Applied Statistical Programming - Spring 2022

Problem Set 6

Due Friday, April 29, 11:59 PM

Instructions

1. Build the Rcpp package described below. Be sure to provide many comments in your code blocks to facilitate grading. Undocumented code will not be graded.
2. Work on git. Continue to work in the repository you forked from https://github.com/johnsontr/AppliedStatisticalProgramming2022 and add your code for Problem Set 6. Commit and push frequently. Use meaningful commit messages because these will affect your grade.
3. You may work in teams, but each student should develop their own Rmarkdown file. To be clear, there should be no copy and paste. Each keystroke in the assignment should be your own.

The Expectation-Maximization Algorithm

The goal is to implement an ensemble of models. You will combine forecasts of US presidential elections using ensemble Bayesian model averaging (EBMA). To do this, you must decide how to weight each component of the forecast in the prediction. The collection of these weighted forecasts form the ensemble, and you will use something called the EM (expectation-maximization) algorithm to make estimates.

The task is to choose values w_k that maximize the following equation:

$$p(y|f_1^{s|t^\star}, ..., f_K^{s|t^\star}) = \sum_{k=1}^{N} w_k N(f_k^{t^\star}, \sigma^2) \tag{1}$$

Assume that the parameter σ^2 is known and that $\sigma^2 = 1$.

The first step of the EM algorithm is to estimate the latent quantity $\hat{z}_k^t$ that represents the probability that observation t was best predicted by model k.

$$\hat{z}_k^{(j+1)t} = \frac{\hat{w}_k^{(j)} N(y^t|f_k^t, 1)}{\sum_{k=1}^{N} \hat{w}_k^{(j)} N(y^t|f_k^t, 1)} \tag{2}$$

In this equation, j is the particular iteration of the EM algorithm, and $N(y^t|f_k^t, 1)$ is the normal cumulative distribution function evaluated at the observed election outcome (The Rcpp equivalent to `dnorm(y, ftk, 1)`).

The second step of the EM algorithm is to estimate the expected value of the weights assuming that all $\hat{z}_k^t$ are correct.

$$\hat{w}_k^{(j+1)} = \frac{1}{n} \sum_t \hat{z}_k^{(j+1)t} \tag{3}$$

The estimation procedure is as follows:

1. Start with the assumption that all models are weighted equally.
2. Calculate $\hat{z}_k^{(j+1)t}$ for each model for each election.
3. Calculate $\hat{w}_k^{(j+1)}$ for each model.
4. Repeat steps 2-3 until convergence to a pre-defined tolerance of your choosing.

The Assignment

1. Write an Rcpp function that will calculate the answer to Equation (2). The output will be a matrix.
2. Write an Rcpp function that will calculate the answer to Equation (3). The output will be a vector.
3. Write an Rcpp function that will complete the entire algorithm.
4. Include one unit test per function.
5. Assemble the code as an R package.
6. Write a development .R file that demonstrates the package's use.