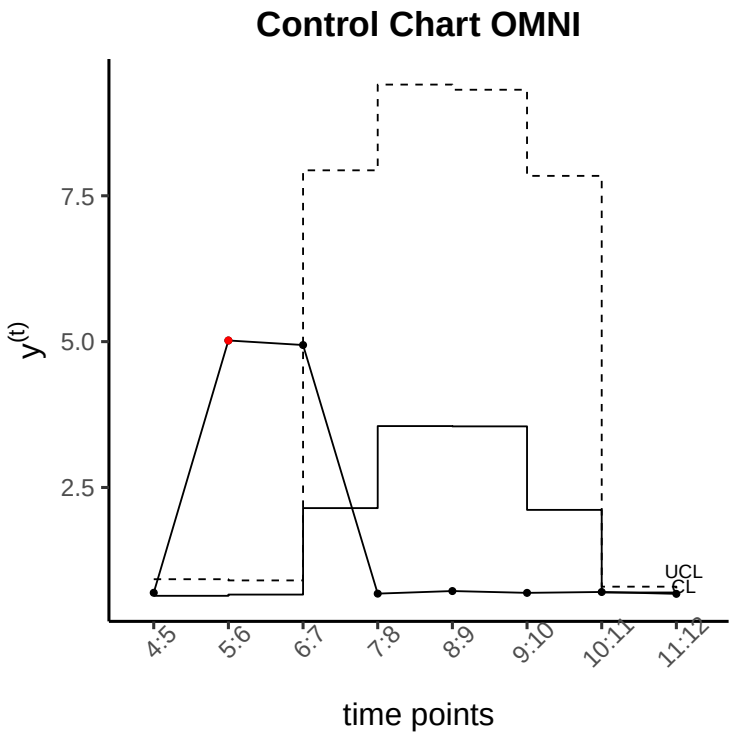
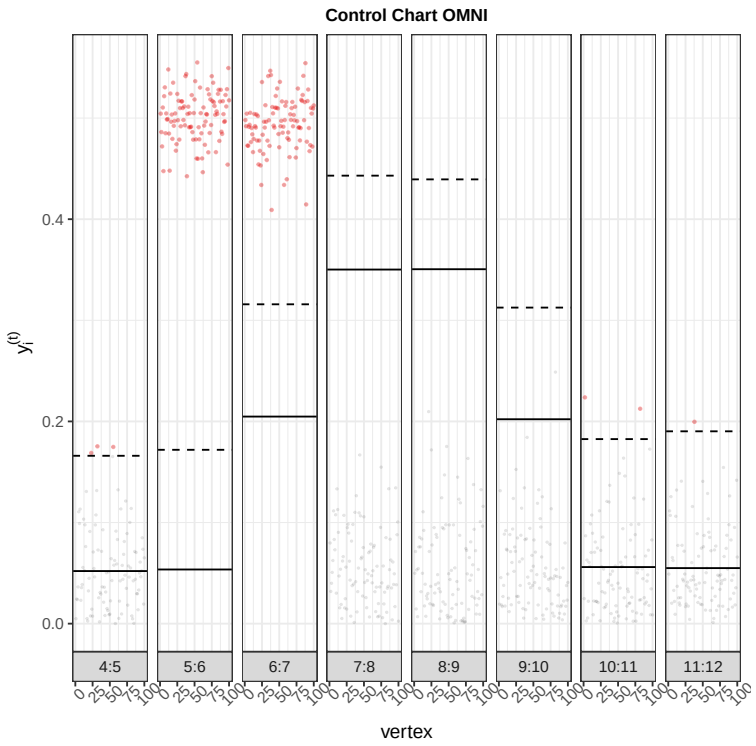
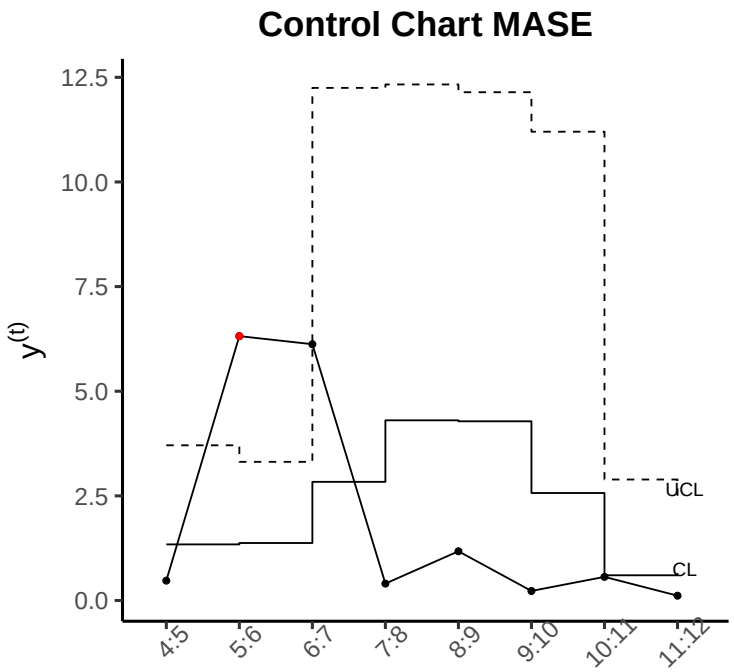
The results below are generated from an R script.

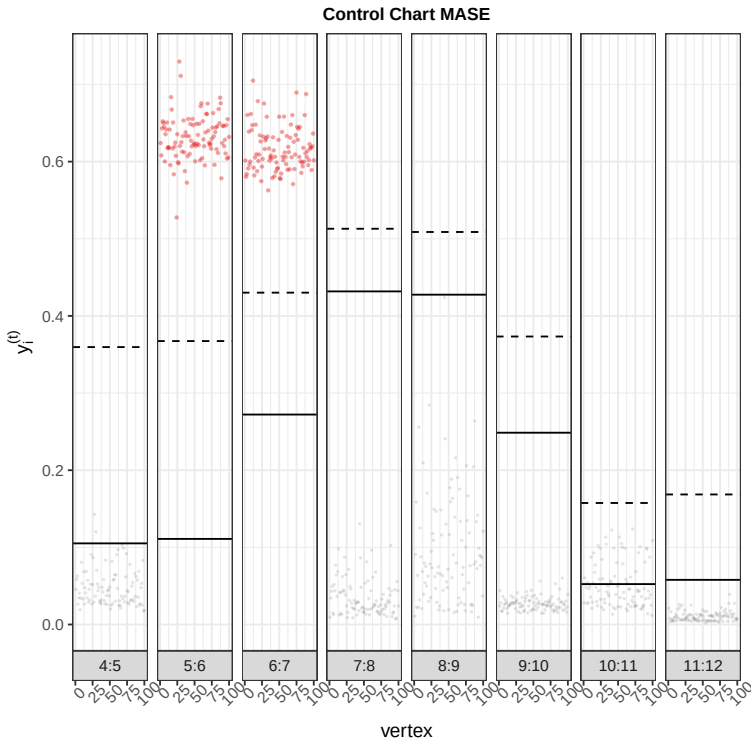
```
#source("utilAD.R")
#source("simulation.R")
#Example 1
#Generate a time series of ER graphs of length 12,
#create graph anomaly at time point 6
n <- 100
glist <- list()
for (i in 1:5) {
  glist[[i]] <- sample_gnp(n,.1)
  }
  glist[[6]] <- sample_gnp(n,.9)
  glist[[7]] <- sample_gnp(n,.1)
  for (i in 8:12) {
    glist[[i]] <- sample_gnp(n,.1)
  }
# Do anomaly detection with OMNI, provide the quantitative control chart for GraphAD and VertexAD
result.OMNI<- qccAD(glist, l=4,d=1,dsvd=NULL,method="OMNI",
 diag.augment = TRUE, approx=FALSE, par=FALSE, numpar=2)

## Warning:  'switch' is deprecated.
## Use 'strip.position' instead.
## See help("Deprecated")
```






```
#print the number of deviation for GraphAD, only positive ones are meaningful
print(result.OMNI$GraphAD)

##      Samples
## Group       [,1]
##     1  0.5524253
##     2 53.7757766
##     3  1.4484721
##     4 -1.4717813
##     5 -1.4676294
##     6 -0.7451977
##     7  0.2686479
##     8 -1.5741267

# Do anomaly detection with MASE
result.MASE<- qccAD(glist, l=4,d=2,dsvd=2,method="MASE",
               diag.augment = TRUE, approx=FALSE, par=FALSE, numpar=2)

## Warning:  'switch' is deprecated.
## Use 'strip.position' instead.
## See help("Deprecated")
```

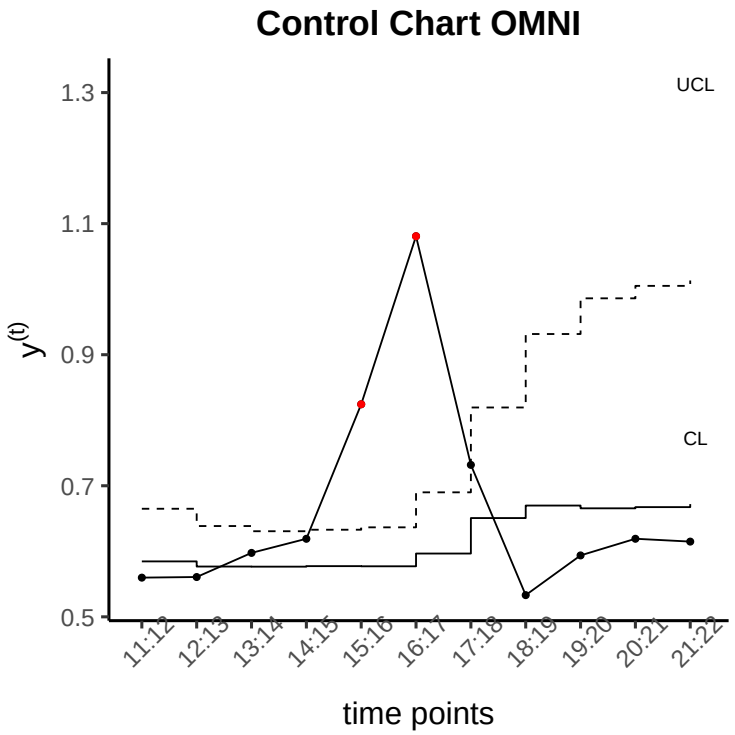


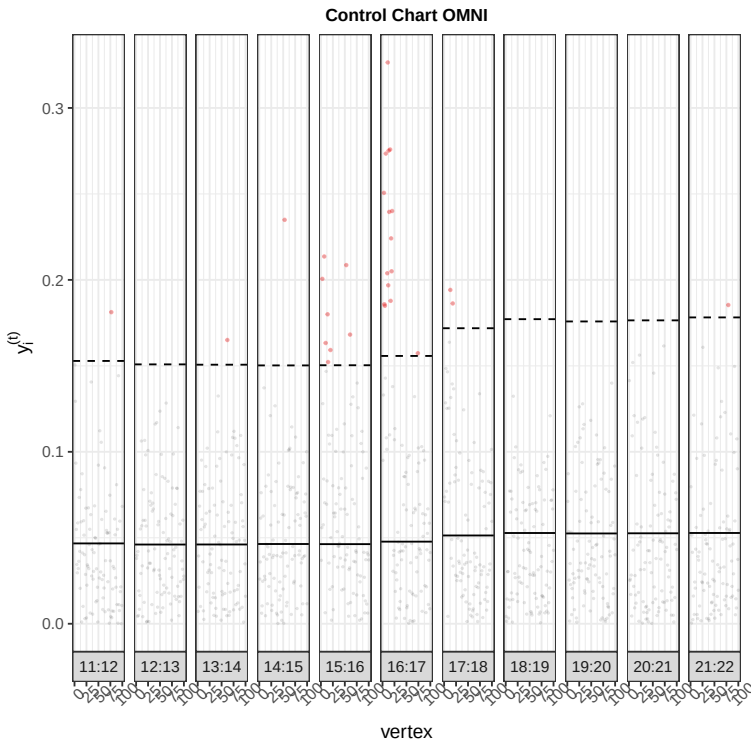



```
#print the number of deviation for GraphAD, only positive ones are meaningful
print(result.MASE$GraphAD)

##      Samples
## Group        [,1]
##     1 -1.09610988
```

```
##      2  7.64596167
##      3  1.04816905
##      4 -1.45843192
##      5 -1.18478469
##      6 -0.81322605
##      7 -0.04974224
##      8 -0.94957297
```

```
#Example 2
# Sample a time series of RDPG graph (length tmax > 17) with same 1-1 matched vertices unweighted
# hollow symmetric undirected graphs, the latent positions i.i.d uniform.
# Some vertices in 16-th and 17-th graphs are given perturbations so there exists anomalies at 16:17.
n <- 100 #number of vertices
nperturb <- 20 #number of perturbed vertices
cperturb <- .12 #number of perturbation, larger cperturb means more obvious anomalies.
rmin <- .2 # parameter for uniform[rmin, rmax].
rmax <- .8 # parameter for uniform[rmin, rmax].
tmax <- 22 # number of graphs must be greater than 17.
#Generate data or load the data you want
glist <- generate.tsg(n, nperturb, cperturb=NULL, rmin, rmax, tmax)$glist

#Do anomaly detection with OMNI in parallel
result.OMNI <- qccAD(glist, l=11,d=1,dsvd=NULL,method="OMNI",
                     diag.augment = TRUE, approx=FALSE, par=TRUE, numpar=2)

## Warning:  'switch' is deprecated.
## Use 'strip.position' instead.
## See help("Deprecated")
```



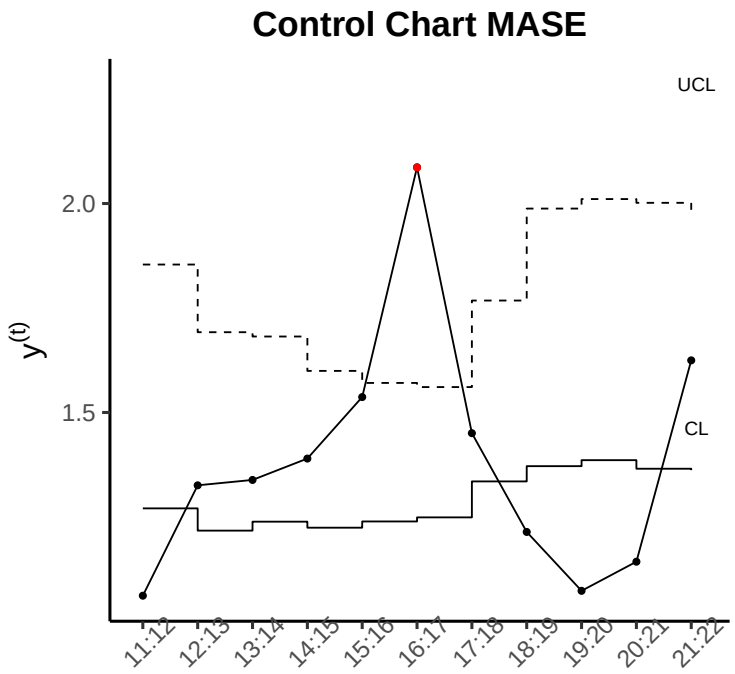


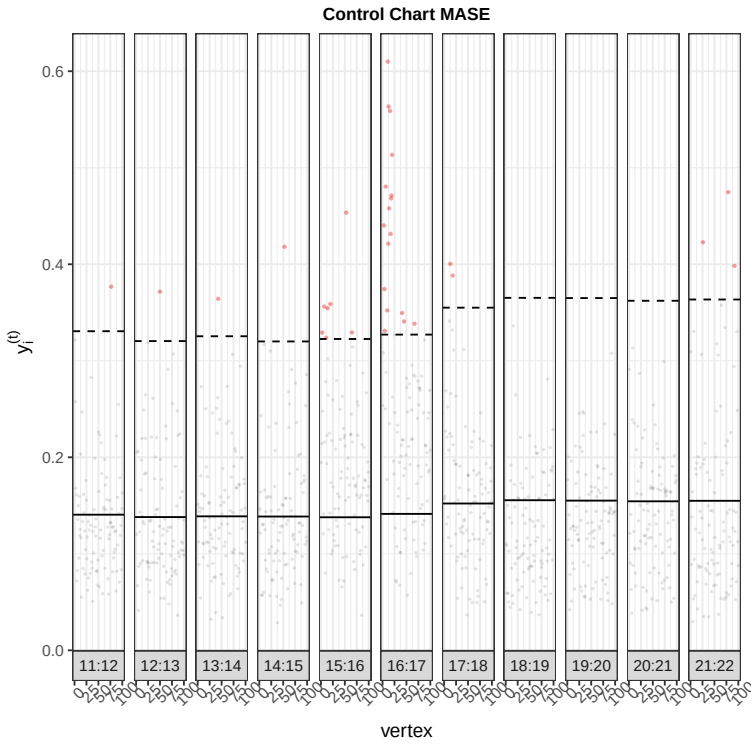

```
#print the number of deviation for GraphAD, only positive ones are meaningful
print(result.OMNI$GraphAD)
```

```
##      Samples
## Group       [,1]
##     1  -0.9260126
##     2  -0.7771847
##     3   1.1667894
##     4   2.2546748
##     5  12.4817008
##     6  15.5529279
##     7   1.4432577
##     8  -1.5661622
##     9  -0.6726323
##     10 -0.4290760
##     11 -0.5030972
```

```
# Do anomaly detection with MASE in parallel
result.MASE<- qccAD(glist, l=11,d=1,dsvd=2,method="MASE",
                    diag.augment = TRUE, approx=FALSE, par=TRUE, numpar=2)
```

```
## Warning:  'switch' is deprecated.
## Use 'strip.position' instead.
## See help("Deprecated")
```

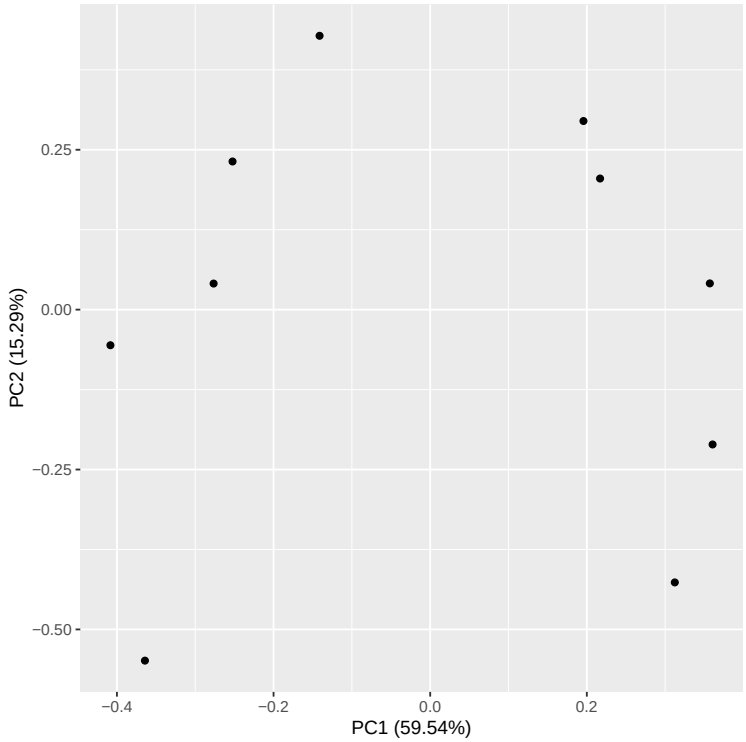




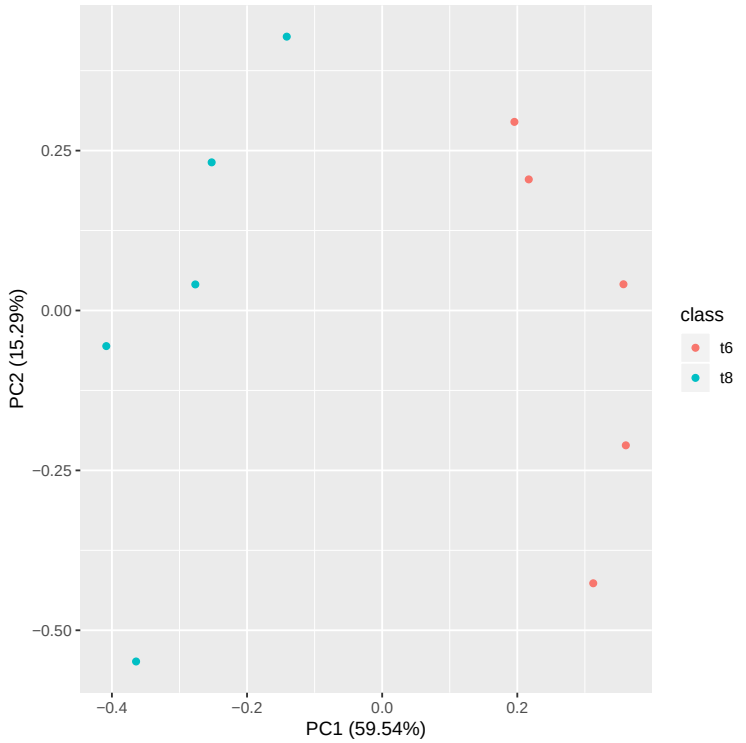

```
#print the number of deviation for GraphAD, only positive ones are meaningful
print(result.MASE$GraphAD)

##      Samples
## Group       [,1]
##     1  -1.0741138
```

```
##    2   0.6851632
##    3   0.6762691
##    4   1.3225986
##    5   2.6942888
##    6   8.0595982
##    7   0.8002741
##    8  -0.7659120
##    9  -1.5011044
##    10 -1.0487262
##    11  1.2654026
```

```
#Example 3
#five of ER tsg with change point at t=6 and five at t=8.
n <- 100
dat <- matrix(0, 10, 8)
for (j in 1:5) {
  glist <- list()
  for (i in 1:5) {
    glist[[i]] <- sample_gnp(n,.1)
  }
  glist[[6]] <- sample_gnp(n,.9)
  for (i in 7:12) {
    glist[[i]] <- sample_gnp(n,.1)
  }
  # Do anomaly detection with OMNI, provide the quantitative control chart for GraphAD and VertexAD
  result.OMNI<- qccAD(glist, l=4,d=1,dsvd=NULL,method="OMNI",
                      diag.augment = TRUE, approx=FALSE, par=FALSE, numpar=2, plot.figure = FALSE)
  dat[j,] <- result.OMNI$GraphAD
}
for (j in 6:10) {
  glist <- list()
  for (i in 1:7) {
    glist[[i]] <- sample_gnp(n,.1)
  }
  glist[[8]] <- sample_gnp(n,.9)
  for (i in 9:12) {
    glist[[i]] <- sample_gnp(n,.1)
  }
  # Do anomaly detection with OMNI, provide the quantitative control chart for GraphAD and VertexAD
  result.OMNI<- qccAD(glist, l=4,d=1,dsvd=NULL,method="OMNI",
                      diag.augment = TRUE, approx=FALSE, par=FALSE, numpar=2, plot.figure = FALSE)
  dat[j,] <- result.OMNI$GraphAD
}
df <- data.frame(dat,class=factor(c(rep("t6",5),rep("t8",5))))
pca_res <- prcomp(dat, scale. = TRUE)
library(ggfortify)
autoplot(pca_res)
```




```
autoplot(pca_res, data = df,colour="class")
```



The R session information (including the OS info, R version and all packages used):

```
sessionInfo()

## R version 3.6.1 (2019-07-05)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Sierra 10.12.3
##
## Matrix products: default
## BLAS:   /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versio
## LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] parallel  stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] AnomalyDetection_0.1.0 rARPACK_0.11-0         igraph_1.2.4.1
##  [4] dplyr_0.8.3            latex2exp_0.4.0        gtools_3.8.1
##  [7] irlba_2.3.3            Matrix_1.2-17          doParallel_1.0.15
## [10] iterators_1.0.12       ggfortify_0.4.10       ggplot2_3.2.1
## [13] qcc_2.7                foreach_1.4.7          knitr_1.24
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.2       RSpectra_0.16-0  highr_0.8        pillar_1.4.2     compiler_3.6.1
##  [6] tools_3.6.1      digest_0.6.20    evaluate_0.14    lattice_0.20-38  tibble_2.1.3
## [11] lifecycle_0.1.0  gtable_0.3.0     pkgconfig_2.0.2  rlang_0.4.5      rstudioapi_0.10
## [16] xfun_0.9         gridExtra_2.3    withr_2.1.2      stringr_1.4.0    vctrs_0.2.4
## [21] grid_3.6.1       tidyselect_0.2.5 glue_1.3.1       R6_2.4.0         purrr_0.3.3
## [26] tidyr_1.0.0      magrittr_1.5     scales_1.0.0     codetools_0.2-16 MASS_7.3-51.4
## [31] assertthat_0.2.1 colorspace_1.4-1 labeling_0.3     stringi_1.4.3    lazyeval_0.2.2
## [36] munsell_0.5.0    crayon_1.3.4

Sys.time()

## [1] "2020-08-13 18:59:49 EDT"
```