Clustering

Sayani Gupta

16/06/2021

Contents

1 How the time series plot of the designs look like for different sample sizes?

```
set.seed(9999)
nx_val = 2 # number of x-axis levels
nfacet_val = 3 # number of facet levels
w1_val = 3 # increment in mean
w2_val = 0 # increment in sd
mean_val = 0 # mean of normal distribution of starting combination
sd_val = 1 # sd of normal distribution of starting combination
```

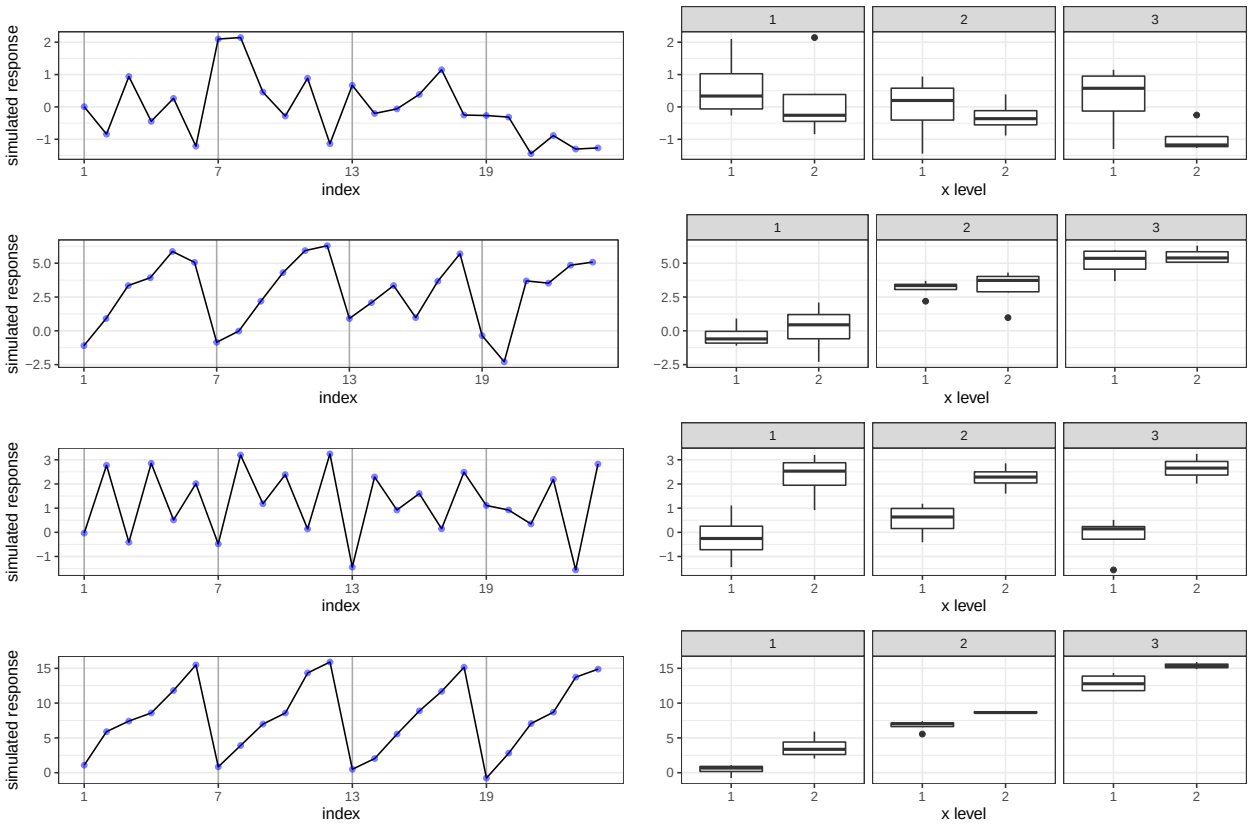
1.1 Number of obervations per combination is 4

```
ntimes_val = 4
```

Table 1: Simulated data across combination of categories (left) and manipulated time series format (right)

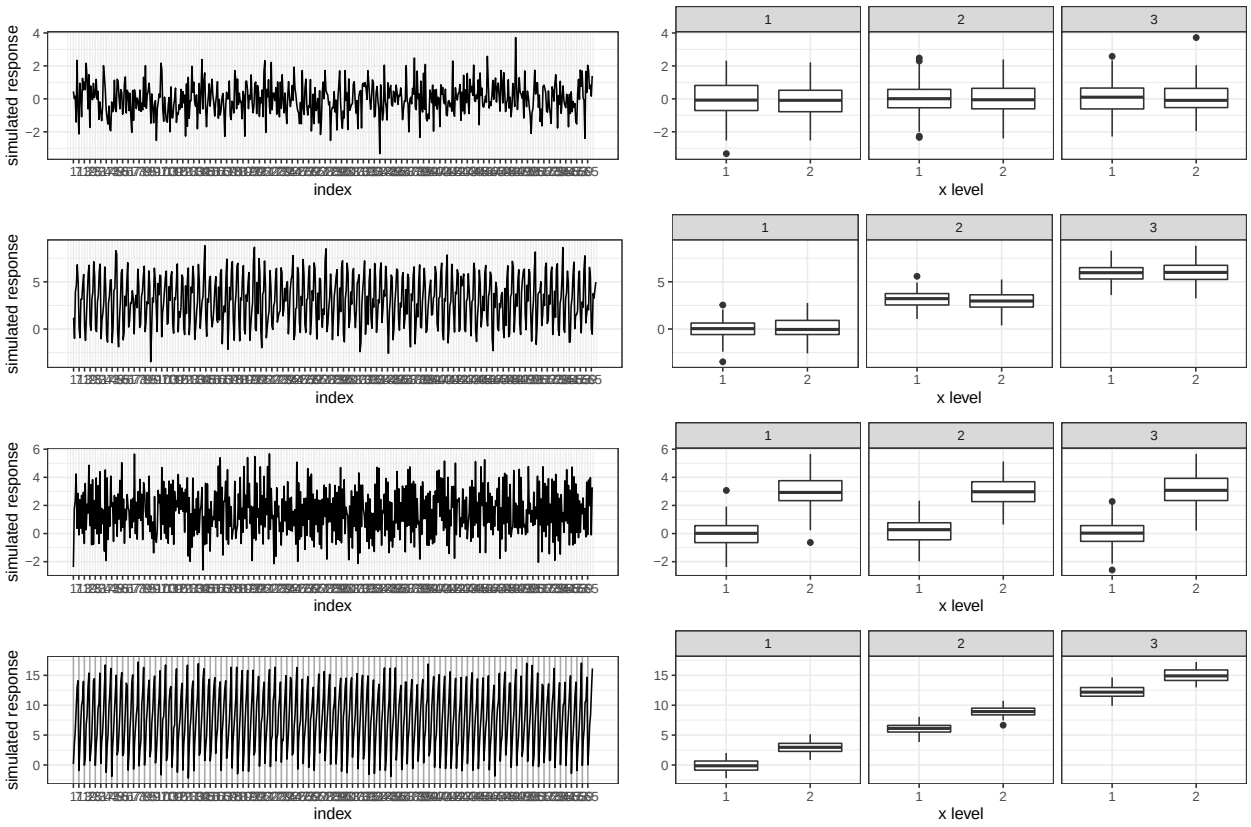
id_facet	id_x	sim_data
1	1	1.0840991
1	1	0.8431089
1	1	0.4943890
1	1	-0.7730161
1	2	5.9038161
1	2	3.9088839
1	2	2.0369233
1	2	2.8117113
2	1	7.4203744
2	1	6.9864382
2	1	5.5500595
2	1	7.0716663
2	2	8.5834288
2	2	8.5708607
2	2	8.8833411
2	2	8.7014611
3	1	11.8207616
3	1	14.3329026
3	1	11.7010540
3	1	13.7342014
3	2	15.4918386
3	2	15.9084482
3	2	15.1561588
3	2	14.8834948

id_facet	id_x	sim_data	index_old	index_new	time
1	1	1.0840991	1	1	1
1	2	5.9038161	5	17	2
2	1	7.4203744	9	10	3
2	2	8.5834288	13	3	4
3	1	11.8207616	17	19	5
3	2	15.4918386	21	12	6
1	1	0.8431089	2	5	7
1	2	3.9088839	6	21	8
2	1	6.9864382	10	14	9
2	2	8.5708607	14	7	10
3	1	14.3329026	18	23	11
3	2	15.9084482	22	16	12
1	1	0.4943890	3	9	13
1	2	2.0369233	7	2	14
2	1	5.5500595	11	18	15
2	2	8.8833411	15	11	16
3	1	11.7010540	19	4	17
3	2	15.1561588	23	20	18
1	1	-0.7730161	4	13	19
1	2	2.8117113	8	6	20
2	1	7.0716663	12	22	21
2	2	8.7014611	16	15	22
3	1	13.7342014	20	8	23
3	2	14.8834948	24	24	24



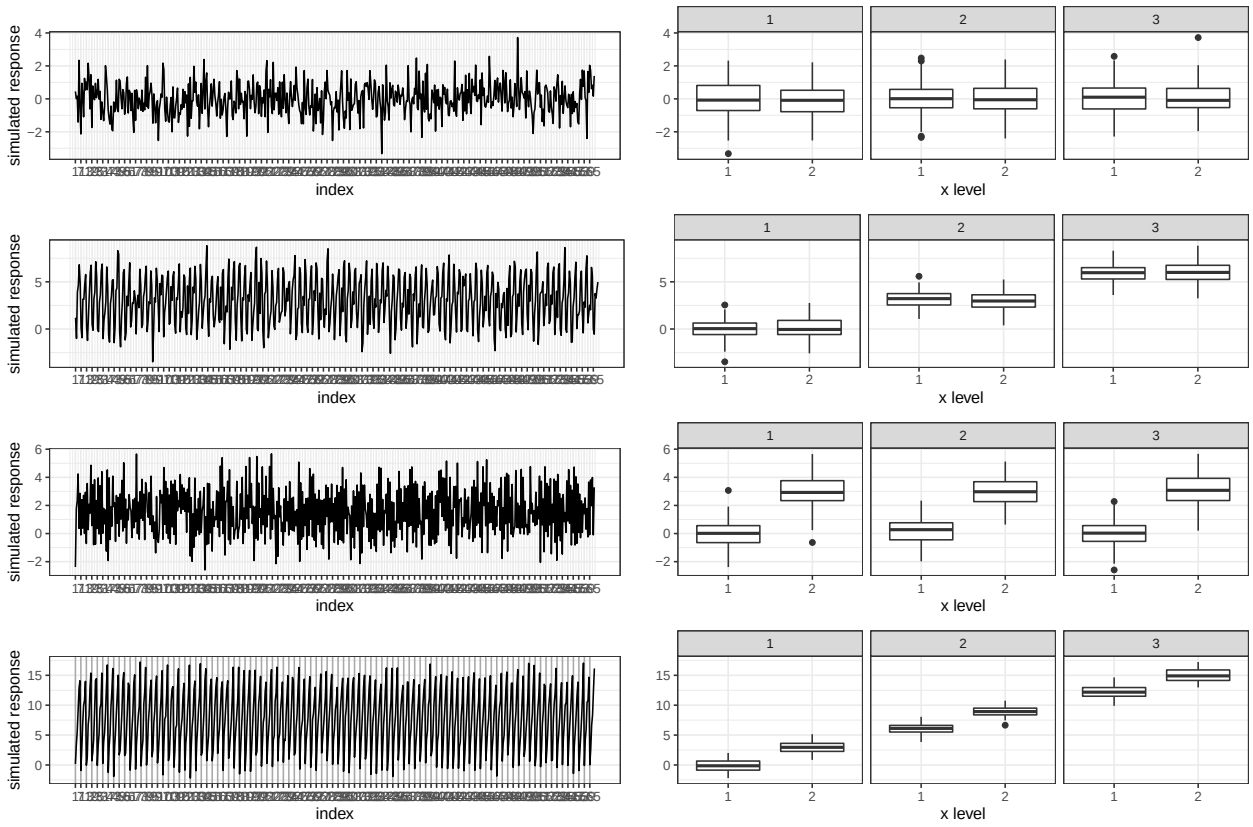
1.2 Number of obervations per combination is 10

```
ntimes_val = 10 # number of observations per combination
```



1.3 Number of obervations per combination is 50

```
ntimes_val = 95 # number of observations per combination
```



2 Simulation: 4 datasets with 2 designs

Generate 4 series, two from design varf and another two from varall and see if clustering happens properly through the approaches

2.1 Generate 2 time series from varx

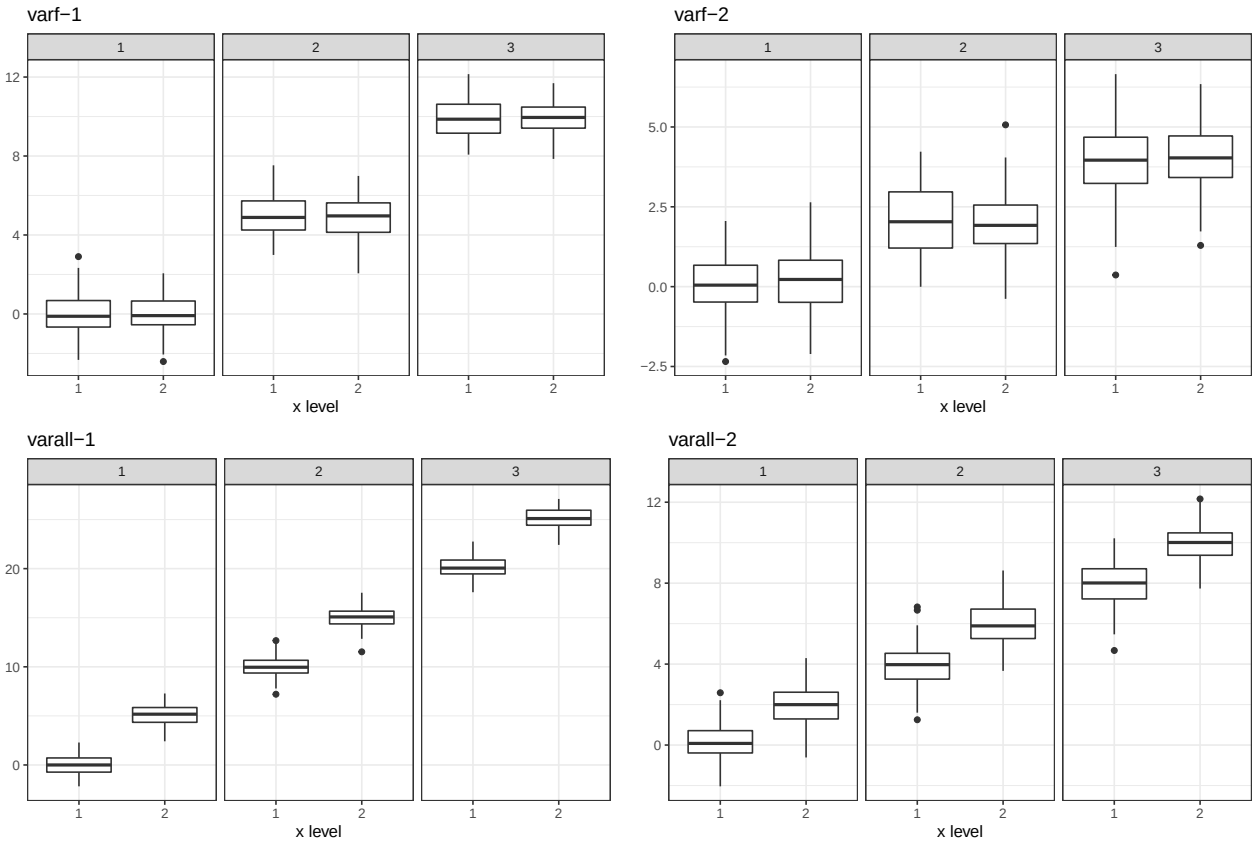
```
data1_varf <- sim_panel(
  nx = nx_val, nfacet =  nfacet_val,
  ntimes = ntimes_val,
  # sim_dist = sim_varf_normal(2, 3, 5, 10, 5, 5)
  sim_dist = sim_varf_normal(2, 3, 0, 1, 5, 0)
) %>% unnest(data)


data2_varf <- sim_panel(
  nx = nx_val, nfacet =  nfacet_val,
  ntimes = ntimes_val,
  # sim_dist = sim_varf_normal(2, 3, 5, 10, 5, 5)
  sim_dist = sim_varf_normal(2, 3, 0, 1, 2, 0)
) %>% unnest(data)
```

2.2 Generate 2 time series from varall

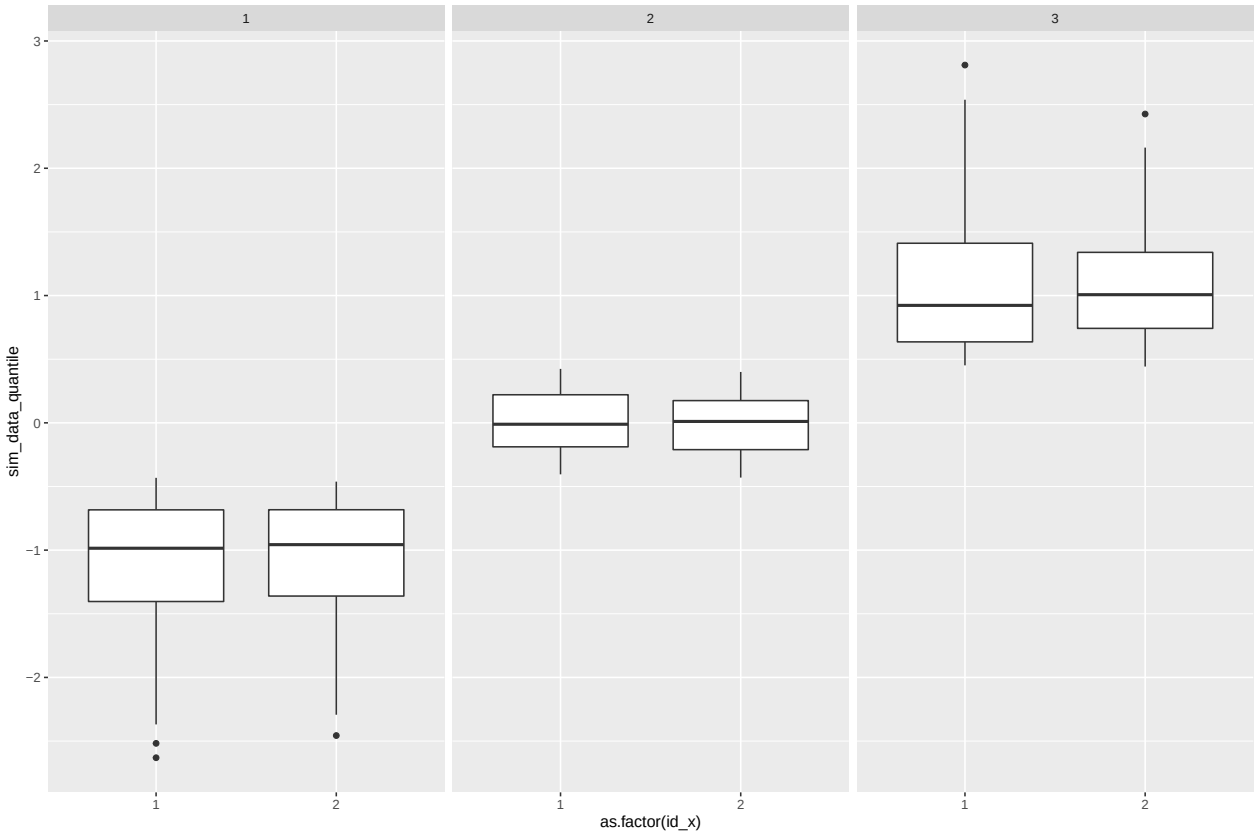
```
data1_varall <- sim_panel(
  nx = nx_val, nfacet =  nfacet_val,
```

```
  ntimes = ntimes_val,
  # sim_dist = sim_varf_normal(2, 3, 5, 10, 5, 5)
  sim_dist = sim_varall_normal(2, 3, 0, 1, 5, 0)
) %>% unnest(data)

data2_varall <- sim_panel(
  nx = nx_val, nfacet =  nfacet_val,
  ntimes = ntimes_val,
  # sim_dist = sim_varf_normal(2, 3, 5, 10, 5, 5)
  sim_dist = sim_varall_normal(2, 3, 0, 1, 2, 0)
) %>% unnest(data)
```

2.3 How do they look?



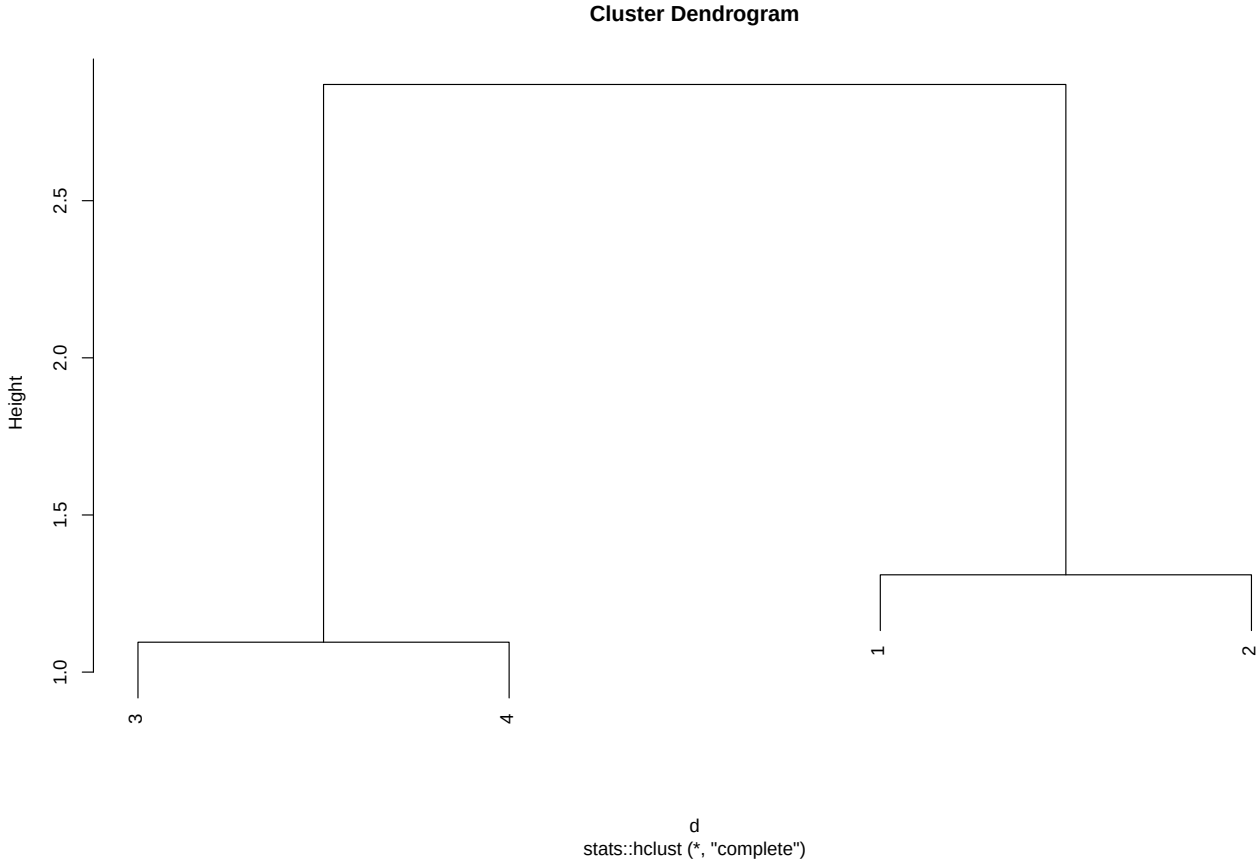
2.4 Compute quantiles of conditional distributions



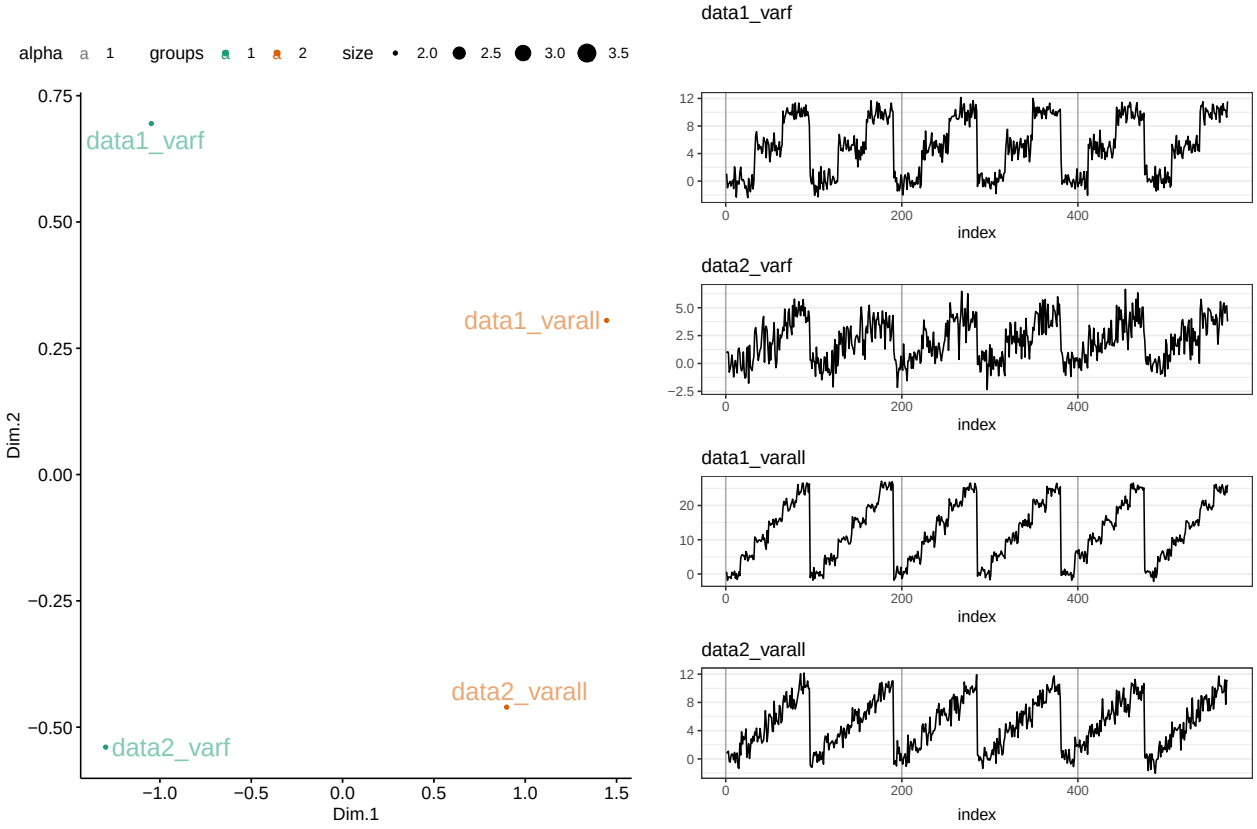
2.5 JS Pairwise distances between data sets

```
data_q_wide <- data_q %>%
  pivot_wider(names_from = data_type,
              values_from = sim_data_quantile) %>%
  select(-c(1,2))

ndata <- data_q %>% distinct(data_type) %>% pull(data_type)
ldata <- length(ndata)
lcomb <-  nx_val*nfacet_val

dist_data <- map(1:ldata, function(x){ # first data
  map(1:ldata, function(y){ # 2nd data
    map(1:lcomb, function(z){ # number of combinations nx*nfacet
    JS(
        prob = quantile_prob_val,
        unlist(data_q_wide[z,x]),
        unlist(data_q_wide[z,y])
      ) %>% as_tibble()
  })%>% bind_rows(.id = "combinations")
})%>% bind_rows(.id = "data_type1")
}) %>% bind_rows(.id = "data_type2")
```

2.6 Hierarchical clustering based on pairwise distances

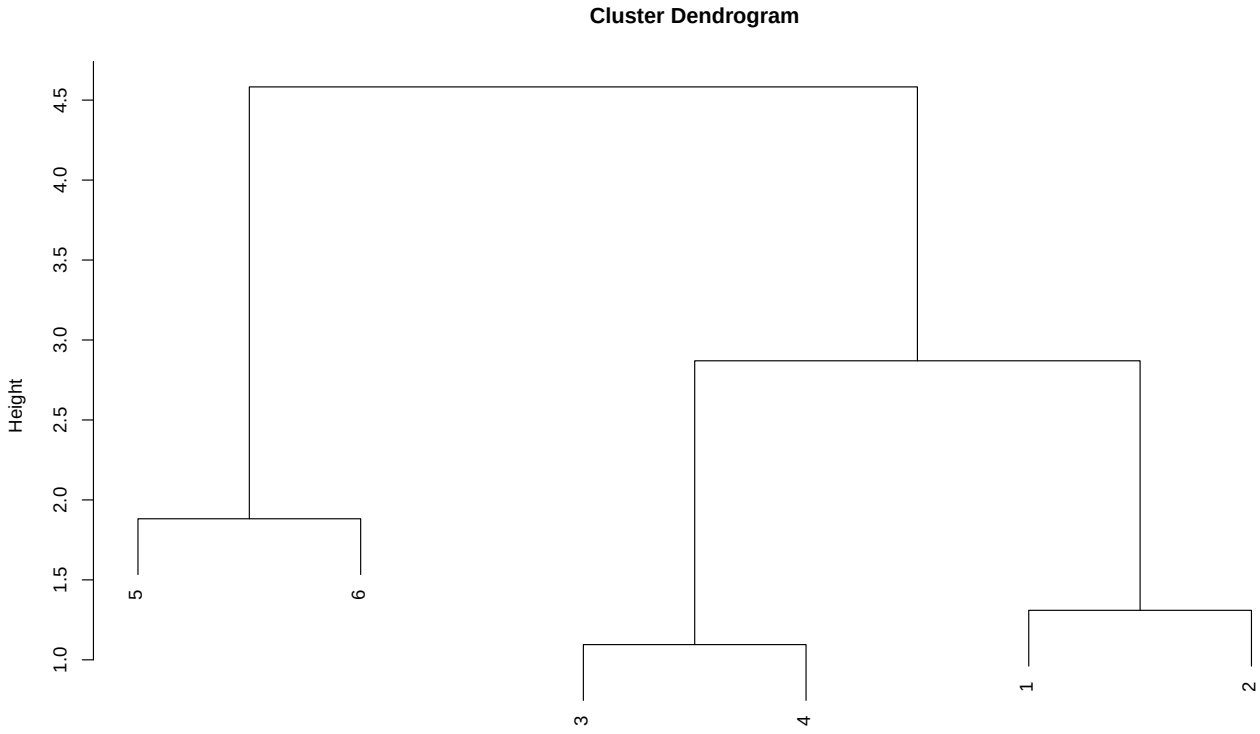


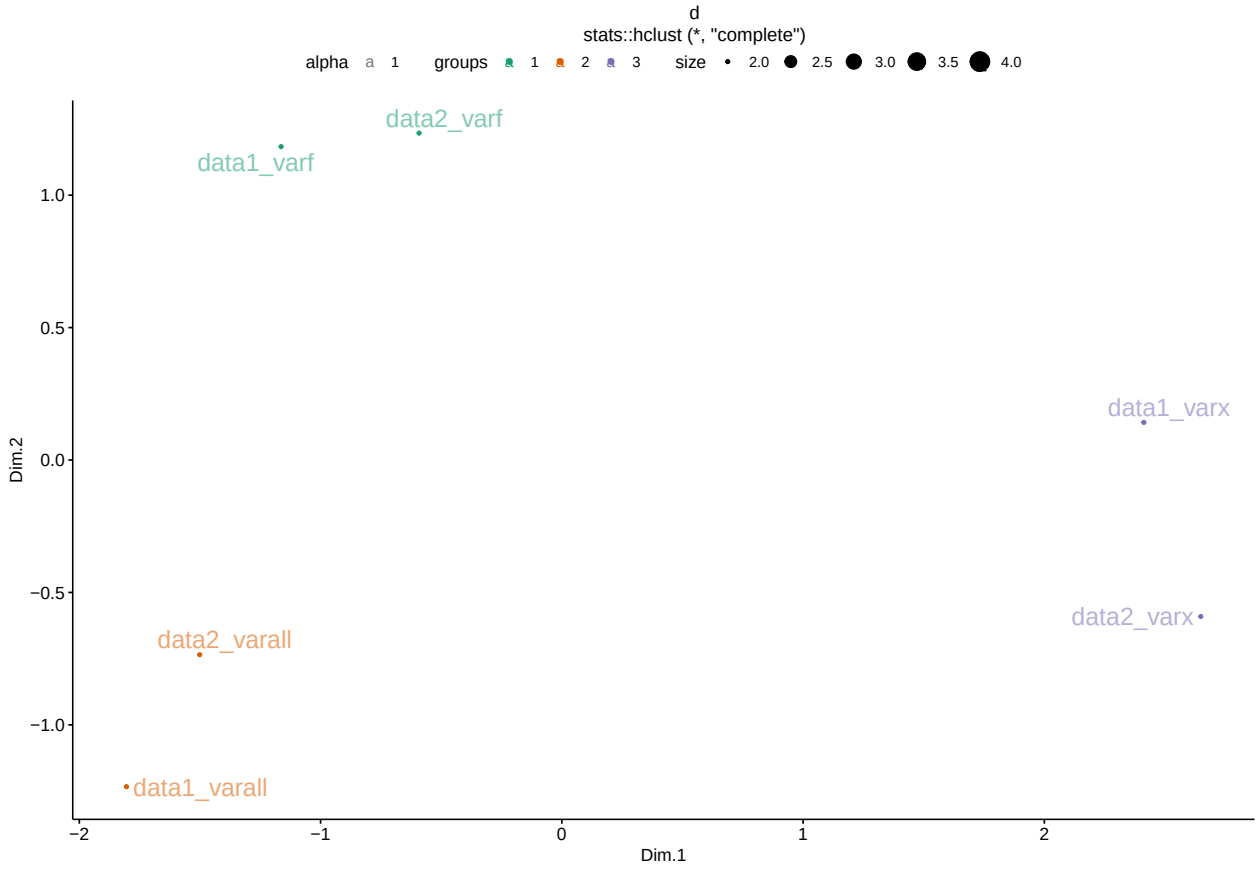
2.7 Multi-dimensional scaling with hierarchical clusters



3 Repeat with 3 designs (for varying sample size)

varf and varall - 100 obs each combination and varx with 10 and 50 observation per combination taken


Cluster Dendrogram
Height
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
5
6
3
4
1
2


d
stats::hclust (*, "complete")
alpha a 1
groups 1 2 3
size 2.0 2.5 3.0 3.5 4.0
data2_varf
data1_varf
data1_varx
data2_varx
data2_varall
data1_varall
Dim.2
1.0
0.5
0.0
−0.5
−1.0
−2
−1
0
1
2
Dim.1