

Estimating Macroeconomic Variables

1 Introduction

This is a problem I wrote as a part of a training course I helped lead while at the Board. The model isn't all that useful in its own right but helps to teach some fundamental programming skills. Save the resulting output from your solutions in a *list* object named **answers**. The data can be found in `data_agg_econ.csv`.

2 Problem

Variable name	Description	Desired movement
<code>gdp_chng</code>	% Change in gross domestic product (annual)	↑
<code>exprt_chng</code>	% Change in net trade exports (annual)	↑
<code>unemp_chng</code>	% Change in unemployment (annual)	↓
<code>prices_chng</code>	% Change in consumer prices (annual)	↑

We are proposing a model that aims to predict whether GDP will increase or decrease in the next year given knowledge of how our other variables have moved in past years. The model we are working with looks like,

$$\hat{G} = \gamma_1 T_{k_1} + \gamma_2 U_{k_2} + \gamma_3 P_{k_3}$$

where T , U , and P are scalar binary variables referring to the behavior of `exprt_chng`, `unemp_chng`, and `prices_chng` respectively. We denote our independent variables as $\vec{X} = \{T, U, P\}$. \hat{G} is our dependent variable, it is *not* binary and can take any real value $[0, 1]$. Our observed G can only be zero or one depending on if GDP decreased or increased respectively. The index vector $\vec{k} = \{k_1, k_2, k_3\}$ represents the number of past years over which we will determine if a majority of the annual changes were desired changes in these variables. Majority defined as greater than half ($\frac{k_i}{2}$) of the observations. The vector $\vec{\gamma} = \{\gamma_1, \gamma_2, \gamma_3\}$ represents the weights we put on each variable, its elements can be any real value. As an example if the data showed `unemp_chng` = $\{.05, -.01, .04, -.12\}$ from 2013 through 2016 (we are at year-end of 2016), U_{k_2} (where $k_2 = 3$) would be **1** because there were two decreases (which are desired for unemployment rates) out of the last three annual changes. U_{k_2} (where $k_2 = 4$) however would be **0** because only two of the last four changes has been desired. \hat{G} represents our prediction of the change in GDP, values closer to 1 indicate a stronger prediction of an increase. Please note that \hat{G} must be between 0 and 1, which means that your model is subject to the constraint that the $\vec{X} \cdot \vec{\gamma} \in [0, 1]$ (\cdot is the dot product).

The goal of the model and optimization is to predict if G is 0 or 1 in the next year, representing a decrease or increase in GDP respectively. To do this we need to determine *both* what the optimal γ 's and set of k 's are. Optimality is defined as giving the best predictions on a training set, where we minimize the difference between our predicted \hat{G} and observed G . We will lead you through the beginning stages of reaching this goal but the final step requires completing the last problem which is not mandatory but is encouraged!

Problem 2.1. Create a function to compute T , U , and P for a given \vec{k} and single year. For your answer assume $\vec{k} = \{8, 4, 6\}$ and use 2012 as your current year. This means that your results, `chngs_846` which will be saved in **answers**, will be a vector of 1's and 0's of length three. Save the function in your code but you do not need to add it to the **answers** data.

Problem 2.2. Use `chngs_846` as your \vec{X} to compute a solution (there are multiple) for $\vec{\gamma}$ where each element is non-zero and < 1 . Your observed G will correspond to GDP's change in 2013, as if you were trying to predict it only knowing data from 2012. To do this, minimize in the mean squared sense the difference between \hat{G} and the observed data G . Create a function to find this $\vec{\gamma}$ as it will be used later. Save this vector as `weights_given`, similarly this will be saved in `answers`. This minimization would look like,

$$\min_{\vec{\gamma}} \frac{\sum_{i=1}^n (\hat{G} - G)^2}{n}.$$

Please note that $n = 1$ here as we are only working with one year's data. In later problems n will be larger.

HINT: Modularize your function so that the function returning the mean squared difference is called within the optimization function. Use an optimization routine such as the `constrOptim` in the `stats` package. When using these you need to specify an initial "guess", θ , choose anywhere in the solution region. The constraint setup is written below, you will still need to translate it to R. A is the constraint matrix, x is what we are searching for, and b is our constraint vector.

$$Ax \geq b$$

$$\begin{pmatrix} T & U & P \\ -T & -U & -P \end{pmatrix} \begin{pmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{pmatrix} \geq \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

Problem 2.3. Using $\vec{k} = \{8, 4, 6\}$ as above, compute your $\vec{X} = \{T, U, P\}$ for all periods in the data, with the exception of 2016 as we don't have a 2017 G to test against. Remove any years for which not every element in \vec{X} can be computed. Save this as a dataframe, of dimension 60×3 , as `X_mtx` in `answers`. For notational reasons we will refer to this matrix (set of \vec{X} 's) as X_M .

Problem 2.4. Compute X_M dataframes for each of the five \vec{k} 's shown below, over the appropriate years for each \vec{k} . Determine which \vec{k} leads to the best prediction of G given a constant $\vec{\gamma} = \{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\}$, again by "best" in the mean squared sense. This will be computed over all the rows of X_M not just one, so \hat{G} and G will be vectors spanning the appropriate observation time. Save only the \vec{k} that results in the best prediction as `k.bestof5` in `answers`.

$$\begin{aligned} \vec{k}^A &= \{1, 1, 1\} \\ \vec{k}^B &= \{2, 2, 3\} \\ \vec{k}^C &= \{12, 3, 4\} \\ \vec{k}^D &= \{5, 7, 9\} \\ \vec{k}^E &= \{6, 3, 4\} \end{aligned}$$

Problem 2.5. In the previous problems you've found both an optimal \vec{k} given an arbitrary $\vec{\gamma}$ and an optimal $\vec{\gamma}$ given an arbitrary \vec{k} . Now you will write function(s) to do both at once. Remember that the possibilities for \vec{k} are finite, as they must be integers where none can be greater than the number of years we have in our data, this puts useful constraints on the solution. Use a package function such as the one suggested above to do your constrained optimization. As a starting point we have included a pseudo-code algorithm, see Algorithm 1, that gives a hint as to a code structure you could use. This is an open ended question on purpose, we will reward hard work and creativity not simply having the correct solution.

COMMENT: I thought it best to do a nested optimization function because the inequality constraints for the $\vec{\gamma}$ search depend on having an X_M which cannot be created without already having a “guess” for \vec{k} . If you can figure out a way to do the optimization at once, simultaneously finding $\vec{\gamma}$ and \vec{k} , I would be very interested to hear about it.

Algorithm 1 Algorithm

```

1: procedure FIND  $\vec{\gamma}$ 'S AND  $\vec{k}$ 'S
2:   Define inputs  $G_{\text{obs}}$  and your initial guesses for  $\vec{k}_\theta, \vec{\gamma}_\theta$ 
3:   Enter outside minimization routine to find optimal  $\vec{k}$ 
4:     Subset  $G_{\text{obs}}$  based on initial guess for  $\vec{k}$ 
5:     Compute constraint vector and matrix for  $\vec{k}$  search
6:     Optimize over  $\vec{k}$ 's
7:       Enter inside minimization routine to find optimal  $\vec{\gamma}$ 
8:         Subset  $G_{\text{obs}}$  based on each iteration's choice for  $\vec{k}$ 
9:         Compute  $\mathbf{X\_mtx}$  for current  $\vec{k}$ 
10:        Compute constraint vector and matrix for  $\vec{\gamma}$  search
11:        Optimize over  $\vec{\gamma}$ 's
12:        Objective function:  $\min_{\vec{\gamma}} \frac{\sum_{i=1}^n (\hat{G} - G)^2}{n}$ 
13:        Return  $\vec{\gamma}^*$ 
14:        Objective function:  $\min_{\vec{k}} \frac{\sum_{i=1}^n (\hat{G} - G)^2}{n}$ 
15:        Return  $\vec{k}^*$ 
16:   Return  $\vec{k}^*, \vec{\gamma}^*$ 
17: Exit procedure

```
