Programming (FEB22012[X])

1. Exercise

Deadline for submission: 2014-09-14 23:59 CET

Instructions / Matlab

In your MATLAB software, you can enter and execute commands in the MATLAB command window. Series of commands can be saved as a script, which can be executed from the command window. Commands thus executed all operate in MATLAB's base workspace. Please keep this in mind when running (multiple) scripts, as scripts may modify existing variables.

Functions on the other hand accept input from and return output to their callers. Each function operates in a separate workspace, which cannot be accessed from another function nor from the MATLAB base workspace. Note that the "workspaces" are a bit different than scoping in Java: variables defined in the global level cannot be referred from within functions. The functions cannot be defined in the MATLAB command line or script files, but must be defined in separate function files. A function file always begins with a function definition line, specifying the signature of the primary function, (name, outputs and inputs). This primary function can be called from the command line, a script, or another function. The name of this primary function should be an exact case-sensitive match with the name of the function file (without the *.m extension). Besides the primary function, a function file may contain additional functions. These functions are only accessible from within the function file.

When making an assignment, you are encouraged to write appropriate functions as well as one working script file demonstrating how to use these functions in order to solve the problem addressed in the assignment. If you do not know how to do something, MATLAB has an extensive help system accessible by entering doc in the command line or by clicking the question mark icon.

The basic language constructs in Matlab are:

```
x = 2; % declaration of a variable is not required

if (x == 2) % blocks in Matlab end with 'end'
        x = 3;
elseif (x == 3)
        x = 4;
else
        x = 5;
end

for (i = 2:5) % for-loops can use integer sequences with ':'
        i % result of statement without semicolon is printed
end

while (i < 10)
        i; % result of statement with semicolon is not printed
end
i # print out value

function b = mysquare(a) % parameters: a, return value: b
        b = a*a;
end
```

```
c = mysquare(2); % call function

A = [2, 3; 4, 5]; % construct matrix
A = A'; % transpose matrix
A = A * A; % matrix multiplication, A = A^2
```

Instructions / ordinary least squares

In this exercise we will implement a linear regression model estimation using matrix operations. Linear regression is taught in more detail during the "advanced" statistics course of the second year of the econometrics program and if you do not know about it yet, read the explanation in `http://en.wikipedia.org/wiki/Linear_regression`. Let us consider a multiple linear regression model with n observed responses on p regressors. Then the model can be stated as $n \times 1$ vector of responses $\mathbf{y}$, $p \times 1$ vector of unknown variables $\boldsymbol{\beta}$, $n \times p$ matrix matrix of regressors $\mathbf{X}$, and $n \times 1$ vector of error variables $\boldsymbol{\epsilon}$ with:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{1}$$

The constant term is included in the set of regressors $\mathbf{X}$ by taking $x_{i1} = 1 \; \forall i \in \{1, \ldots, n\}$. The coefficient β_1 corresponding to this regressor is the intercept. Minimizing the sum of squared residuals of this model provides us the OLS estimator (`http://en.wikipedia.org/wiki/Ordinary_least_squares`) for $\boldsymbol{\beta}$ with the following formula:

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X'X})^{-1}\boldsymbol{X'y} \tag{2}$$

Exercise

Download the data set from `http://smaa.fi/static/prog2/2011/data/food.dat`. The dataset consists of 40 observations of household weekly incomes (in 100$'s, first column) and food expenditures (in $'s, second column), and two other variables that are not of interest to us. Make a Matlab program that

1. Implements Ordinal Least Squares estimator (Eq. 2) as a function taking as parameters:
   (a) matrix of observations
   (b) column index of the variable to be predicted
   (c) set of column indices of the regressor variables to use

   and returning the $\hat{\beta}$.
2. Loads the data set into Matlab
3. Calculates parameters of the regression model using the OLS estimator
4. Plots the data points as a scatter diagram together with the regression line. The x-axis should be the weekly income and y-axis the food expenditures. Name the axes appropriately and color the data points plotted in black and the regression line in red. Include the functional form of the regression line next to it in the diagram.

For plotting, see Matlab's built-in function `plot`. For more information about matrix computations in Matlab, see `http://www.mathworks.nl/help/techdoc/learn_matlab/f2-8955.html`.

Instructions / submission

Include in each source file your names and student numbers as a comment in the beginning of the file. Submit the exercise as a zip file containing only the source files (.m) in root of the zip. Submit via Blackboard. Note that incorrectly submitted or non-running exercises are automatically awarded 0 points. Remember to document your code and use descriptive variable names.