# 

# 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 so 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 part of 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.

# Rental property market in Adelaide

In this assessment you will investigate Adelaide City rental property market.

## Downloading data

You must get your own data for this exam. You will use *rvest* package to scrape data from <https://www.rent.com.au/>. As you are focused on Adelaide CBD and surrounding suburbs, then your starting point will be <https://www.rent.com.au/properties/adelaide-sa-5000/> .

You don’t want to put too much pressure on the “Rent.com.au” website. Please, review all questions, make a plan of what data you need, then create one R-script to download and clean all required 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().

Download data just once, store it and then use stored data for the analysis. Don’t download the same data multiple times.

## Data analysis

**Question 0**. **Introduction** in 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** **Rental prices.** Study the distribution of rental prices overall and then rental prices with respect to the type of the property and to the number of bedrooms. What are relationships between prices and type of the property, and between prices and the number of bedrooms?

**Question 2. Car spaces.** Study the number of car spaces overall and then with respect to the type of the property and to the number of bedrooms.

**Question 3. Location, location, location.** Study the popularity of different streets withing Adelaide and different surrounding suburbs (ignore street names outside Adelaide – take just suburb names). Describe the most prominent streets/suburbs and their features, e.g., average price, most common property type, etc. No need to discuss all locations and all features. Focus on important or interesting details.

## Interactive dashboard

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. Your dashboard is not a simple copy of the written report but an enhanced and interactive version of it.

## 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.
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 a discussion. However, data visualisations should be interactive, e.g., add/remove/change some parameters – number of bedrooms or car spaces. Table is a type of data visualisation too, so it can be included in the dashboard. 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 – If there are problems with internet access

Here is something you don’t need but it is good to have a backup plan.

If you get error messages while trying to scrape data from <https://www.rent.com.au/> ; or if you don’t have access to this website (e.g., overseas students or poor internet connection) – you can use an archive *rent\_html.zip*. It contains html pages you need for the exam.

You use *rvest*, and all code for the web scraping is the same – you just use a different address to load data

# normal way – with internet access

read\_html(″https://www.rent.com.au/properties/adelaide-sa-5000/p1″)

# offline - no internet access, html files are in working directory

read\_html(″https\_\_\_www.rent.com.au\_properties\_adelaide-sa-5000\_p1.html″)

As you can see, it is the same address but special symbols “/” and “:” are replaced by underscore, as they cannot be a part of the file name, and an extension “.html” is added.

Offline html page might take too long to open them in the internet browser as they would try to connect to internet. Just stop the browser and you see the page. Offline html pages look different (really messy) in the internet browser comparing to the original web design as they don’t have access to server-based CSS. However, offline pages are identical to the pages on the web server in terms of the content. The same web scraping code will work for both – internet-based and offline pages.

There are seven files provided as there were 132 properties listed on the day when I downloaded the data. It will be different for you. You should not try to cheat and get a list of files in the working directory. You should create file names manually and honestly “download” them.

Most probably, you don’t need this archive and this section. But if you have problems with internet – they should help.