EAS509 Homework 6 (100 points). Key

Submit your answers as a single pdf attach all R code. Failure to do so will result in grade reduction.

Question 1 (100 points)

High-Performance Computing (HPC) resources (a.k.a. supercomputers) are complex systems. Slight changes in hardware or software can drastically affect their performance. For example, a corrupted lookup table in a network switch, an update of a linux kernel, a drop of hardware support in a new software version, and so on.

One way to ensure the top performance of HPC resources is to utilize continuous performance monitoring where the same application is executed with the same input on a regular basis (for example, daily). In a perfect world, the execution time will be exactly the same, but in reality, it varies due to system jitter (this is partially due to system processes taking resources to do their jobs).

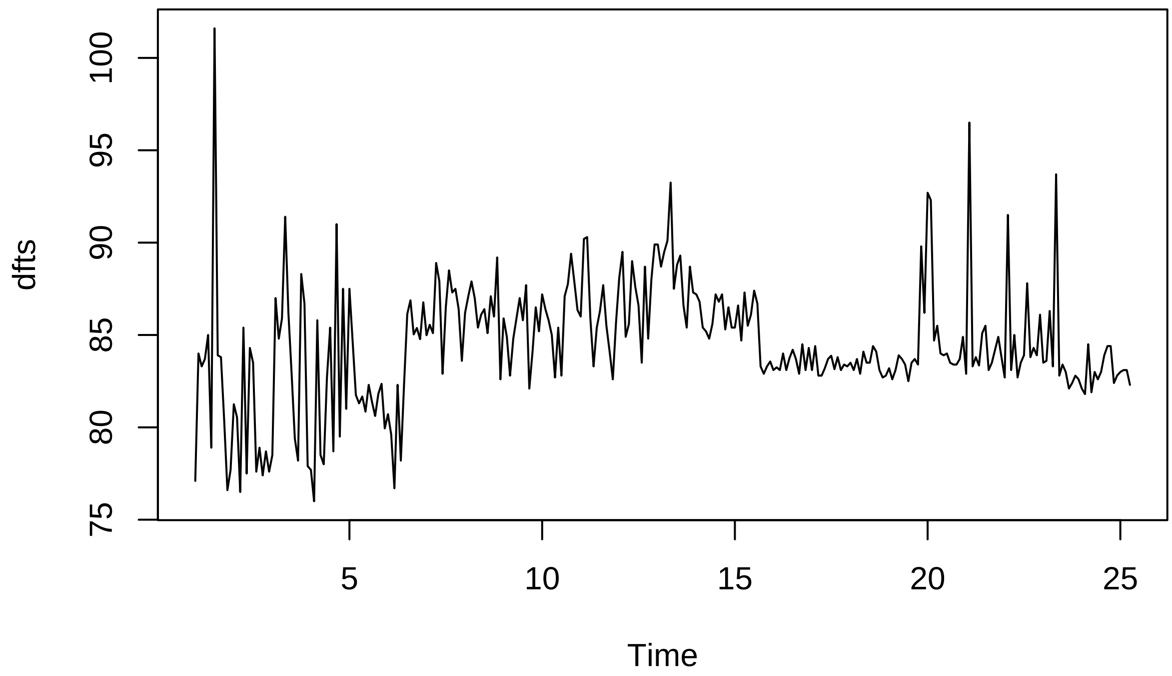
So normally, the execution time will be distributed around a certain value. If performance degradation occurs, the execution time will be distributed around different value.

An automated system that inform system administrators on performance change can be a very handy tool.

In this exercise, your task will be to identify the number and location of the change point where performance was changed. NWChem, an Quantum Chemistry application, was used to probe the performance of UB HPC cluster.

1.1 `UBHPC_8cores_NWChem_Wall_Clock_Time.csv` file contains execution time (same as run time or wall time) of NWChem performing same reference calculation. Read the file and plot it run time on date. (10 points)

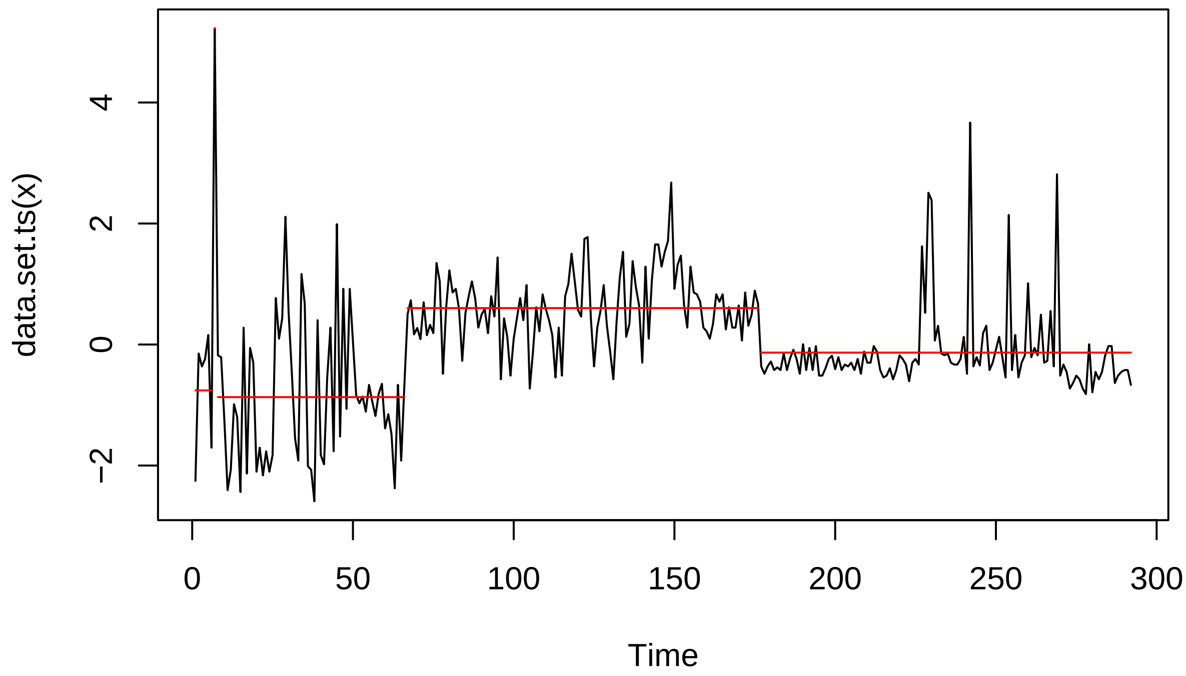
```
df <- read.csv('UBHPC_8cores_NWChem_Wall_Clock_Time.csv')
df$date <- as.POSIXct(df$date, format = "%d/%m/%Y %H:%M")
df$date <- as.Date(df$date)

dfts = ts(df$run_time, frequency = 12)
ts.plot(dfts)
```




```
#mean changepoint analysis
mvalue = cpt.mean(as.vector(scale(dfts)), method='PELT')
cpts(mvalue)
```

```
## [1]   6   7  66 176
```

```
plot(mvalue)
```



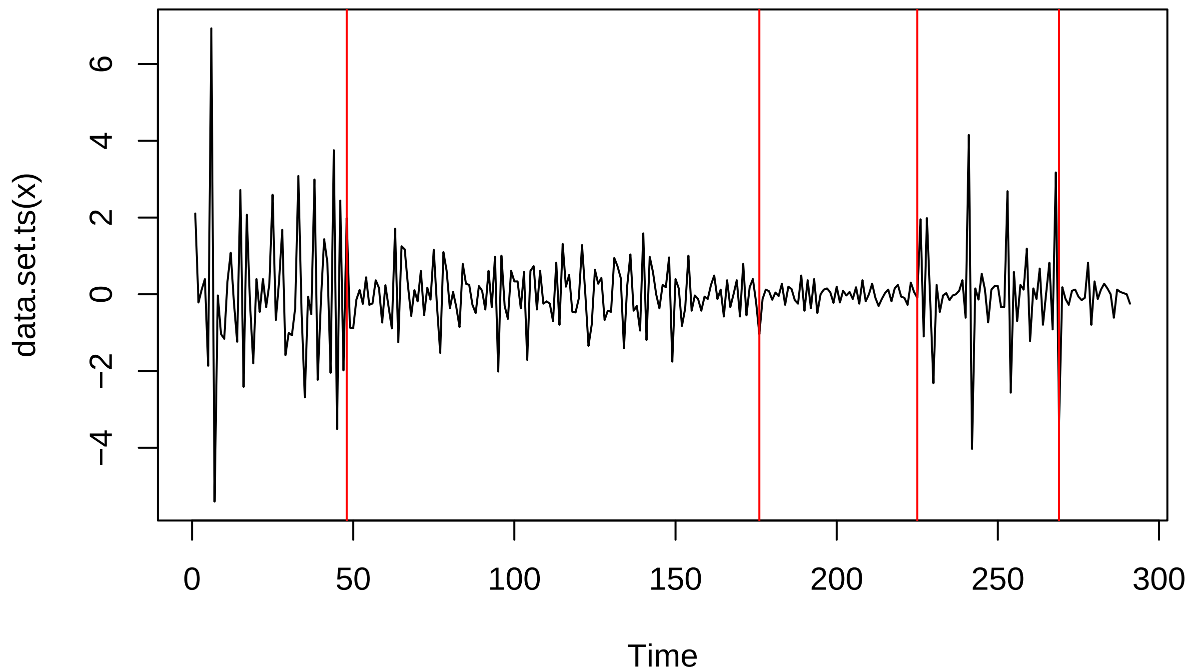

```r
summary(mvalue)
```

```
## Created Using changepoint version 2.2.4
## Changepoint type      : Change in mean
## Method of analysis    : PELT
## Test Statistic  : Normal
## Type of penalty       : MBIC with value, 17.03026
## Minimum Segment Length : 1
## Maximum no. of cpts   : Inf
## Changepoint Locations : 6 7 66 176
```

```r
# variance changepoint analysis
vnvalue = cpt.var(diff(as.vector(scale(dfts))), method='PELT')
cpts(vnvalue)
```

```
## [1]  48 176 225 269
```

```r
plot(vnvalue)
```




```
summary(vnvalue)
```

```
## Created Using changepoint version 2.2.4
## Changepoint type      : Change in variance
## Method of analysis    : PELT
## Test Statistic  : Normal
## Type of penalty       : MBIC with value, 17.01997
## Minimum Segment Length : 2
## Maximum no. of cpts   : Inf
## Changepoint Locations : 48 176 225 269
```

1.2 How many segments/change points can you eyeball? What are they? (10 points)

Usinf cpt mean: 5 changepoints 6, 7, 66, 176 Usinf cpt var: 2 changepoints 48, 176, 225, 269

1.3 Create another column **seg** and assign segment number to it based on previous question. (10 points)

```
seg_data = df %>% mutate(segement = row_number()) %>% mutate(seg = ifelse(row_number() < 6, 1,
                                                                    ifelse(row_number() %in% 6:7, 2
                                                                        ifelse(row_number() %in% 7:(
                                                                            ifelse(row_number() %in%
                                                                                   5)))))

seg_data
```

```
##            date  run_time segement seg
```

```
## 1   2017-09-11  77.10000        1   1
## 2   2017-10-11  84.00000        2   1
## 3   2017-11-11  83.30000        3   1
## 4   2017-12-11  83.70000        4   1
## 5         <NA>  85.00000        5   1
## 6         <NA>  78.90000        6   2
## 7         <NA> 101.60000        7   2
## 8         <NA>  83.90000        8   3
## 9         <NA>  83.80000        9   3
## 10        <NA>  80.40000       10   3
## 11        <NA>  76.60000       11   3
## 12        <NA>  77.70000       12   3
## 13        <NA>  81.25000       13   3
## 14        <NA>  80.55000       14   3
## 15        <NA>  76.50000       15   3
## 16        <NA>  85.40000       16   3
## 17        <NA>  77.50000       17   3
## 18        <NA>  84.30000       18   3
## 19  2017-02-12  83.50000       19   3
## 20  2017-03-12  77.60000       20   3
## 21  2017-04-12  78.90000       21   3
## 22  2017-05-12  77.40000       22   3
## 23  2017-06-12  78.70000       23   3
## 24  2017-07-12  77.60000       24   3
## 25  2017-08-12  78.50000       25   3
## 26  2017-09-12  87.00000       26   3
## 27  2017-11-12  84.80000       27   3
## 28  2017-12-12  85.90000       28   3
## 29        <NA>  91.40000       29   3
## 30        <NA>  86.20000       30   3
## 31        <NA>  82.90000       31   3
## 32        <NA>  79.40000       32   3
## 33        <NA>  78.20000       33   3
## 34        <NA>  88.30000       34   3
## 35        <NA>  86.70000       35   3
## 36        <NA>  77.90000       36   3
## 37        <NA>  77.70000       37   3
## 38        <NA>  76.00000       38   3
## 39        <NA>  85.80000       39   3
## 40        <NA>  78.50000       40   3
## 41        <NA>  78.00000       41   3
## 42        <NA>  82.70000       42   3
## 43        <NA>  85.40000       43   3
## 44        <NA>  78.70000       44   3
## 45        <NA>  91.00000       45   3
## 46  2018-01-01  79.50000       46   3
## 47  2018-03-01  87.50000       47   3
## 48  2018-04-01  81.00000       48   3
## 49  2018-08-01  87.50000       49   3
## 50        <NA>  84.65000       50   3
## 51        <NA>  81.75000       51   3
## 52        <NA>  81.30000       52   3
## 53        <NA>  81.67000       53   3
## 54        <NA>  80.85000       54   3
```

```
## 55  2018-02-02  82.30000       55   3
## 56  2018-03-02  81.40000       56   3
## 57  2018-04-02  80.61667       57   3
## 58  2018-05-02  81.81111       58   3
## 59  2018-06-02  82.35714       59   3
## 60  2018-07-02  79.94286       60   3
## 61  2018-08-02  80.71429       61   3
## 62  2018-09-02  79.61667       62   3
## 63  2018-10-02  76.70000       63   3
## 64  2018-11-02  82.30000       64   3
## 65        <NA>  78.20000       65   3
## 66        <NA>  82.30000       66   3
## 67        <NA>  86.13333       67   4
## 68        <NA>  86.87778       68   4
## 69        <NA>  85.03333       69   4
## 70        <NA>  85.38000       70   4
## 71        <NA>  84.77500       71   4
## 72        <NA>  86.77000       72   4
## 73        <NA>  84.98750       73   4
## 74        <NA>  85.55000       74   4
## 75        <NA>  85.10000       75   4
## 76        <NA>  88.90000       76   4
## 77        <NA>  87.90000       77   4
## 78        <NA>  82.90000       78   4
## 79  2018-02-03  86.50000       79   4
## 80  2018-03-03  88.50000       80   4
## 81  2018-04-03  87.30000       81   4
## 82  2018-05-03  87.50000       82   4
## 83  2018-06-03  86.40000       83   4
## 84  2018-07-03  83.60000       84   4
## 85  2018-08-03  86.20000       85   4
## 86  2018-09-03  87.10000       86   4
## 87  2018-10-03  87.90000       87   4
## 88  2018-11-03  87.00000       88   4
## 89  2018-12-03  85.40000       89   4
## 90        <NA>  86.10000       90   4
## 91        <NA>  86.40000       91   4
## 92        <NA>  85.10000       92   4
## 93        <NA>  87.10000       93   4
## 94        <NA>  86.00000       94   4
## 95        <NA>  89.20000       95   4
## 96        <NA>  82.60000       96   4
## 97        <NA>  85.90000       97   4
## 98        <NA>  84.90000       98   4
## 99        <NA>  82.80000       99   4
## 100       <NA>  84.80000      100   4
## 101       <NA>  85.90000      101   4
## 102       <NA>  87.00000      102   4
## 103       <NA>  85.80000      103   4
## 104       <NA>  87.70000      104   4
## 105       <NA>  82.10000      105   4
## 106       <NA>  84.10000      106   4
## 107       <NA>  86.50000      107   4
## 108 2018-02-04  85.20000      108   4
```

```
## 109 2018-03-04  87.20000       109   4
## 110 2018-04-04  86.40000       110   4
## 111 2018-05-04  85.80000       111   4
## 112 2018-06-04  85.00000       112   4
## 113 2018-07-04  82.70000       113   4
## 114 2018-09-04  85.40000       114   4
## 115 2018-10-04  82.80000       115   4
## 116 2018-12-04  87.10000       116   4
## 117       <NA>  87.75000       117   4
## 118       <NA>  89.40000       118   4
## 119       <NA>  87.90000       119   4
## 120       <NA>  86.35000       120   4
## 121       <NA>  86.00000       121   4
## 122       <NA>  90.20000       122   4
## 123       <NA>  90.30000       123   4
## 124       <NA>  85.90000       124   4
## 125       <NA>  83.30000       125   4
## 126       <NA>  85.40000       126   4
## 127       <NA>  86.30000       127   4
## 128       <NA>  87.70000       128   4
## 129       <NA>  85.50000       129   4
## 130 2018-02-05  84.10000       130   4
## 131 2018-03-05  82.60000       131   4
## 132 2018-05-05  85.70000       132   4
## 133 2018-06-05  88.10000       133   4
## 134 2018-07-05  89.50000       134   4
## 135 2018-08-05  84.90000       135   4
## 136 2018-09-05  85.60000       136   4
## 137 2018-10-05  89.00000       137   4
## 138 2018-11-05  87.60000       138   4
## 139       <NA>  86.60000       139   4
## 140       <NA>  83.50000       140   4
## 141       <NA>  88.70000       141   4
## 142       <NA>  84.80000       142   4
## 143       <NA>  88.00000       143   4
## 144       <NA>  89.90000       144   4
## 145       <NA>  89.90000       145   4
## 146       <NA>  88.70000       146   4
## 147       <NA>  89.50000       147   4
## 148       <NA>  90.10000       148   4
## 149       <NA>  93.25000       149   4
## 150       <NA>  87.50000       150   4
## 151       <NA>  88.80000       151   4
## 152       <NA>  89.30000       152   4
## 153       <NA>  86.60000       153   4
## 154       <NA>  85.40000       154   4
## 155 2018-01-06  88.70000       155   4
## 156 2018-02-06  87.30000       156   4
## 157 2018-04-06  87.20000       157   4
## 158 2018-05-06  86.80000       158   4
## 159 2018-06-06  85.40000       159   4
## 160 2018-07-06  85.20000       160   4
## 161 2018-08-06  84.80000       161   4
## 162 2018-09-06  85.60000       162   4
```

```
## 163 2018-10-06  87.20000      163   4
## 164 2018-11-06  86.80000      164   4
## 165       <NA>  87.20000      165   4
## 166       <NA>  85.30000      166   4
## 167       <NA>  86.50000      167   4
## 168       <NA>  85.40000      168   4
## 169       <NA>  85.40000      169   4
## 170       <NA>  86.60000      170   4
## 171       <NA>  84.70000      171   4
## 172       <NA>  87.30000      172   4
## 173       <NA>  85.50000      173   4
## 174       <NA>  86.10000      174   4
## 175       <NA>  87.40000      175   4
## 176       <NA>  86.70000      176   4
## 177       <NA>  83.30000      177   5
## 178       <NA>  82.90000      178   5
## 179 2018-02-07  83.30000      179   5
## 180 2018-04-07  83.56667      180   5
## 181 2018-05-07  83.10000      181   5
## 182 2018-06-07  83.25000      182   5
## 183 2018-07-07  83.10000      183   5
## 184 2018-09-07  84.00000      184   5
## 185 2018-11-07  83.10000      185   5
## 186 2018-12-07  83.75000      186   5
## 187       <NA>  84.20000      187   5
## 188       <NA>  83.70000      188   5
## 189       <NA>  82.90000      189   5
## 190       <NA>  84.50000      190   5
## 191       <NA>  83.10000      191   5
## 192       <NA>  84.30000      192   5
## 193       <NA>  83.10000      193   5
## 194       <NA>  84.40000      194   5
## 195       <NA>  82.80000      195   5
## 196       <NA>  82.80000      196   5
## 197       <NA>  83.20000      197   5
## 198       <NA>  83.70000      198   5
## 199       <NA>  83.87778      199   5
## 200       <NA>  83.15000      200   5
## 201       <NA>  83.80000      201   5
## 202       <NA>  83.10000      202   5
## 203 2018-01-08  83.40000      203   5
## 204 2018-02-08  83.30000      204   5
## 205 2018-03-08  83.50000      205   5
## 206 2018-04-08  83.10000      206   5
## 207 2018-06-08  83.70000      207   5
## 208 2018-07-08  82.90000      208   5
## 209 2018-08-08  84.10000      209   5
## 210 2018-09-08  83.50000      210   5
## 211 2018-10-08  83.50000      211   5
## 212 2018-11-08  84.40000      212   5
## 213 2018-12-08  84.10000      213   5
## 214       <NA>  83.10000      214   5
## 215       <NA>  82.70000      215   5
## 216       <NA>  82.80000      216   5
```

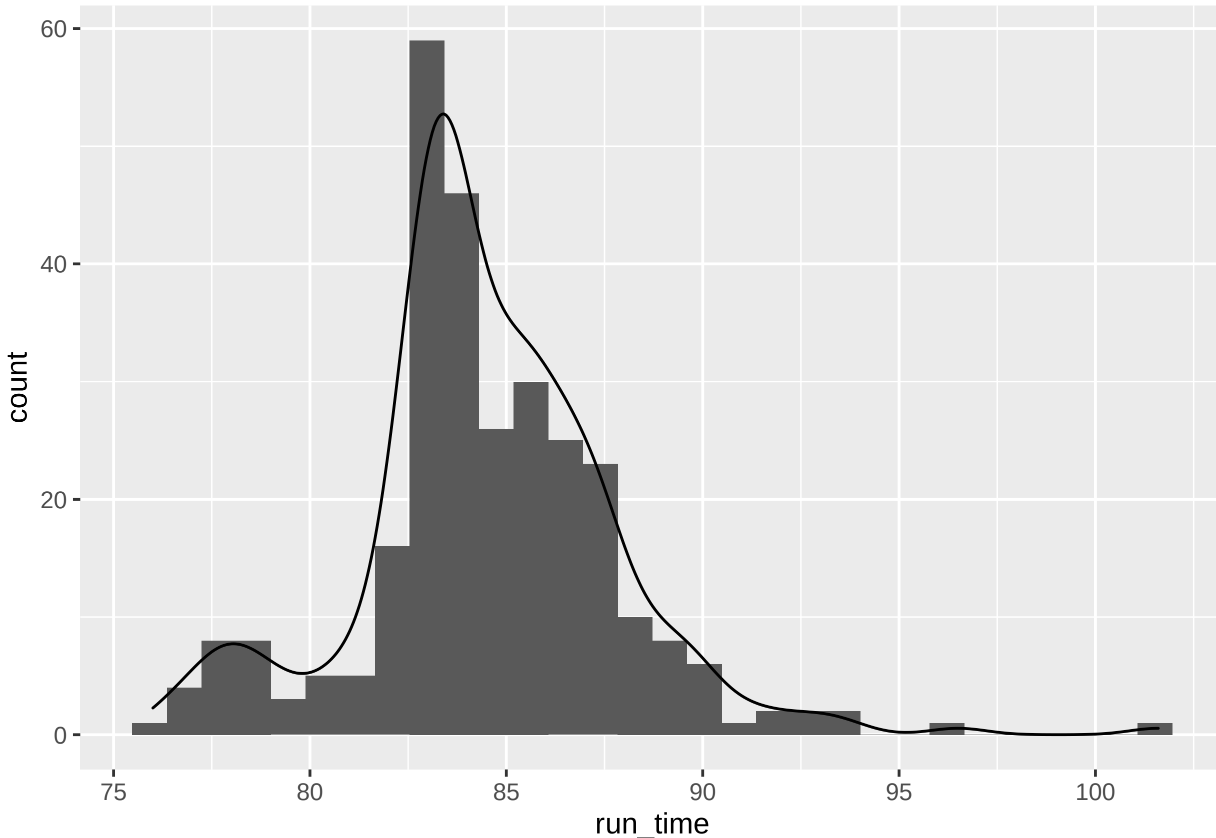
```
## 217       <NA>  83.20000      217   5
## 218       <NA>  82.60000      218   5
## 219       <NA>  83.10000      219   5
## 220       <NA>  83.90000      220   5
## 221       <NA>  83.70000      221   5
## 222       <NA>  83.40000      222   5
## 223       <NA>  82.50000      223   5
## 224       <NA>  83.50000      224   5
## 225       <NA>  83.70000      225   5
## 226       <NA>  83.40000      226   5
## 227       <NA>  89.80000      227   5
## 228       <NA>  86.20000      228   5
## 229       <NA>  92.70000      229   5
## 230       <NA>  92.30000      230   5
## 231       <NA>  84.70000      231   5
## 232       <NA>  85.50000      232   5
## 233 2018-01-09  84.00000      233   5
## 234 2018-02-09  83.90000      234   5
## 235 2018-03-09  84.00000      235   5
## 236 2018-04-09  83.50000      236   5
## 237 2018-05-09  83.40000      237   5
## 238 2018-06-09  83.40000      238   5
## 239 2018-07-09  83.70000      239   5
## 240 2018-09-09  84.90000      240   5
## 241 2018-11-09  82.90000      241   5
## 242       <NA>  96.50000      242   5
## 243       <NA>  83.30000      243   5
## 244       <NA>  83.80000      244   5
## 245       <NA>  83.35000      245   5
## 246       <NA>  85.10000      246   5
## 247       <NA>  85.50000      247   5
## 248       <NA>  83.10000      248   5
## 249       <NA>  83.50000      249   5
## 250       <NA>  84.20000      250   5
## 251       <NA>  84.90000      251   5
## 252       <NA>  83.80000      252   5
## 253       <NA>  82.70000      253   5
## 254       <NA>  91.50000      254   5
## 255       <NA>  83.10000      255   5
## 256 2018-01-10  85.00000      256   5
## 257 2018-02-10  82.70000      257   5
## 258 2018-03-10  83.50000      258   5
## 259 2018-04-10  83.90000      259   5
## 260 2018-05-10  87.80000      260   5
## 261 2018-06-10  83.80000      261   5
## 262 2018-07-10  84.30000      262   5
## 263 2018-09-10  83.90000      263   5
## 264 2018-10-10  86.10000      264   5
## 265 2018-11-10  83.50000      265   5
## 266 2018-12-10  83.60000      266   5
## 267       <NA>  86.30000      267   5
## 268       <NA>  83.30000      268   5
## 269       <NA>  93.70000      269   5
## 270       <NA>  82.80000      270   5
```

```
## 271       <NA>  83.40000      271   5
## 272       <NA>  83.00000      272   5
## 273       <NA>  82.10000      273   5
## 274       <NA>  82.40000      274   5
## 275       <NA>  82.80000      275   5
## 276       <NA>  82.60000      276   5
## 277       <NA>  82.10000      277   5
## 278       <NA>  81.80000      278   5
## 279       <NA>  84.50000      279   5
## 280       <NA>  81.90000      280   5
## 281       <NA>  83.00000      281   5
## 282       <NA>  82.60000      282   5
## 283       <NA>  83.00000      283   5
## 284       <NA>  83.90000      284   5
## 285       <NA>  84.40000      285   5
## 286 2018-01-11  84.40000      286   5
## 287 2018-02-11  82.40000      287   5
## 288 2018-03-11  82.80000      288   5
## 289 2018-04-11  83.00000      289   5
## 290 2018-05-11  83.10000      290   5
## 291 2018-06-11  83.10000      291   5
## 292 2018-07-11  82.30000      292   5
```

1.4 Make a histogramm plot of all run times. (10 points)

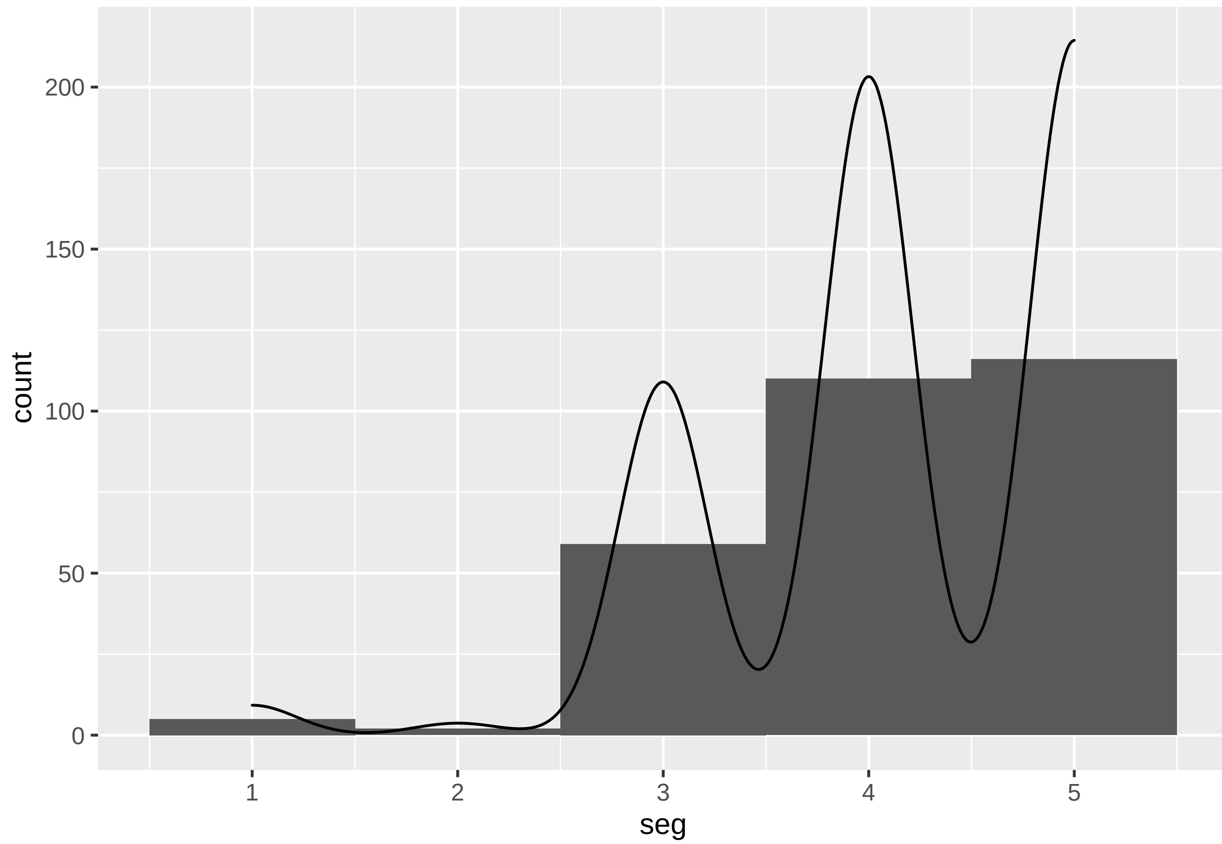
```
ggplot(seg_data, aes(x =run_time)) + geom_histogram()+ geom_density(aes(y=..count..))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



1.5 Make a histogram plot of for each segments. (10 points)

```
ggplot(seg_data, aes(x=seg))+geom_histogram(binwidth=1)+geom_density(aes(y=..count..))
```



1.6 Does it look reasonably normal? (10 points)

From figure 1.4, we can tell that it is a right skewed but a reasonably normal curve.

1.7 Identify change points with `cpt.meanvar` function. Use `PELT` method and `Normal` for `test.stat`. Plot your data with identified segments mean. (10 points)

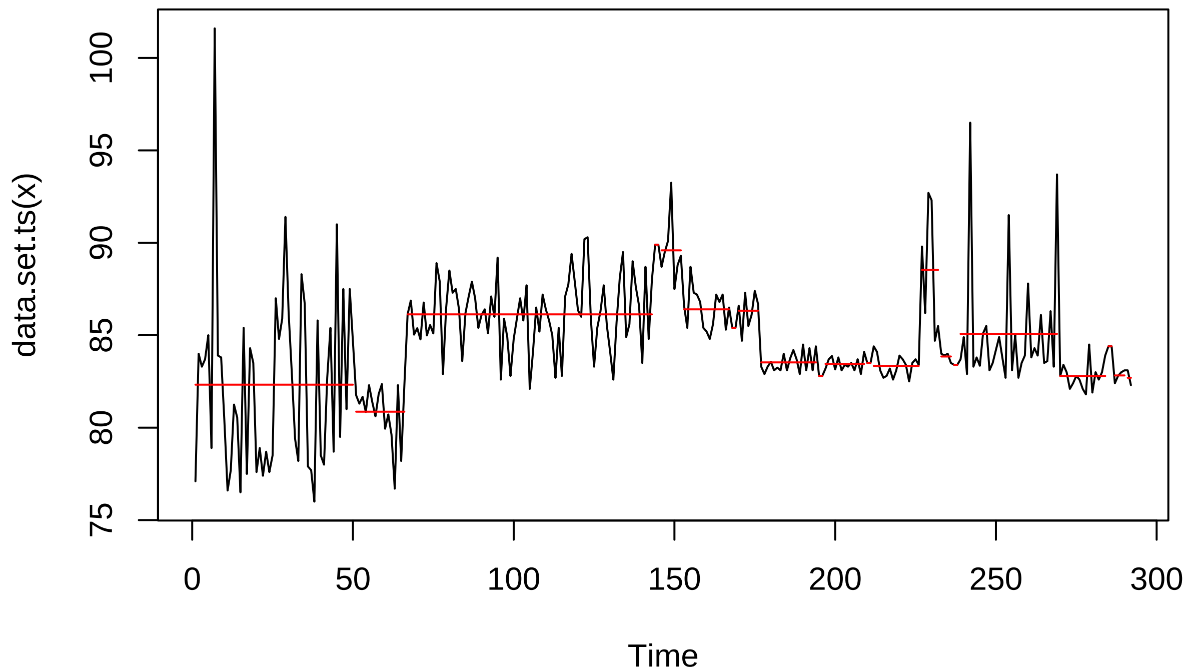
```
set.seed(49)
df_meanvar = cpt.meanvar(df$run_time, test.stat='Normal', method='PELT',penalty = 'SIC')

cpts(df_meanvar)
```

```
##  [1]  50  66 143 145 152 167 169 176 194 196 209 211 226 232 236 238 269 284 286
## [20] 290
```

```
pen.value(df_meanvar)
```

```
## [1] 17.03026
```

```
plot(df_meanvar)
```



hints: run `cpt.meanvar` on the `run_time` column (i.e. `df$run_time`)

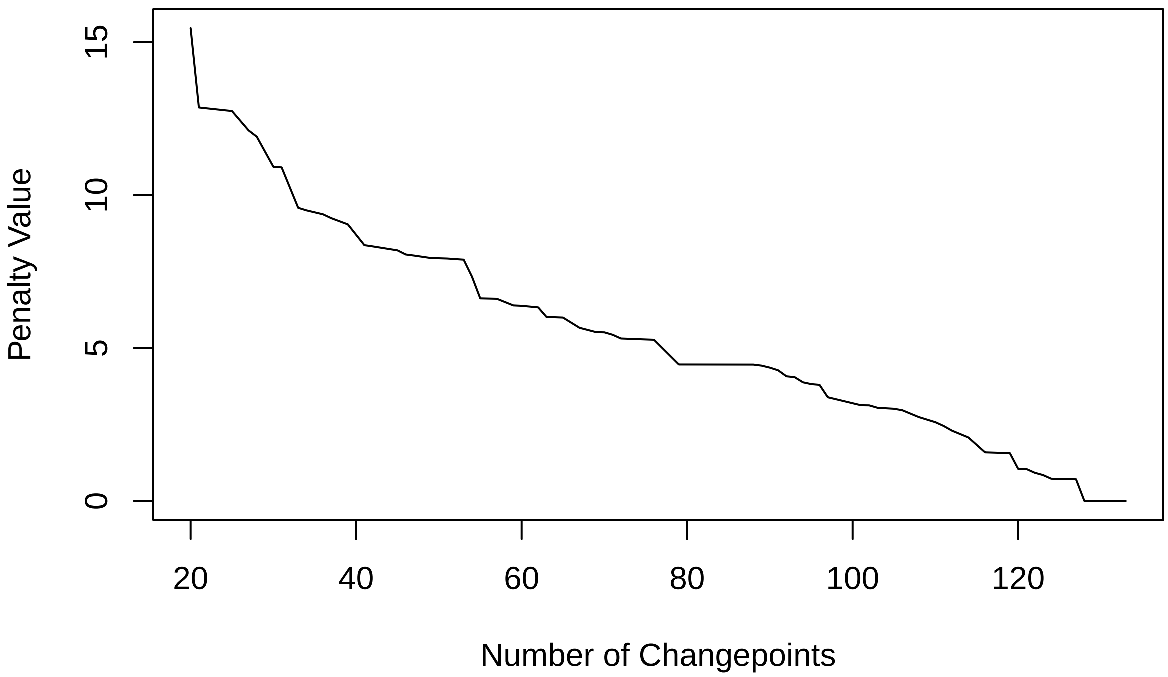
use `pen.value` funtion to see current value of penalty (MBIC value), use that value as guide for your penalty range in next question.

1.8 Using CROPS procedure find optimal number of points. Plot data with optimal number of segments. (10 points)

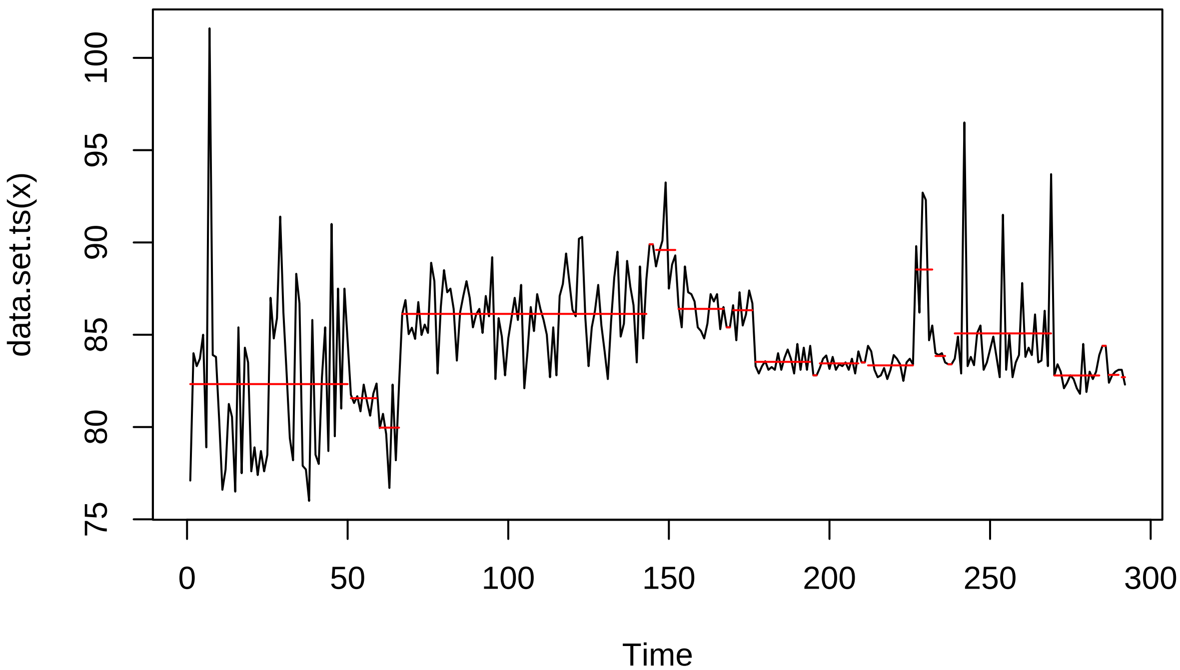
```
df_meanvar_crops = cpt.meanvar(df$run_time, method="PELT", penalty="CROPS",
                  pen.value=c(0, pen.value(df_meanvar)))
```

```
## [1] "Maximum number of runs of algorithm = 115"
## [1] "Completed runs = 2"
## [1] "Completed runs = 3"
## [1] "Completed runs = 5"
## [1] "Completed runs = 9"
## [1] "Completed runs = 17"
## [1] "Completed runs = 33"
## [1] "Completed runs = 59"
## [1] "Completed runs = 85"
## [1] "Completed runs = 93"
## [1] "Completed runs = 95"
```

```
plot(df_meanvar_crops, diagnostic=TRUE)
```




```
plot(df_meanvar_crops,ncpts=21)
```



1.9 Does your initial segment guess matches with optimized by CROPS? (10 points)

our guess was 5 with mean and 20 using mean var. Our guess matches with the one optimized with CROPS

1.10 The run-time in this example does not really follow normal distribution. What to do you think can we still use this method to identify changepoints? (10 points)

mean, var and meanvar assumes that the data is normalized. We need to normalize the data by differencing it such that the mean fits over x axis(mean =0) and the variance is 1.

PS. Just in case if you wounder. On 2018-02-21 system got a critical linux kernel update to alleviate Meltdown-Spectre vulnerabilities. On 2018-06-28 system got another kernel update which is more robust and hit the performance less