for each date in a date range, filter for dates above the low value given, and below the high value given.
* Some plots are made the same way: could be changed to functions that are called.
* Instead of having two eventReactive calls for the same button, have one that: when the button is pushed filter the data generally and then create two datasets (one filtered for scotland, one for the healthboard) finally return these as a list.
* It might be more efficient to perform this kind of operation once in data preparation, rather than each time a button is pressed: mutate(winter\_flag = case\_when(month %in% c(1, 2, 3, 10, 11, 12) ~ "Winter", TRUE ~ "Summer"))
* Both leaflet plots are produced in the same way - instead of repeating the code, a function could be created, that you call for the left inputs to create the left map, and the right inputs for the right map. You could save this function in a separate file in R/
* If placeholder text isn’t dynamic (doesn’t depend on user input) does it need to be rendered in the server?