GR5065 Assignment 6

Due by 4PM EST on April 30, 2020

1 Coronavirus

Read

http://www.robertkubinec.com/post/kubinec_model_draft/

which presents a Stan model for the number of coronavirus tests and cases in the United States. The code and data can be cloned from https://github.com/saudiwin/corona_tscs or you can click on the "Clone or download" button followed by the "Download ZIP" link that can then be unzipped in your working directory. The Stan program is in the `retrospective_model_paper` directory and called `corona_tscs_betab_scale.stan`.

1.1 Prior Predictive Distribution

Write a Stan function to draw (once) from the prior predictive distribution of the gross number of coronavirus infections across U.S. states and time using the model described in the last section on "Identifying the Latent Scale". However, instead of using a standard normal prior for

$$\log q_{ct} - \log I_{ct}$$

put a prior of your choice on the ratio

$$\frac{q_{ct}}{I_{ct}}$$

where q_{ct} is the proportion of country / region c that has been tested for the coronavirus at time t and I_{ct} is the proportion of country / region c that is been infected at time t.

You might utilize the fact that on Thursday, it was announced that there have been an estimated 2.7 million infections in the state of New York according to tests for coronavirus antibodies that were administered somewhat randomly at places like grocery stores. There had been about 400,000 tests for the coronavirus in the state of New York.

Your Stan function should return a single matrix with rows equal to `num_country` and columns equal to `time_all` and cells being the number of infections then and there. The arguments to your Stan function should be much the same as the symbols in the data block of the `corona_tscs_betab_scale.stan` file, like

```
functions {
  matrix coronavirus_PPD_rng(int time_all, int num_country, int[] cases,
                             int[] tests, int[,] time_outbreak, int S,
                             matrix ortho_time, matrix suppress,
                             matrix time_outbreak_center, vector count_outbreak,
                             real phi_scale, vector country_pop) {
    matrix[num_country, time_all] infections;
    // fill in the rest
    return infections;
  }
}
```

You will have to copy the body of the transformed data block into your function in order for it to work properly. When you are finished, make sure it compiles by calling `rstan::expose_stan_functions` and call it once using the same elements of the list called `real_data` as in the paper. However, you will need to change the matrix stored as `time_outbreak` into a R list with one element for each row of the `time_outbreak_matrix`.

1.2 Mistakes

What, if any, are the mistakes in the `corona_tscs_betab_scale.stan` file, in the sense that they are inconsistent with what is described in the text of the paper?

2 Constant Elasticity of Substitution (CES) Models

Read section 1, 2, and 5.2 of

https://cran.r-project.org/web/packages/micEconCES/vignettes/CES.pdf

The other sections just present a severely Frequentist approach to estimating the parameters of the CES model given in equation 1 and other variants. We are actually estimating the parameters of equation 4, but when doing so (in Stan), it is better to take the natural logarithm of both sides of equation 4 and treat the right-hand side as a conditional expectation so that the assumption of normally distributed, additive errors is more plausible.

Section 5.2 attempts to replicate published results in a previous study, which results in a spectacular failure where the point estimator errors out, results in parameter point estimates that are not "economically meaningful", or does not resemble the published point estimates. You can set up the data with

```r
data(GermanIndustry, package = "micEconCES")
GermanIndustry$time <- GermanIndustry$year - 1960
GermanIndustry <- subset( GermanIndustry, year < 1973 | year > 1975)
GermanIndustry <- GermanIndustry[ , c("Y", "K", "E", "A")]
summary(GermanIndustry)
```

```
##        Y                K                E               A
##  Min.   : 453.5   Min.   : 24.25   Min.   :561.0   Min.   :10.79
##  1st Qu.: 612.5   1st Qu.: 32.76   1st Qu.:609.8   1st Qu.:11.09
##  Median : 816.5   Median : 61.71   Median :628.5   Median :11.53
##  Mean   : 763.0   Mean   : 69.23   Mean   :645.7   Mean   :11.86
##  3rd Qu.: 869.9   3rd Qu.: 85.39   3rd Qu.:702.8   3rd Qu.:12.78
##  Max.   :1002.2   Max.   :145.54   Max.   :748.8   Max.   :13.16
```

where

- `Y` is "gross value added of the West German industrial sector (in billionDeutsche Mark at 1991 prices)"
- `x1 = K` is capital input, the "gross stock of fixed assets of the West German industrial sector (in billion Deutsche Mark at 1991 prices)"
- `x2 = E` is energy input, "the final energy consumption of the West German industrial sector (in GWh)."
- `x3 = A` is labor input, "the total number of persons employed in the West German industrial sector (in million)"

2.1 Stan Program

Write a Stan program to draw from the posterior distribution of the parameters in equation 4. However, γ, δ, δ_1, ρ_1, ρ, and ν should all be defined in the transformed parameters block as an inverse CDF transformation of a cumulative probability in the parameters block. Which inverse CDFs you use are up to you. In addition, in the generated quantities block, calculate the posterior distribution of the Allen-Uzawa elasticity of substitution and the Hicks-McFadden elasticity of substitution based on Appendix B.3, although you have to scroll up to the start of Appendix B to see all of the definitions.

2.2 Posterior Distribution

Call the `stan` function in the rstan package to draw from the posterior distribution of the parameters in your Stan progam condtional on the data on `Y`, `K`, `E`, and `A` in the `GermanIndustry` dataset. You should try to make the results free of any warning messages.

2.3 Comparison to Frequentist Estimation

The help page for the `cesEst` function (that estimates CES models) in the micEconCES package includes a helpful warning message that says

> Warning: The econometric estimation of a CES function is (almost) always very problematic, because very different parameter vectors could result in very similar values of the objective function (sum of squared residuals). Hence, even if the optimizer reports that the nonlinear minimization has converged, there might be another rather different parameter vector that results in a lower sum of squared residuals.

If so, the negative objective function across the parameter space will not be proportional to a multivariate normal distribution, which is the necessary assumption for testing a null hypothesis about any (combination of) the parameters.

Are any of these issues a problem for Bayesian estimation of a CES model using Stan? Explain why or why not.