GRADUATE STUDENT STAT 840 A1

Vsevolod Ladtchenko 20895137

Problem 5

(a)

Type 1 error is when we reject the null hypothesis when it is in fact true. Since the null hypothesis states that $H_0 : \theta = 0$, we will simulate this distribution, and see how often we reject this hypothesis, meaning that the data from $N(0, 1)$ suggests to us that the distribution is $N(\theta, 1)$.

The sequential testing algorithm: based on the description, I assume we need to sample individual points in a loop, and if the test statistic is above c, we count that as a type 1 error occurence. There must be some limit to how many times we sample an individual point, say n times. So if the test statistic is above c at any point between 1 and n, type 1 error occured.

This is the same as computing the test statistic for all n points, and then checking if any of them is above c.

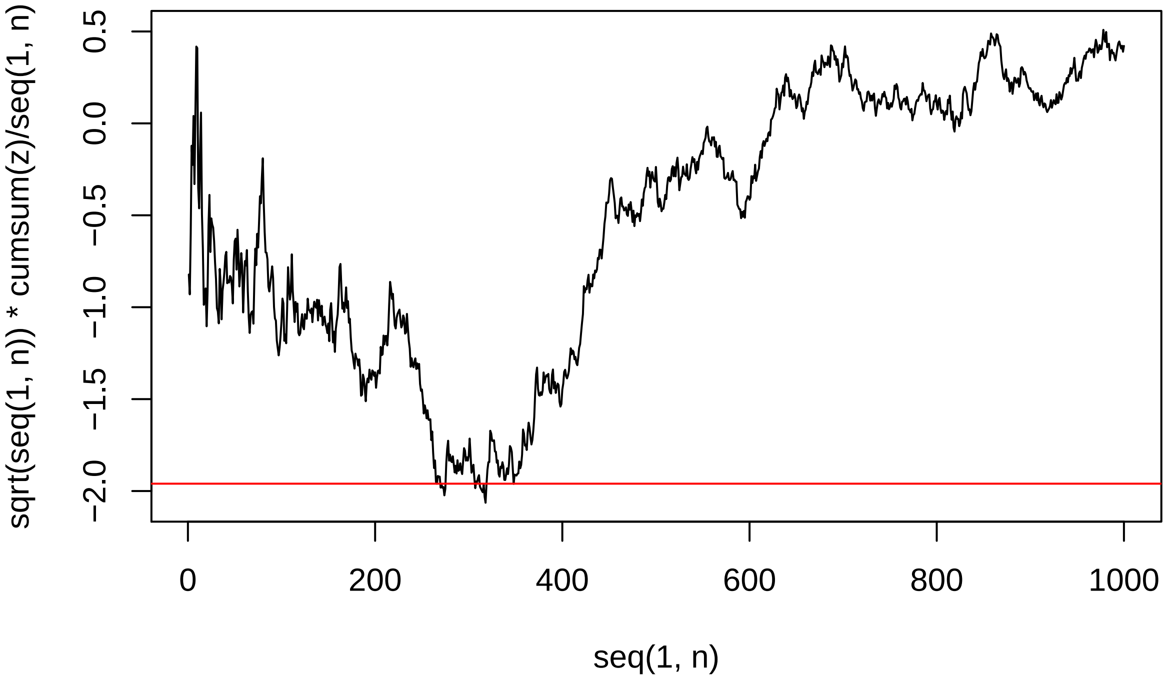
```r
count = 0 # how many times we got type 1 error
c = qnorm(0.975) # pre-specified critical value 1.96
nn = 1000 # number of MC runs

for (i in 1:nn)
{
  n = 1000
  z = rnorm(n, 0, 1)
  test = sqrt(seq(1,n)) * abs(cumsum(z)) / seq(1,n)

  if (max(test) > c) # type 1 error achieved if test stat > c
  {
    count = count + 1
  }
}
count / nn
```

```
## [1] 0.54
```

```r
plot(seq(1,n), sqrt(seq(1,n)) * cumsum(z) / seq(1,n), type='l')
abline(h= c, col='red')
abline(h=-c, col='red')
```



(b)

Is this type 1 probability what I was expecting? No it is not, since we are expecting 5% by using the 1.96 quantile.

It is assumed we are using a two-tailed 5% test since the test statistic includes an absolute value, and our quantile is 1.96.

With this new threshold we get an error of 10%. But removing the absolute value gives an error of 5%. This means that 5% of the time, one of the five tests is above 2.13. I believe this is an error in the question, because the simulation with absolute value gives 5% when we set N=1, as expected. A higher threshold is needed if we use absolute value, like something around 2.4.

```r
do_test = function(c = 1.96)
{
  count = rep(0,5) # count number of occurences at step i
  nn = 1000 # number of MC runs
  n = 5

  for (i in 1:nn)
  {
    z = rnorm(n, 0, 1)
    test = sqrt(seq(1,n)) * abs(cumsum(z)) / seq(1,n)

    hits = which(test > c) # indices of points > c
    if (length(hits) > 0) # type 1 error achieved if some point > c
    {
      idx = hits[1] # index of first element to be > c
      count[idx] = count[idx] + 1
    }
  }
  cat('p1..p5:', count, '\n')
  return(sum(count)/nn)
}
do_test(1.96)
```

```
## p1..p5: 52 30 27 22 20

## [1] 0.151
```

```
do_test(2.13)
```

```
## p1..p5: 30 26 14 19 6

## [1] 0.095
```

```
do_test(2.4)
```

```
## p1..p5: 18 13 8 13 6

## [1] 0.058
```

(c)

Since $.2\sqrt{100} = 2 > 1.96$, we have evidence against H_0 at the 5% level.

(i)

Running our test from part (a), we run 10_000 simulations, each sample size 100. If we check that any of the 100 test statistics are above c, we get type 1 error rate of 37%. But if we only check the 100th test (or any other single index), the error rate is 5%.

Thus, the problem with my analysis for the scientist is that they may have stopped after seeing a good enough sample (since there is a 37% chance of that happening) instead of actually getting a result that carries 5% significance.

If it is true that the scientist has pre-specified the number of samples $n = 100$ before running the experiment, then there is truly a 5% significance in their result. But if they modified n as they go along the experiment, then this result has 37% significance.

This is similar to throwing dice. The probability of getting a 3 is 1 if you roll until you get a 3. Otherwise it is 1/6 if you roll once.

```
count1 = 0
count2 = 0
c = 1.96
nn = 10000 # number of MC runs

for (i in 1:nn)
{
  n = 100
  z = rnorm(n, 0, 1)
  test = sqrt(seq(1,n)) * abs(cumsum(z)) / seq(1,n)

  if (max(test) > c) # any test > c
  {
    count1 = count1 + 1
  }
  if (test[100] > c) # only 100th test > c
  {
    count2 = count2 + 1
  }
}
count1 / nn
```

```
## [1] 0.3673
```

```
count2 / nn
```

```
## [1] 0.0494
```

(ii)

This is a special case of part (b) where we are doing 2 sequential tests. Modifying the simulation and trying values, we see that c = 2.175 seems to give about 5% type 1 error.

What is disturbing in this analysis is that we need to increase c every time the scientist decides to re-run the experiment, after seeing that the previous run was not significant. From part (a) we know this statistic grows without bound so if the scientist really wanted to get a significant result, they can just keep adding samples, and they are guaranteed significance eventually.

```
do_test = function(c = 1.96)
{
  count = rep(0,2)
  nn = 100000 # number of MC runs
  n = 2

  for (i in 1:nn)
  {
    z = rnorm(200, 0, 1)
    z1 = z[seq(1,100)]
    test1 = sqrt(100) * abs(mean(z1))
    test2 = sqrt(200) * abs(mean(z))

    if (test1 > c)
    {
      count[1] = count[1] + 1
    }
    else
    {
      if (test2 > c)
      {
        count[2] = count[2] + 1
      }
    }
  }
  cat('p1..p5:', count, '\n')
  return(sum(count)/nn)
}
do_test(2.175)
```

```
## p1..p5: 2964 2099
## [1] 0.05063
```

To calculate the p-value of this procedure having test < 2.1, we can simulate it. The conditions are that the first 100 samples are below c = 2.175, and then the second statistic is below 2.1.

```
do_test = function(c = 1.96)
{
  p = 0
  nn = 100000 # number of MC runs

  for (i in 1:nn)
  {
    z = rnorm(200, 0, 1)
    z1 = z[seq(1,100)]
    test1 = sqrt(100) * abs(mean(z1))
    test2 = sqrt(200) * abs(mean(z))

    if (test1 > c)
    {
    }
    else
    {
      if (test2 < 2.1)
      {
        p = p + 1
      }
    }
  }
  return(p/nn)
}
do_test(2.175)
```

```
## [1] 0.94524
```

If the full data set of 200 points is published online, then a different scientist looking at it will use a critical value of 1.96 (relative to the standardized variable sqrt(200)xbar_200)

```
pnorm(2.1)
```

```
## [1] 0.9821356
```

The p-value this second scientist will see is 0.98 which is significant at the 5% level, even though in reality it is not due to the sequential procedure.

This result does not agree with what the original scientist did in the previous part because it doesn't take into account that they looked at the data and decided to augment it after not getting significance at a sample size of 100. This makes the second scientist's test statistic invalid.