Joe Hakim, Steven Chen, Christopher Le, Richard Liu

Introduction to Computational Medicine I

Project 1 Part 1 – Written Report

Cardiac output (CO) in patients is an essential measurement for the diagnosis and treatment of many different cardiac diseases. Currently, one of the only ways to measure CO is via thermodilution, which involves very invasive procedures. This method is both time consuming and usually only limited to critically ill patients. In order to make these measurements more easily accessible to other patients, recent studies have developed methods to estimate CO from other more easily obtainable measurements like arterial blood pressure (ABP). In this project, we used the PhysioToolkit MATLAB functions used in an article written by Sun et. al. (2009) to replicate their methods in estimating CO from ABP waveforms. The data we used were from real patients in the MIMIC II database.

In order to reproduce the figures shown in the paper, we created a MATLAB code that utilizes functions in the PhysioToolkit to extract data points based on ABP waveforms of patients. In general, we imported the ABP data, processed the data using the 2analyze functions, and superimposed points corresponding to the onset time, end of systole (estimated by 0.3\*sqrt(beat period)), and end of systole (estimated by the lowest non-negative slope method), onto a plot of 20 pulses of the ABP waveform. There are two plots for each of the four patients, one starting at 10 hours, and one starting at 11 hours.

The core of this part was to overlay features of waveforms on a continuous set of highly-sampled real-time data. The features, most notably the end of systole times, depended on a relatively high signal-to-noise ratio of the waveform. Note that each differentiation of data decreases this ratio, so since the “first-minimum-slope” method relies on the slope of the curve, the accuracy of this placement critically depends on the quality of the data, which we qualitatively observed to be sufficient. Otherwise, a less naive method of looking at the slope (including smoothing, etc.) would be necessary to accurately place the feature.

The other features were easy to place, comparatively. We chose four different patients, each of which had qualitatively dissimilar waveforms, and the script was able to precisely locate systole, diastole (both not shown in graph), as well as onset time. The end of systole time is less consistent, and essentially determined by one of two highly heuristic strategies, so the exact placement was different across patients with different shaped waveforms.

One exception we found was in patient 1502. The shape of the waveform was such that the troughs had similar shape as the peaks, so the features assigned placed a new onset twice for every individual pulse. To resolve this, the code assessing the onset times would have to be changed. Specifically for patient 1502, the problem is that there are two troughs being recognized as diastole. So a possible resolution would be to detect when the diastolic or systolic pressures alternate with period two (with sufficient distance in between), and adjust the detection threshold to have less volatile systole/diastole points.

After using PhysioToolkit and the estimateCO\_v3.m function to obtain cardiac output estimates from the arterial blood pressure data, we calibrated it using the C2 method from the article. The C2 method tweaks the CO estimates by a factor of the first cardiac output measurement from thermodilution divided by the corresponding ABP measurement.

In part 4, we first obtained 3 estimates of CO using different estimators, with IDs 5 (Liljestrand), 6, and 7. We then loaded the n file data into MATLAB, isolated the cardiac output from thermodilution (COtd) values, and recorded the time and value of the first nonzero COtd output. We then scaled the CO estimates using the C2 calibration, and plotted the 3 different estimates on separate graphs over the span of 12 hours, with the COtd values superimposed on the graphs. We also plotted additional graphs for heart rate, MAP, and PP along the same 12 hour time interval, the data to which was given from the abpfeature.m function.

At first glance, the CO estimators with IDs 6 and 7 seem to more accurately estimate CO than estimator 5 based on the given COtd measurements. However, both estimators 6 and 7 seem to fluctuate more than estimator 5 does, especially in subject 20. Since cardiac output should be relatively stable over a long period of time, estimator 5 still seems to more accurately estimate CO.

The calibration of the predicted CO data vs. the measured thermodilution (TD) data, which is the normalization of the predicted CO waveform to actual clinical data, was achieved in two ways. C1 involved using the root-mean-square of the TD data from the entire time of measurement, and was more accurate than C2, which only normalized using the first TD point. However, in a clinical setting, C2 is more realistic to have, in that the entire point is to reduce the need of invasive TD measurement, and to be able to predict from a smaller amount of sampling. Hypothetically, a more flexible solution could include a strategy to dynamically update the calibration for each of any number of TD measurements. This way, the CO estimation can be as accurate as the amount of data permits, and if the physicians happen to have more TD data, the system could adapt to utilize them.

Joe Hakim contributed to the code to part 3, Richard Liu and Chris Le contributed to the code to part 4, Steven Chen contributed to the run-time script, and each team member contributed to both the report and presentation slides. Each team member revised the code as concerns arose regarding functionality and style.