# 

# COMP 5070 Statistical Programming for Data Science

## Take-home Exam

* This take-home exam is worth 25% of your overall grade. Doublecheck the course webpage for the deadline. This is an exam – if you are late with the submission, then LearnOnline system will be closed, and you will get zero mark for the exam.
* The exam is to be submitted as uncompressed files using LearnOnline (no zip, tar, rar, etc). Your submission should include (see more details at the end of the instructions):

1. well-presented report in Word or PDF format
2. R-scripts used to download data, produce presented results/graphs, run dashboard app
3. data set you scrape from the internet.

* The exam is out of 100 marks. There are three main parts in the exam:

1. downloading data using *rvest* package (25%)
2. run data analysis and prepare a written report (50%)
3. create an interactive dashboard using package *shiny* (25%).

* Below are possible penalties on the top of the total mark:

1. Upto 10 marks for none-optimised code. It is OK to use for-loop for scraping data as internet download is slow, so for-loop inefficiency is not important. In most other situations it is (probably!) a bad idea to use for-loop. Use vectorisation as much as possible.
2. Upto 10 marks for not clear coding style. Code clarity is an important criterion for your submission. Thus, you should choose meaningful variable names and adopt the use of comments – you don't need to comment every single line, as this will affect readability – however you should aim to comment at least each section of code.
3. Upto 10 marks if the code does not run successfully. I should be able to run your R-code without any modifications, that is, I open your code in RStudio and press Ctrl+Shift+S. If there are any errors – you lose points.
4. Upto 10 marks for poor presentation of the written report.

* This is an exam, so there is no provision for a late submission. Please don’t send me your submission by email – I am not allowed to accept them. All submission should go through LearnOnline.
* Plagiarism is a specific form of academic misconduct. Although the University encourages discussing work with others and the Social Forum will support this, ultimately this submission is to represent your individual work. If plagiarism is found, all parties will be penalised. You should retain copies of all assignment computer files used during development. These files must remain unchanged after submission, for the purpose of checking if required.
* **Read very carefully hints and tips in the Appendix.**

# Job market for Data Scientist in Australia

In this assessment you will investigate job opportunities for Data Scientist in Australia based on the data from the most popular job search website “Seek.com.au”.

## Downloading data (25 points)

You must get your own data for this exam. You will use *rvest* package to scrape data from [https:// www.seek.com.au/](https://www.rent.com.au/). As you are focused on Data Scientist jobs only, then your starting point will be <https://www.seek.com.au/data-scientist-jobs>.

You don’t want to put too much pressure on the “Seek.com.au” website. Please, review all questions, make a plan of what data you need, then create one R-script to download all required data, clean the data and save the data as a dataframe on the hard-drive using functions save() or saveRDS(). Later you will run analysis and create a dashboard that will load your data file by functions load() or readRDS().

Test your code on a small piece of data. If everything works fine, then download full data just once, store it and then use stored data for the analysis. Try to avoid downloading the same data multiple times.

## Data analysis (50 points)

**Question 0**. **Introduction** to the business case, data available and analysis to be presented later.

It might be beneficial if you write an introduction *after* you completed all other tasks. This way you would have a good understanding of what to put in this section as you already know what you’ve done and what data you’ve used.

**Question 00**. **Conclusion** at the end of the report briefly summarising all your findings. You have multiple questions to investigate, so you will have multiple topics to summarise in the conclusion.

**Question 1** “**The state to be”.** Study the number of positions with respect to states. Then study the distribution of salaries overall and then salaries with respect to the State. What are relationships between states, number of positions and salaries?

**Question 2. “What's in a name?”** Study the popularity of different job titles and then different industries (top level classification). Compare salaries in different industries.

**Question 3. “Seize the day, then let it go.”** Study the number of positions advertised changes over time. Then analyse the relationship between day of the week and the number of positions.   
**Hint:** For this question to be processed correctly, I advise to download the data in the afternoon. I do suspect that Seek.com.au don’t use Adelaide time or even Australian time.

## Interactive dashboard (25 points)

You will create a dashboard using package *shiny* as an alternative delivery to the written report. The dashboard should cover the same topics as a report. Take care about a good design for the dashboard. One graph per screen is (probably) a bad idea (unless your graph is very detailed, and you need a lot of space for it). All graphs on one page could be a bad idea too (unless your graphs are very simple).

All graphs on the dashboard should be interactive. The dashboard is not a simple copy of the written report but an enhanced and interactive version of it. Numerical summary (e.g., table) is a data visualisation too for the purpose of a dashboard. Fell free to include them and try to make them interactive.

## Submission structure

Here is a suggested structure for the exam submission. While it is not compulsory to follow this structure, it should help you to better organise your exam, avoid missing any relevant files, and speed-up marking process. These are five files suggested for your submission:

1. R-script file *download.R* with all code used to scrape data from the internet, clean it and store on the hard drive ready for the analysis.
2. R data file – *examdata.RData* or *examdata.rds* – where you store all information prepared by *download.R* script. If you have multiple variables to store, then you can use function *save()* or you can put all your variables in one list and use function *saveRDS()*.
3. R-script *analysis.R* with all code used to load and analyse the data file and to output all statistics and data visualisations for the written report.
4. MSWord or PDF file with a written *report*. It should be well-organised and well-prepared report with introduction, conclusion, and all required discussions. Good report should have a title, headings and sub-heading, page numbers, titles for figures and tables.   
   Report should NOT include R-code or R-outputs but results of your analysis. I can see your code in the submitted R-script.   
   For the report you should always aim to have three elements: numerical summary, graphical summary and discussion combining together and explaining the first two.
5. R-script *app.R* with the code of your interactive dashboard. The dashboard will include mostly the same visualisations as your report, as the dashboard is an enhanced copy of the report, but without discussions. However, data visualisations should be interactive, e.g., add/remove/adjust some parameters – state, salary level, industry, time. Like the report, the dashboard should have appropriate titles and headings – it should be easy for the reader to understand what is what on the dashboard.

## Getting help

You are allowed to ask questions during the exam. I will be monitoring the forum and try to respond as quick as possible. At the same time, I will not answer direct questions like “how to do question 1?” or “I’ve done this – is it correct?”. When you ask questions – remember about academic integrity and don’t reveal a solution or a part of it.

## Appendix

Web scraping for Seek.com.au is not difficult but a little bit tricky. If you try to extract data by using one class only, it might crash RStudio. Seek.com.au use a bit “strange” system of hierarchical classes, so one class can be used for a huge number of objects. As a result, RStudio might not be able to handle so many objects and crashes. It is possible to get data by carefully expanding hierarchy of classes, however you know that you can use different elements to extract data from HTML, not just a class.

Package *rvest* allows to process multiple elements of the similar type at the same time. However, I would advise to focus on one position at a time as they might have slightly different structure. For example, not so many positions have salary advertised.

**Important:** you must download your own data for the exam. **However**, if you fail to get/prepare the data, you can use the data I downloaded and prepared for you. You will lose 25 points for not doing web scraping, but you will have a chance to earn points for the analysis and the dashboard.

The data file has a date as a part of its name. This is the date when I downloaded these data. You will use it in the question 3 for time-based analysis.

My data file has too much information as I downloaded all information available. It is not expected that you use all these variables or download the same information for your own data. Use only what you need for the analysis.