MSc/PGDip/PGCert Health Data Science
Assignment cover sheet

Swansea University
Prifysgol Abertawe
Medical School
Ysgol Feddygaeth

Module: **(Please tick the appropriate box).**	☐ PMIM102/J Scientific Computing in Healthcare ☐ PMIM202/J Health Data Modelling ☐ PMIM302 Introductory Analysis of Linked Health Data ☐ PMIM402/J Machine Learning in Healthcare ☐ PMIM502/J Health Data Visualisation ☒ PMIM602 Advanced Analysis of Linked Health Data
Module part (if applicable):	Reflective Journal
Title of assignment:	*Reflective Journal*
Student ID number:	2311233
Word count (if applicable):	n/a
Declaration:	I understand the following conditions which apply throughout this course: 1. I confirm that I am the sole author of this work. 2. I understand that proof reading by a third party is not permitted. 3. I understand the need for academic integrity and that all my submitted work will adhere to its principles. 4. I understand that the teaching team will take measures to deter, detect and report any academic misconduct. 5. I agree to my work being submitted to the TurnItIn academic database. 6. I understand the importance of assignment deadlines and the need to seek help in good time where personal circumstances interrupt my work.
Please copy and paste this declaration onto the front of the submission.	

Issue #1

1. **What was/were the problem(s) experienced and in what specific exercise(s)?**

   Problem Location**:** I found problems in day 1 training session 1, Step 10, and in day 3 training session 3, step 5 in the for-loop block, in the mutate function, where "!!sym" is used. Only the first location is considered for issue exploration as the idea could be transferred and applied to the second location.

   Context**:** I understand that training session 1 is about implementation and analysis of platform file suitable to perform research on a group of four internally and externally linked datasets. The 4 datasets represent the population of patients with diabetes mellitus. Files are loaded and processed to generate an ultimate file. To process the hospital records data, in step 10, the "!!sym" syntax is used.

2. **What was/were the reason(s) or contributing factor(s) for the problem(s) that arose?**

   There were 3 factors that contributed to the problem. They are:

   1. Insufficient syntax comments: Although, there are couple of lines of description associated with the syntax before the utilisation of "!!sym" function, it was insufficient for a person who is relatively new to the R syntax. It was the first time that I saw the syntax being used in the given setting. The first time I saw the "!!" operator I was confusing it with the "not" or symbolical "!" boolean operator.
   2. Uniqueness of the syntax: I enquired about the syntax to the course instructors. It was found that the syntax is not so commonly used and thus there was a need to do extensive research for the syntax.
   3. Nested nature of the syntax: The usage of "!!" and "sym" was used inside the short hand if-else function, nested in the mutate function joined using the pipe operator. The debugging such nested function is difficult using "browser" function in R. If it was possible, I could have put "browser" function in the syntax and checked line by line to understand the flow of data and logic. But due to the nested implementation it became difficult to use "browser" and perform step by step debugging.

3. **How did you investigate the problem(s) encountered and how did you solve the issue(s)?**

   Investigation of the problem encountered was carried out sequentially as follows:

   1. Internet research and documentation: In this step, I searched the issue on the internet and R documentation. I came across the concept of "diffused expressions" in rlang which I thoroughly examined and tested its execution with similar snippet examples. Initially I thought "!!sym" is a single function, but through documentation and tests I realized that the "!!" and the "sym" are two different syntax. I found out it only works together within the mutate function. It is a special feature of the mutate function to handle diffusion expressions.
   2. Implementation breakdown: Here, the implementation of the logic in the reference file is examined step by step. I broke down the implementation in 3 steps. First, where the vector of diabetes codes is defined. Second, where the diabetes indicator column in the data frame is created. Third, where

the for loop is executed to run over 1 to 21 diagnosis code columns on the HDMSdata2 file. I realised that the vector created in the first step is used to tag the indicator column in the HDMSdata2 file created in the second step. It is set as 1 if the diabetes code is found in the diagnosis code columns otherwise set as 0. This process helped me segregate the syntax, understand the big picture, and pin-point the problem.

4. **What conceptual and analytical understanding was gained from experiencing the problem(s)?**

   Conceptual understanding**:** During exploration, I gained the conceptual understanding on diffusion expressions in R syntax. I understood the correct meaning and usage of the syntax. My confusion on using single exclamatory "!" as boolean not operator versus the double exclamatory "!!" operator was cleared. I solidified my understanding on the correct usage of combination of the operator and symbolic function.

   Analytical understanding: During investigation, I learnt how to critically analyse and identify the correct approach to debug nested functions. I understood how to identify reliable sources of documentation for referencing the syntax. I broke down the implementation into smaller parts and experimented with smaller sections of syntax to understand the input and output of the syntax logic. This helped me to connect the dots. I learnt a clever trick to dynamically access the data frame columns.

5. **How could your learning(s) from this experience be applied to prevent the problem(s) occurring in future research work?**

   I learnt that,

   1. The implementation of "!!sym" is a clever way to access the columns in the data frame when implementing for-loops to access the columns dynamically. However, the implementation leads to loss in clarity and readability of the syntax. As diffused expressions are not that widely and commonly used, it is new knowledge for the developers to explore and invest time to get familiar with it. This clever way of accessing the columns and processing the columns might come in handy for future experiments at personal level.

   2. It is important to understand the context of the syntax implementation. Looking at the syntax and around the syntax of interest helps in deciphering nested functions. Experimenting with smaller chunks of code helps in debugging.

   3. I should not assume the future user or reviewer of the syntax would be aware and understand all the parts of the syntax with just a short description. The individual who is using my syntax could invest valuable time to get familiar with it. To make it as seamless and smoother process as possible, for the analyst a detailed documentation along with references to the syntax should be provided. I realized the importance of thorough and detailed documentation.

Issue #2

1. **What was/were the problem(s) experienced and in what specific exercise(s)?**

   Problem Location**:** I found the problem in the training session 3 on day 3, on the step 17.

   Context**:** The reference code file used here is the R + tidyverse html file - aalhd_day3.html. The premise of the training session 3 is to implement methods of analysis of linked data files that require random sampling procedures for complex longitudinal designs. An algorithm is devised to correctly classify the women into disease stages 1, 2 and 3 using the variables: timepreg, stgpt1, stgpt2 and stgpt3.

2. **What was/were the reason(s) or contributing factor(s) for the problem(s) that arose?**

   1. The major factor contributing to the problem was mismatch of reported outputs. When the numbers reported at step 17 between the workbook and the reference syntax file are compared, they do not match. See the below table.

Stages	Workbook	Ref code
stagpreg [stagpreg = 1]	417	405
stagpreg [stagpreg = 2]	85	93
stagpreg [stagpreg = 3]	87	93

   2. The secondary factor was vastness of the code. The context of the syntax at each steps was too long to follow and keep up with.

3. **How did you investigate the problem(s) encountered and how did you solve the issue(s)?**

   The investigation was carried out by examining the steps 1-17. Initially, it was a daunting task to figure out which part of the code is causing the problem. So, sampling was performed to reduce the time of investigation. Out of 18 steps to check, the last 3 previous step, the first 3 previous steps and the middle 3 steps were selected for examination. During each step, the input and the output value were checked between the sources. I had to pay attention to the minute detail in each part of the syntax. It was found that at step 17, there was a discrepancy in the logic for partition on the “timepreg” variable. The logic was changed. The algorithm was modified from the old logic

```
rpfdiab5 <- rpfdiab4 %>%
    mutate(stagpreg=ifelse(timepreg<=stgpt1, 1, 0)) %>%
    mutate(stagpreg=ifelse(timepreg>=stgpt1 & timepreg<=(stgpt1+stgpt2), 2, stagpreg)) %>%
    mutate(stagpreg=ifelse(timepreg>=stgpt1+stgpt2, 3, stagpreg))
```

   to the new logic.

```
rpfdiab5 <- rpfdiab4 %>%
  mutate(stagpreg=ifelse(timepreg<=stgpt1, 1, 0)) %>%
  mutate(stagpreg=ifelse(timepreg>stgpt1 & timepreg<=(stgpt1+stgpt2),
                         2, stagpreg)) %>%
  mutate(stagpreg=ifelse(timepreg>(stgpt1+stgpt2), 3, stagpreg))
```

Step by step close examination of the syntax helped me figure out the problem.

4. **What conceptual and analytical understanding was gained from experiencing the problem(s)?**

   Conceptual understanding: I learnt the idea of stage partition and its relevance. The error in the logic was rectified through empirical approach and critical analysis. References and hints from the workbook aided with adding clarity to the concept.

   Analytical understanding: The empirical analysis and sample selection of steps approach helped me to triangulate the error efficiently. 50% of the time was saved. The analytical investigation encouraged me towards exploration of the syntax beyond the instructions. It is quite possible for differences to occur in the sources and mistakes could be made. This gave me a lesson to cross check the sources as well.

5. **How could your learning(s) from this experience be applied to prevent the problem(s) occurring in future research work?**

   1. Importance of documentation: It is important to document minute details. Though there was hint of the devised algorithm, a complete implementation of the expected algorithm would have been better for absolute comparison. If there was no documentation for reference for the given problem, it would have been extremely difficult to triangulate the source of error.
   2. Importance of double checks: I realize that it is utmost important to double check the syntax implementation. It is very common for logical typographical errors to occur. It is important to cross check source materials to avoid any potentially minor or major errors.