URBS362
Quantitative Research Methods

ASSIGNMENT 01 – REGRESSION ANALYSIS
PREDICTING HOUSES SELL PRICES

You need to explain the variation in sell prices for a sample of houses in the city of Windsor, Ontario. To do so, you'll use **R** and **RStudio**[1]. The dataset is named HOMES.csv (comma delimited). The dataset contains 50 observations. The variables are the following:

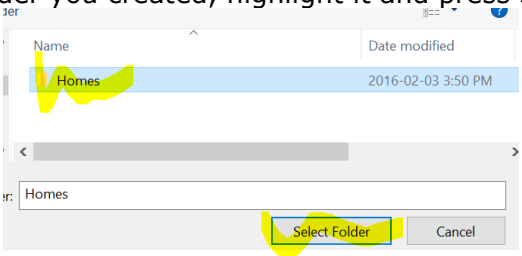
ID: Just an ID field for easier referencing.
SELLPRIC: The price at which the house was sold (in thousands of dollars). Continuous.
LIVINGAREA: The living area size (in thousands sq. ft.). Continuous.
ROOMS: Number of rooms. Discrete.
BEDRMS: Number of bedrooms. Discrete.
BATHRMS: Number of bathrooms. Discrete.
AGE: Age of the house since construction (in years). Continuous.
ACRES: Size of lot (in acres). Continuous.
TAXES: Last property taxes paid (in dollars). Continuous.

You're not yet familiar with R and we won't learn regression analysis with R in a lab. These instructions will give you commands to write and some information regarding these commands. The assignment will of course be marked based on the rights commands, but mostly on your understanding of the outputs. Questions (and how many points they're worth) are written in *italics* and have to be answered in a different document, that you will hand the **29th of February, before 4:00PM** (5% deduction per day, starting at 4:01PM). The assignment is done **individually**. Use three decimal points when reporting values.

1. Download the HOMES.csv file. Create a new folder in your computer to store the dataset.

2. First, we need to create a workspace in RStudio, to load a dataset. After installing R and RStudio (they're already installed in the GIS Lab on the 12th floor), open RStudio.

File menu > New Project > Existing Directory > Browse
Locate the new folder you created, highlight it and press Select Folder.



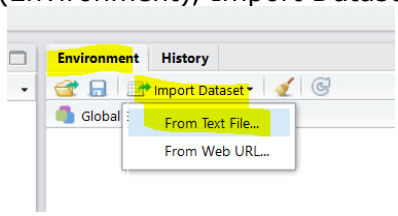
Press Create Project and wait.

File menu > New File > R Script.

[1] You need to download R first (free and open source) here : http://cran.stat.sfu.ca/ .
Then, you have to install RStudio (a graphic user interface) over R, right here:
https://www.rstudio.com/products/rstudio/download/
They work both on Windows, Mac and some Linux distros.

Instead of typing all commands in the console, we will use a script. That leaves a trace of everything we did. To run a command, put the typing cursor on the command line and press Ctrl-Enter (or highlight several lines of command to run them all simultaneously).

On the top right panel (Environment), Import Dataset > From text file.



Import HOMES.cvs. Make sure the dataframe (bottom right) looks fine, with headers and values. If so, press Import. Have a look at the dataset HOMES.

In your R Script (probably still named Untitled1), type down:

```
attach(HOMES)
```

With the cursor still on that line, hit Ctrl-Enter (I won't write this everytime, just consider that all commands have to be followed by Ctrl-Enter!). You just loaded the dataset in the data frame. Just to be sure it is loaded:

```
names(HOMES)
```

It should display the variables' names, within quotation marks, in the Console (bottom left).

Why don't you save the R Script right now (do that a couple of times during the assignment, a Ctrl-S is quick and easy).

File menu > Save

Choose a name that makes sense, like HOMES_Script, then press enter.

Phew! that was long. Let's start working on our model.

3. Because we are intending to build a linear regression model, it is important to inspect the normality of the distributions. That normality is an assumption, because the assessment of the significance relies on a normally distributed model. It is also important to assess the linearity of the relationships between the dependent variable and the independent variables. We will do both at the same time. First, install a new package in R (that might take a short moment), then load it in RStudio. Don't forget the Ctrl-Enter after typing each command. The smoother FALSE means that we removed some lines from the plots, for easier reading. Try TRUE if you want to see.

```
install.packages("car")
library(car)
scatterplotMatrix(~SELLPRIC+LIVINGAREA+ROOMS+BEDRMS+BATHRMS+AGE+ACRES+TAXES
, smoother=FALSE)
```

In the bottom right panel, a matrix of 8x8 plots will be displayed. It is tiny. Press Zoom right above for a larger picture. In the diagonal, you have the shape of the distribution: do the look normal? Don't mind too much the discrete variables (ROOMS, BEDRMS AND BATHRMS), we will use them as is.

Q1 (1): Other than these three, which variables don't look normal?

This means we will have to transform the non-normal variables. We will make some use of placeholder names. Placeholder names replace the original names of the variables. In the following command, y is the placeholder name (y is simple enough and easily interpretable, this is the dependent variable).

```
y<-log10(SELLPRIC)
```

If you need to transform LIVINGAREA in log10, use the following command. AreaLOG is the placeholder name, and you're asking R to transform LIVINGAREA into Log10. Do something similar for each non-normal variables (except the three discrete ones).

```
AreaLOG<-log10(LIVINGAREA)
```

Notice in the top right panel, below Data, there is a Values section, where you can see the placeholder names, the newly transformed variables, in other words. They should all state *num [1:50]*, which means numerical data, with 50 observations.

Rerun the scatterplotMatrix above, but change the variable names if necessary! (e.g. SELLPRIC should become y.) Be very careful here, because R is capital sensitive. If you named a transformed variable AreaLOG, only this name will work, and not arealog or Arealog or AreaLog. However, when you start typing variables names, RStudio might be good enough to suggest you existing names, which is a nice feature.

How does it look like? Do the distributions look more normal that way? Probably.

Q2 (3): Except the three discrete variables, which transformed variables are still slightly non-normal?

4. It is also important to assess the linearity of the relationships between the dependent variable and the independent variables. Look at the first column in your series of plots: these are relating y with all the independent variables. While not always perfect, the relationships look somehow linear; it will be good enough for the purpose of this assignment.

Q3 (2): One scatterplot shows a negative relationship, which?

5. To find the Pearson's correlation coefficient, use this simple command (in this case, it finds the Pearson's *r* between AcresLOG and y):

```
cor(AcresLOG,y)
```

Q4 (4): Report the Pearson's correlation coefficient, for each relationship between y and the continuous independent variables.

6. Build a multiple regression model, between the dependent variable (y) and the independent variables, using a command's template such as this (remember to replace the x by the actual variable names):

```
summary(lm(y~x1+x2+x3+x4+x5+x6+x7))
```

Q5 (2): Report R2, Adjusted R2 and the p-value. Is the model significant?

Q6 (2): Write the regression equation.

Q7 (2): Which independent variables are not significant?

7. Rebuild the model again, but this time, remove all variables that were far from being significant. If an independent variable was almost significant, it is suggested to keep it in the model anyways, as it might become significant with a different model. Use the same command seen at step 6.

Q8 (2): Report R2, Adjusted R2 and the p-value. Is the model significant?

Q9 (2): Write the regression equation.

Q10 (1): Which independent variables are not significant?

8. Let's find out if the residuals are normally distributed and if there is a problem of heteroscedasticity. Simply use the following layout command:

```
layout(matrix(c(1,2,3,4),2,2))
```

Then use this command template (change the variables' name):

```
plot(lm(y~x1+x2+x3+x4))
```

We will use only one plot ("Residuals vs Fitted"). This plots the residuals (y-axis) (Observed values – Predicted values), therefore positive values mean the model under-predicted (Observed was greater than Predicted) and negative values, the opposite. The x-axis (Fitted Values) represents the log transformed SELLPRIC variable.

Q11 (2): Do the residuals look like they're randomly distributed around Y=0 (the dotted line)?

Q12 (4): Write a short interpretation (one sentence) of this plot, what is the relationship between the residuals and the Fitted values?

9. This suggest a problem in the distributions, so we will look for diagnostic tests. First, we want to inspect colinearity: are the independent variables correlated together, in other words, do they tell the same story? Use the following template:

```
vif(lm(y~x1+x2+x3+x4))
```

VIF stands for Variance Inflation Factor (Rogerson's textbook covers this test, take a look!). As a rule of thumb, a VIF > 2 is slightly problematic, but not dramatic. However, a VIF > 5 means at least two of the independent variables are highly correlated.

Q13 (2): Is there a colinearity issue with the current model? Why?

10. We will build a new scatterplot matrix, but only using the variables used in the current model (rather than with all the variables, as we did in step 3). Use the same command as step 3, but with proper variables.

    Look for all the possible relationships between the independent variables. Even if the VIF was not problematic, we still see some correlation between some variables.

Q14 (4): As you did in step 5, find the Pearson's r between all the independent variables and report them. Which variables are the most correlated?

Q15 (2). Build a new model, but remove one of the correlated variables (but keep the other!). Keep the one that you think, as a researcher, would theoretically and statistically be more influential on SELLPRIC (or, in this model, the log transformed SELLPRIC: y). Report R2, Adjusted R2 and the p-value. Is the model significant?

Q16 (2): Write the regression equation.

Q17 (1): Which independent variables are not significant?

11. As we did in step 8, build the four plots and look for the Residuals VS Fitted plot.

Q18 (2): What command did you write to build the plots?

The plot probably suggests some heteroscedasticity. However, the residuals plot also labels some observations that might be problematic, as outliers.

Q19 (2): Which observation's IDs look like they could potentially be outliers?

12. We will build a new model that doesn't include these outliers. To do so, we will create a modified dataset and subsequently use it. In the following command, we create a placeholder name for the modified dataset (Hmod, which stands for HOME modified). Then, we specify RStudio that this modified dataset should come from the original HOMES dataset, but excluding three observations (the operator *!=* means *not equal to*). Replace *NN* in the following command by the ID labelled on the previous plot (your answers in Q16).

```
Hmod <- HOMES[ which(HOMES$ID !=NN & HOMES$ID != NN & HOMES$ID != NN), ]
```

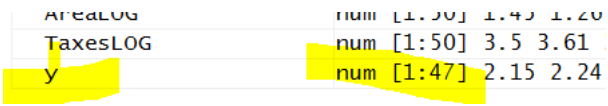
13. Attach the dataset:

```
attach(Hmod)
```

14. Because this new dataset excluded some observations, we need to retransform its original variables into log10. We will basically overwrite the previous placeholder names, but this time we have to make sure that there are 47 observations and not 50.

```
y=log10(Hmod$SELLPRIC)
```

Notice, from the top right panel, that y now has 47 observations and not 50. This is what we are aiming for.



Do the same again, but for the three independent variables left, such as this:

```
AcresLOG=log10(Hmod$ACRES)
```

They should all have 47 observations.

Q20 (2): Rebuild the model again, with this new modified dataset (the variables names stayed the same, if you followed the previous steps). Report R2, Adjusted R2 and the p-value. Is the model significant?

Q21 (2): Write the regression equation.

Q22 (1): Which independent variables are not significant?

15. If you build the Residuals vs Fitted plot again, you will notice that there is still some problem with the residuals distribution, the model can't predict with very high accuracy houses at the lowest and highest sell prices. Maybe the model is missing independent variables. Install the *lmtest* package and load it. It contains the Ramsey's RESET test that looks for omitted variable. If the p-value is less than 0.05, the model has omitted variables and the researcher probably wants to add some (only if possible; are the available variables significant and independent from the others?); if it's greater than 0.05, then the model probably doesn't need more independent variables. In the *resettest* command below, remember to change the x with the real variable names.

```
install.packages("lmtest")
library("lmtest")
resettest(lm(y~x1+x2+x3))
```

Q23 (3): How do you interpret the RESET's output? What could you do, if this was the only dataset available to you?

While not perfect, the residuals plot is better than with our previous models. Other methods could prove helpful (bootstrap resampling, Weighted models), but they are out of scope for this course. However, further investigations from the researchers could also reveal useful, for example by exploring more in depth unusual and influential observations, as well as the explanatory variables.