STA 250, HW 2

Longphi Nguyen

November 12, 2013

Problem 1

quantile	0%	25%	50%	75%	100%
SE	0.008525917	0.009691226	0.009961909	0.010279175	0.011785323

Table 1: Quantiles of SE values

The algorithm calculates r =job mod 50 and s =job div 50 to set the seed values per job. There are 5 subsamples, obtained by randomly selecting n^γ observations (with uniform probability and without replacement) from the full dataset. For each subsamples, there will be 50 boostrap samples obtained using a multinomial distribution with n size. Then, use the bootstrap sample to a a least squares fit. From Table 1, the SE's are fairly small, as expected. And, from Figure 1, there are no patterns in the SE's that would raise a red flag. So, it is expected to that this worked fairly well.

Problem 2

1. **Mapper("x y")**. Outputs "$x_{hi}, y_{hi}, 1$", which are x and y rounded up to the first decimal.
2. **Reducer("x,y ,1")**. A box is determined by $(x - .1, x, y - .1, y)$. Whenever new inputs x^* and y^* fall in this box, increase the bin count by 1; else, print the number of counts for the box, form a new box with x^* and y^* being the upper bounds, and reset the count to 1.
3. **Implementation**. Mapper() rounds x and y up since the lower bound is supposed to be strictly less than, so we don't want to accidentally output something that looks to be a part of a different box. Also, Mapper only needs to print the upper bounds, since the lower bounds can be determined from them by subtracting .1. Since Hadoop will sort by these upper bounds, the Reducer() then relies on this sorting to determine when a new box occurs.
4. **Time**. Using 2 large core instances, the Mapper() finished in $12:33$ minutes, and the Reducer() finished in $22:45$ minutes. Based on Figure 2, the results seem reasonable because it's pretty.
5. **Improvement**. My Reducer() has an if statement to avoid printing the first read line. However, this check is done for each stdin line, which shouldn't be necessary. So, to avoid redundancy (and possibly a little computation time), this should be fixed. This can likely be done by doing what needs to be done for the first line before the for loop.

Problem 3

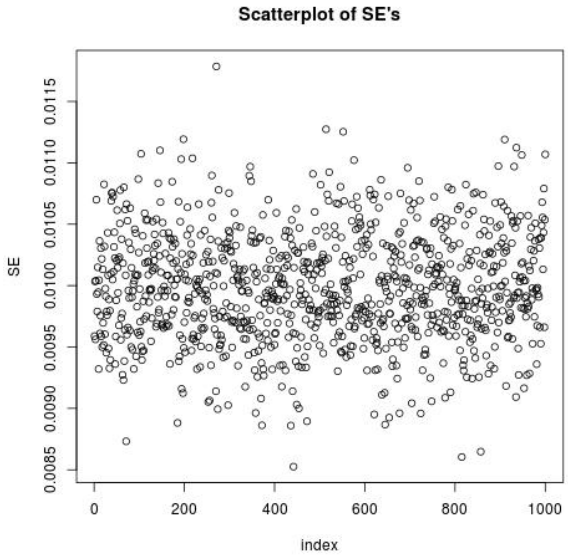

Figure 1: SE for Little Bag of Bootstraps

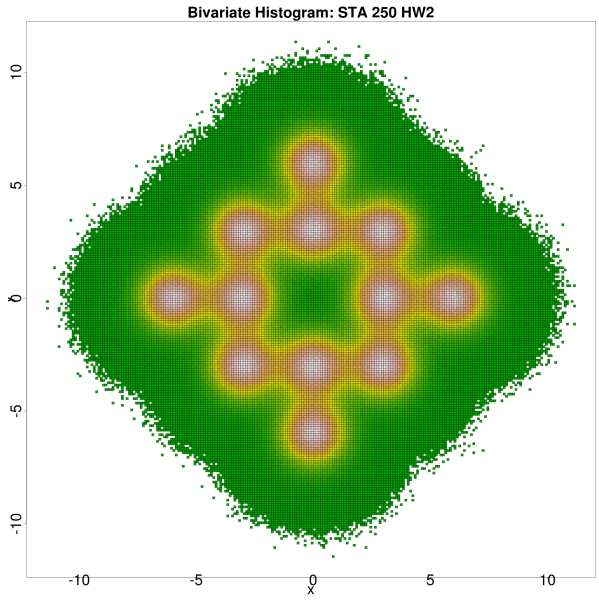

Figure 2: Histogram of Bivariate

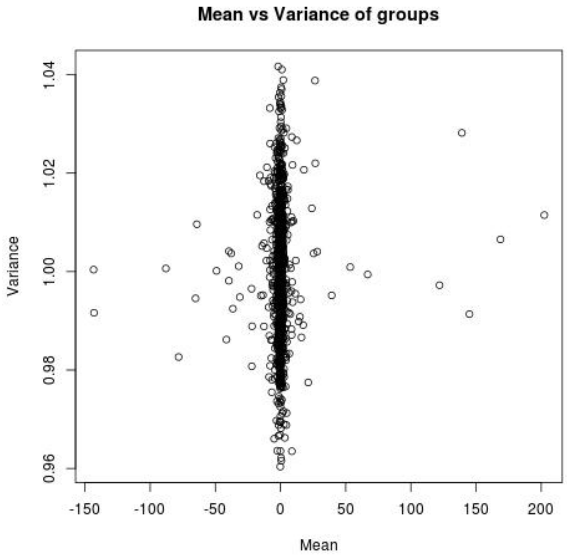

Figure 3: Within-group mean vs variance