STAT0030 Assessment 2 — Instructions

For this assessment you should submit online – on the course Moodle page using the link "ICA2: Click here to submit your assignment". Make sure none of the files contains your surname, as the marking must be anonymous. You must submit two files:

- An electronic copy of your `StudentNumber.rmd` file, containing your R markdown code. For example, if your student number is 18239004, your R markdown script should be saved in the file `18239004.rmd`.
- A single PDF file named `StudentNumber.pdf` containing the knitted output of the Rmarkdown file. This should correspond **exactly** to what is produced when knitting the submitted `.rmd` file.

Any output within your pdf should be clearly presented and structured according to the question parts. Your report (including the graphics but excluding the hidden code) should not exceed 5 pages.

STAT0030 Assessment 2 – Marking guidelines

The assessment is marked out of 40. The marks are **roughly** subdivided into the following components.

1. Exploratory analysis (10 marks): investigation and commentary of initial statistical properties, relationships, and anything of note which helps justify your choice of graphs and modelling strategy.
2. Graphical presentation (5 marks): appropriate choice of graphs and formatting.
3. Modelling strategy (10 marks): marks here will be based on a structured, justified, well-principled approach with clear and concise discussion.
4. Interpretation of final model (5 marks): comparison of the two final models and commentary on their quality.
5. Quality of the code (10 marks): your code should be clean, readable (with sufficient commenting for the user) and efficient.

STAT0030 Assessment 2 — Questions

The file `grocery.csv` contains a sample of sales and promotional information of 12 products from a set of grocery stores over a subset of a period of 156 weeks, beginning January 2009 through December 2011. All 12 products belong to the frozen pizza category. The file includes the following variables:

VARIABLE NAME	DESCRIPTION
BASE_PRICE	regular price of item
PRICE	actual amount charged for the product at shelf
WEEK_END_DATE	week ending date
STORE_NUM	store number
UPC	(Universal Product Code) product specific identifier
MANUFACTURER	manufacturer
DISPLAY	product was a part of in-store promotional display
FEATURE	product was in in-store leaflet
TPR_ONLY	temporary price reduction only (i.e., product was reduced in price but not on display or in an advertisment)
UNITS	units sold

The response variable here is UNITS, i.e., how many units of that product were sold in a particular week at a particular store. The variables PRICE, BASE_PRICE and WEEK_END_DATE are all numeric. STORE_NUM, UPC and MANUFACTURER are all categorical. DISPLAY, FEATURE and TPR_ONLY are all boolean (i.e., they are 1 if the condition is TRUE and 0 otherwise). Make sure R knows what type to assume for each variable.

Your overall goal is to build a (generalised) linear model and an advanced regression model to predict UNITS given the available covariates, and to compare whether one is significantly better at predicting the response than the other. Detailed instructions are as follows:

1. Download the file `grocery.csv` from the STAT0030 Moodle page. Read the data into R using `read.csv` with the argument `header=TRUE`.

2. Obtain summary statistics and make useful plots of the data — i.e., that are relevant to the objectives of the study. Such plots might include, but are not necessarily restricted to, pairwise scatter plots for quantitative variables with different plotting symbols or colours for each product, or boxplots for categorical variables. Put plots together in a single figure where appropriate and consider the possibility of using log scales for the quantitative variables.

3. Find a linear model or generalised linear model (i.e., one using `lm()` or `glm()`, refer to Lab 6) that enables UNITS to be predicted from the other variables and that is not more complicated than necessary. You may wish to consider using log transformations of one or more of the explanatory variables or of the response variable and to consider interactions between variables. You should consider a wide enough range of models to

make your choice of model convincing and use appropriate diagnostics to assess them. But ultimately you are required to recommend a single `lm()` or `glm()` model that is suitable for use (in the grocery retail industry, for example) and to justify your recommendation.

4. Find an advanced regression model (for example, using gradient boosting or random forests, see Lab 7) to predict `UNITS` using the available covariates. As above, you are encouraged to consider a variety of models, but ultimately you are required to recommend a single model from this family.

5. Perform 10-fold cross validation to compute the cross-validated Root Mean Square Error (RMSE) of each of your two models in parts 3 and 4. For each of your folds, compute the RMSE of your models from parts 3 and 4, so that you obtain 10 pairs of RMSE values. Perform a paired t-test to assess whether one of your two models is significantly better than the other in terms of predictive power.

6. Write a brief report on your analysis in four sections:

   (a) Describe briefly what you found in your exploratory analysis in part 2.

   (b) Describe briefly (without too many technical details) what models you considered in parts 3 and 4 and why you chose the models you did. Provide your own brief interpretation of these two models in the context of the application.

   (c) Using your results from part 5, briefly discuss the advantages and disadvantages of each of the two models and select a single "best" model, clearly explaining your reasoning.

   (d) State your final model clearly. Use your model to qualitatively describe how product sales depend on all the different covariates. Also give an estimate of what the average effect of decreasing `PRICE` by 10% (but keeping all other covariate values constant) would be on the product with `UPC = 7192100337`, during `WEEK_END_DATE = 39995` at `STORE_NUM = 8263`.

Your `.rmd` file should include all your code but you should use the option `echo = FALSE` so that your code does not appear in the knitted report. You do not need to include all your output and graphics. Instead, include whatever details and output you think are important to your model building and conclusions. You can control whether any output from a code chunk is included or excluded from the knitted report using `eval = TRUE` and `eval = FALSE` in the R chunk options. Your report (including the graphics but excluding the hidden code) should not exceed 5 pages. Your report should be at a level that can be understood easily by somebody with an MSc in Statistics.

STAT0030 Assessment 2 — General hints

1. In general, there is not a single 'right' answer to each question. To obtain a good mark you should approach the questions sensibly and justify what you're doing. Credit will be given for code that is clear and readable, while code that is inadequately commented will be penalised. You might like to use scripts `cosapprox.r` (Lab 1) and `tablet.r` (Lab 3) as models.

2. The assessment is designed to test your ability to use the computer to learn about a real data set. This will be assessed not only on your computing skills, but also on your ability to carry out a sensible and informed statistical analysis: material from your other courses will be relevant here. To earn high marks for this question, you need to take a structured and critical approach to the analysis and to demonstrate appropriate judgement in your choice of material to present.

3. Marks will be deducted if your `.pdf` file does not correspond *exactly* to the results we obtain when we knit the `.rmd`. You should assume that the input file is available at the same location as your `.rmd` file.

4. More credit will usually be given for code that is more generally applicable, rather than tailored to a particular situation or set of data. For example, if you were asked to print out the mean age of a group of people, you could do either of the following:

   - Calculate the mean before you write your final script, and then insert a line

     ```
     cat("Mean age is 25.3\n")
     ```

     (or whatever the mean happens to be) into your script.
   - In your script, create an object (say `xbar`) that holds the mean age, and then insert the line

     ```
     cat(paste("Mean age is",xbar,"\n"))
     ```

     into your script.

   The second approach is clearly more general and will earn more credit, since it will work for other similar data also.

5. All graphs should be clearly and appropriately labelled (giving units of quantitative variables), titled and formatted. By 'appropriately formatted' we mean, for example, that axis scales should be well chosen.

6. Your program should be **well commented**. If you have defined functions, these should consist of a header section summarising the logical structure, followed by the main body of the script. The main body should itself contain comments.

7. Refer to the feedback you received on in-course assessment 1.

STAT0030 Assessment 2 — Technical hints

1. The dataset contains a mix of categorical and numerical covariates and will be automatically saved as a `data frame` in R. However, some of the advanced regression methods require an object of type `matrix` for the covariates. To convert a caterogical variable (stored as a `factor` variable in R) to a numerical one, you will need to create dummy variables for each category present. You can write your own function to do this, or use an R library such as `fastDummies`. To convert a data frame with numerical values `x` to a matrix, use `as.matrix(x)`.

2. You are not expected to build very complicated models for this assessment. Instead, we want you to be able to demonstrate your understanding of the advantages and disadvantages of different models, how to think about model building, and how one might go about comparing different properties of the models.

3. You are allowed to use R libraries for this assessment. However, you need to make sure that you fully understand what each function is doing and briefly explain it either in your writeup or in your code. Unnecessary use of libraries beyond what we have used in STAT0030 (or ones mentioned in these instructions) may result in a lower mark.