Milk (v5)

Contents

Preliminary

```
library(tidyverse)
library(mixOmics)
walk(dir("~/Documents/timeOmics_dev/R/", pattern = ".R$", full.names = TRUE),source)
```

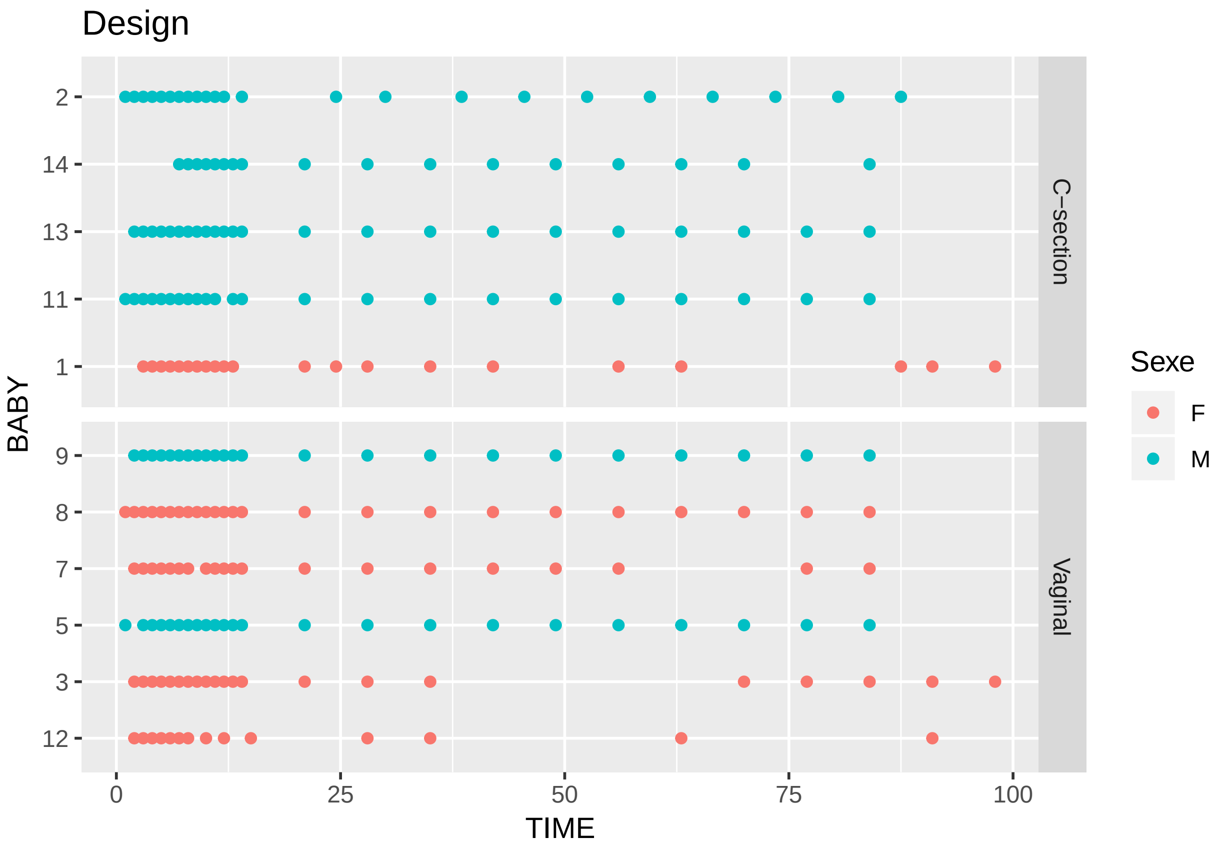
Data Description & Design

Original paper (Development of the Human Infant Intestinal Microbiota, Palmer et al. 2007) studied gastrointestinal microbiome of 14 baby during the first year of life. They collected an average of 26 stool samples from 14 healthy full-term human infants. They have also included vagina and milk microbiome composition from the mothers and stool samples from mothers and fathers.

For demonstration purposes and because babies' gut almost reach an "adult-like" composition, we have focused our attention on the first 100 days of life. We have also excluded babies who recieved an antibiotic treatment during that period, because antibiotics can change drastically microbiome composition.

Our final design consists in an average of 21 timepoits for each of the 11 selected babies.

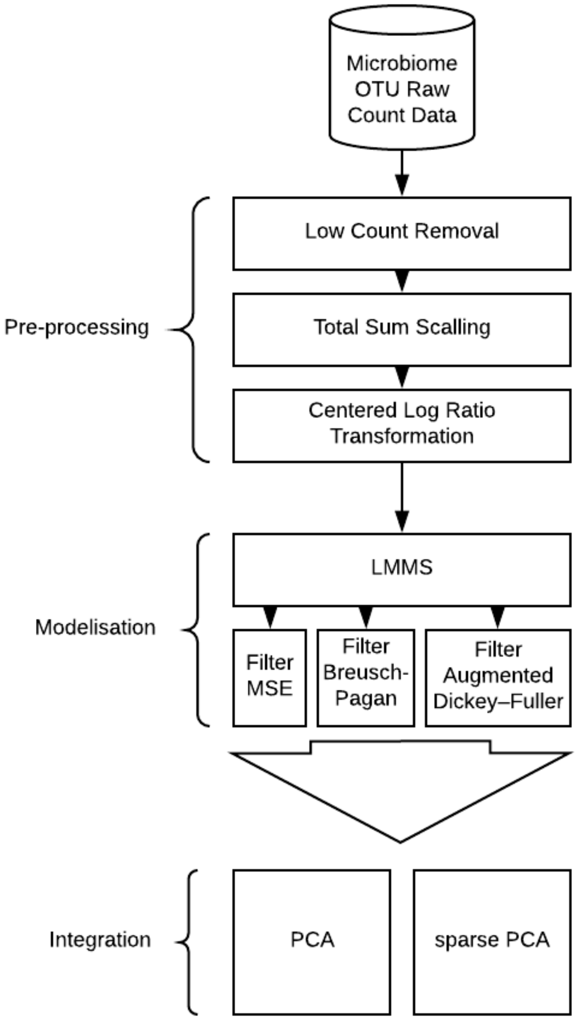
```
load("./milk_data.RData")
ggplot(data= design , aes(x = TIME, y = BABY, color = Sexe)) +
  geom_point() + facet_grid(Delivery~., scales = "free_y") + ggtitle("Design")
```



```
design %>% dplyr::select(BABY, TIME) %>% mutate(BABY = as.numeric(BABY)) %>%
  group_by(BABY) %>%summarise(n_timepoints = n()) %>% knitr::kable()
```

BABY	n_timepoints
1	21
2	23
3	21
5	23
7	20
8	24
9	23
11	23
12	14
13	23
14	17

Workflow



Analysis

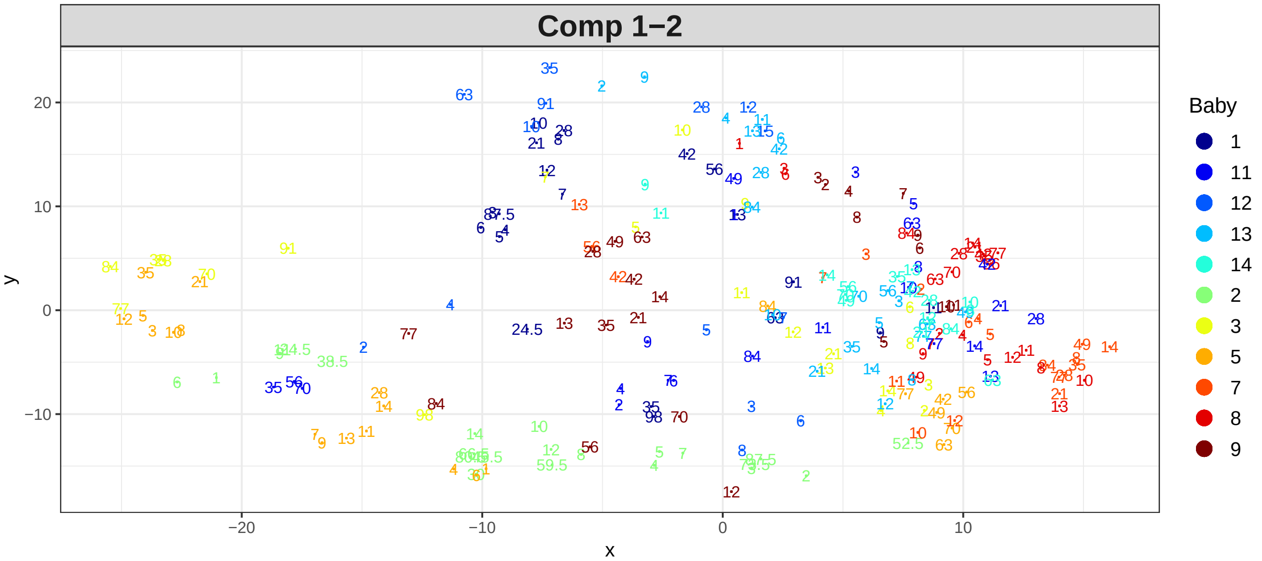
Pre-processing

I perform standard preprocessing steps :

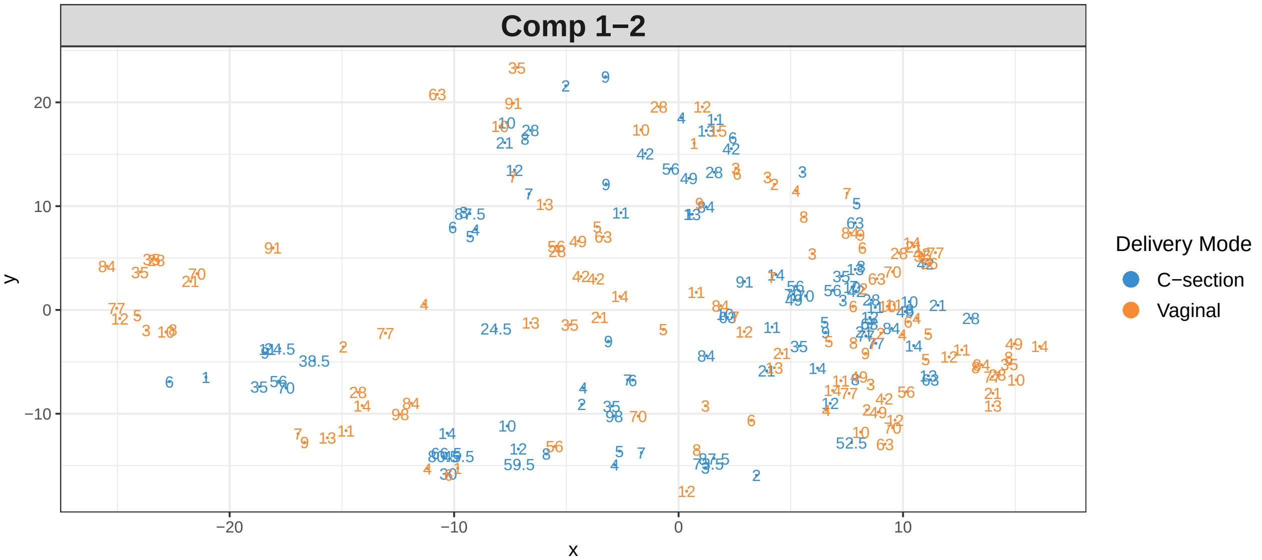
- Low Count Removal
- Total Sum Scalling
- Centered Log Ratio Transformation

```
OTU_norm <- norm_OTU(OTU, AR = T)
# option AR(Abondance Relative) = data allready in AR.
# need to add a "pouillème" in order to cumpute CLR.

#pca(OTU_norm, ncomp = 10) %>% plot
```

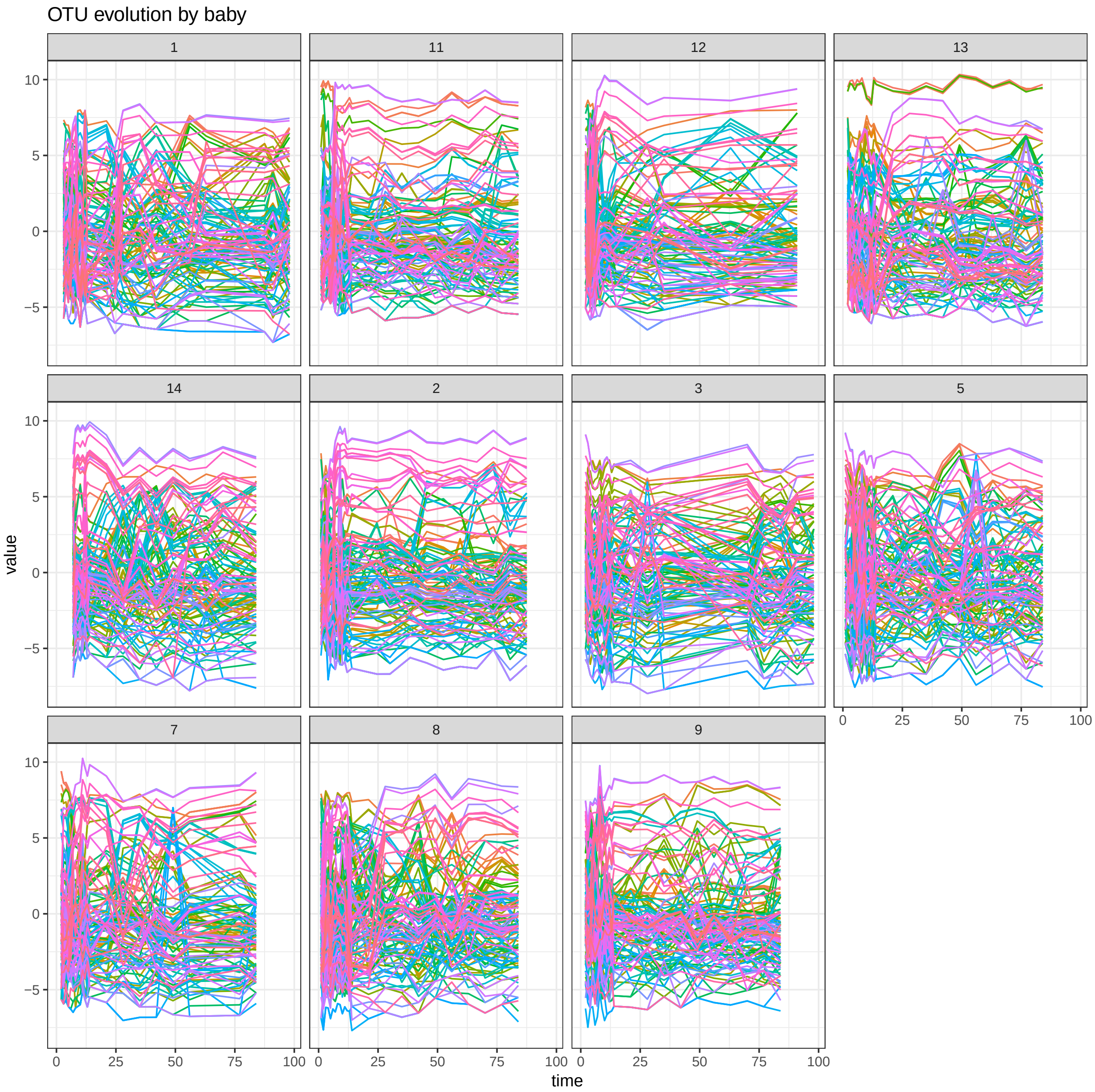
```r
pca.res <- pca(OTU_norm, ncomp = 4)

plotIndiv(pca.res, group = design$BABY, ind.names = design$TIME,
          comp = c(1,2), legend.title = "Baby", legend = T, title = "Comp 1-2")
```



```r
plotIndiv(pca.res, group = design$Delivery, ind.names = design$TIME,
          comp = c(1,2), legend.title = "Delivery Mode", legend = T, title = "Comp 1-2")
```



```r
# per sample OTU evolution
OTU_norm %>% as.data.frame() %>% rownames_to_column("sample") %>%
  gather(OTU, value, -sample) %>%
  mutate(time = sample %>% str_split("_") %>% map_chr(~.x[2]) %>% as.numeric)%>%
  mutate(baby = sample %>% str_split("_") %>% map_chr(~.x[1])) %>%
  ggplot(aes(time, value, col=OTU)) + geom_line() + facet_wrap(~baby) + theme_bw() +
  theme(legend.position = "none") + ggtitle("OTU evolution by baby")
```



Splines and Filter

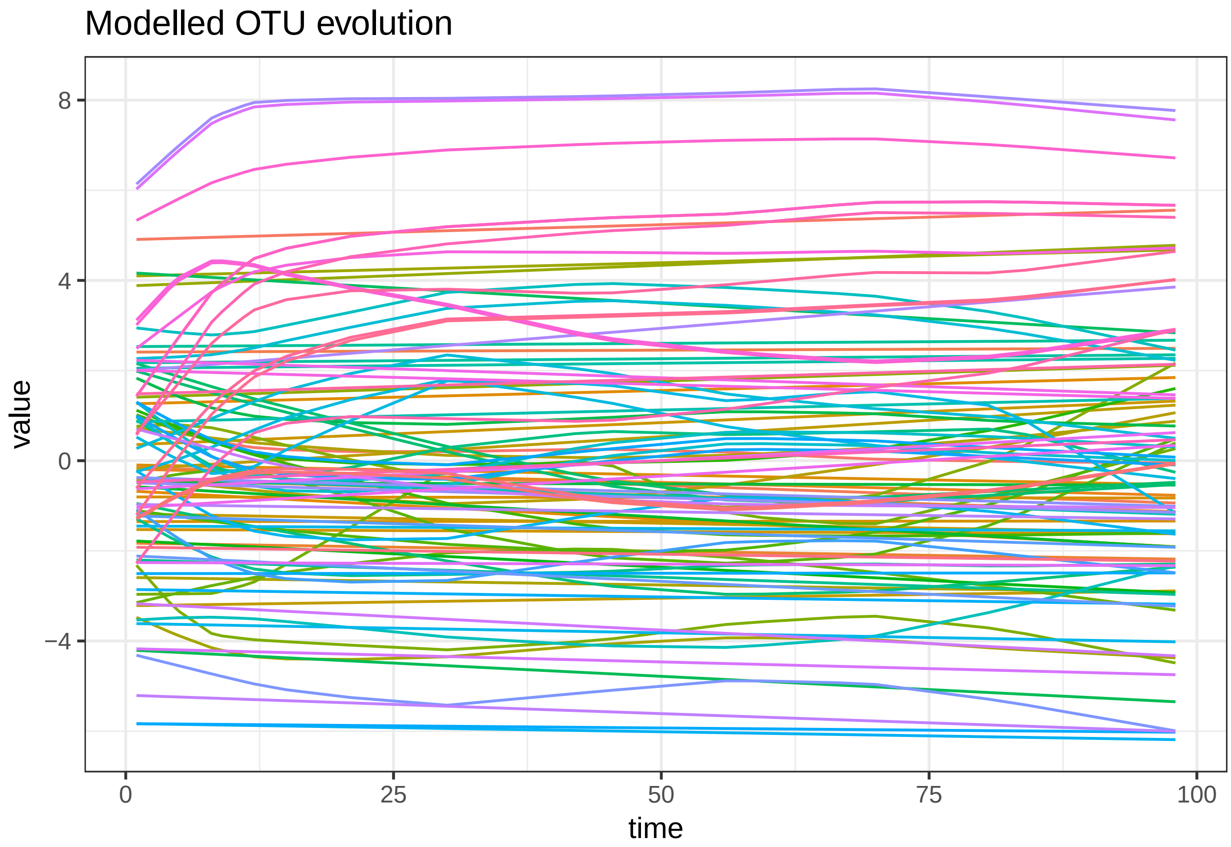
```
time_lmms <- rownames(OTU_norm) %>% str_split("_") %>% map_chr(~.x[2]) %>% as.numeric
sample_id = rownames(OTU_norm)

# cubic p-spline
spline.MILK.cubicpspline = lmms::lmmSpline(data = OTU_norm, time = time_lmms,
                                          sampleID = sample_id,
                                          basis = 'cubic p-spline', keepModels = T,
                                          numCores = 2)

spline.MILK.pspline = lmms::lmmSpline(data = OTU_norm, time = time_lmms,
                                      sampleID = sample_id,
                                      basis = 'p-spline', keepModels = T,
                                      numCores = 2 )

spline.MILK.cubic = lmms::lmmSpline(data = OTU_norm, time = time_lmms,
                                    sampleID = sample_id,
                                    basis = 'cubic', keepModels = T,
                                    numCores = 2)
# summary
spline.MILK.cubicpspline@modelsUsed %>% table %>% as.data.frame() %>%
  set_names("ModelUsed", "Cubic P-spline") %>%
  left_join(spline.MILK.pspline@modelsUsed %>% table%>% as.data.frame() %>%
              set_names("ModelUsed", "P-spline")) %>%
  left_join(spline.MILK.cubic@modelsUsed %>% table%>% as.data.frame() %>%
              set_names("ModelUsed", "Cubic")) %>%
  knitr::kable()
```

ModelUsed	Cubic P-spline	P-spline	Cubic
0	99	76	86
1	24	47	37

I kept P-spline basis (less straight lines). Filter splines based on Homoskedasticity test and MSE cutoff.

```
filter.spline.res <- wrapper.filter.splines(OTU_norm, spline.MILK.pspline)
index.filter <- (rownames(spline.MILK.pspline@predSpline) %in% filter.spline.res$to_keep) %>%
  which()
## filter plot to add ## MSE / pvalue
```

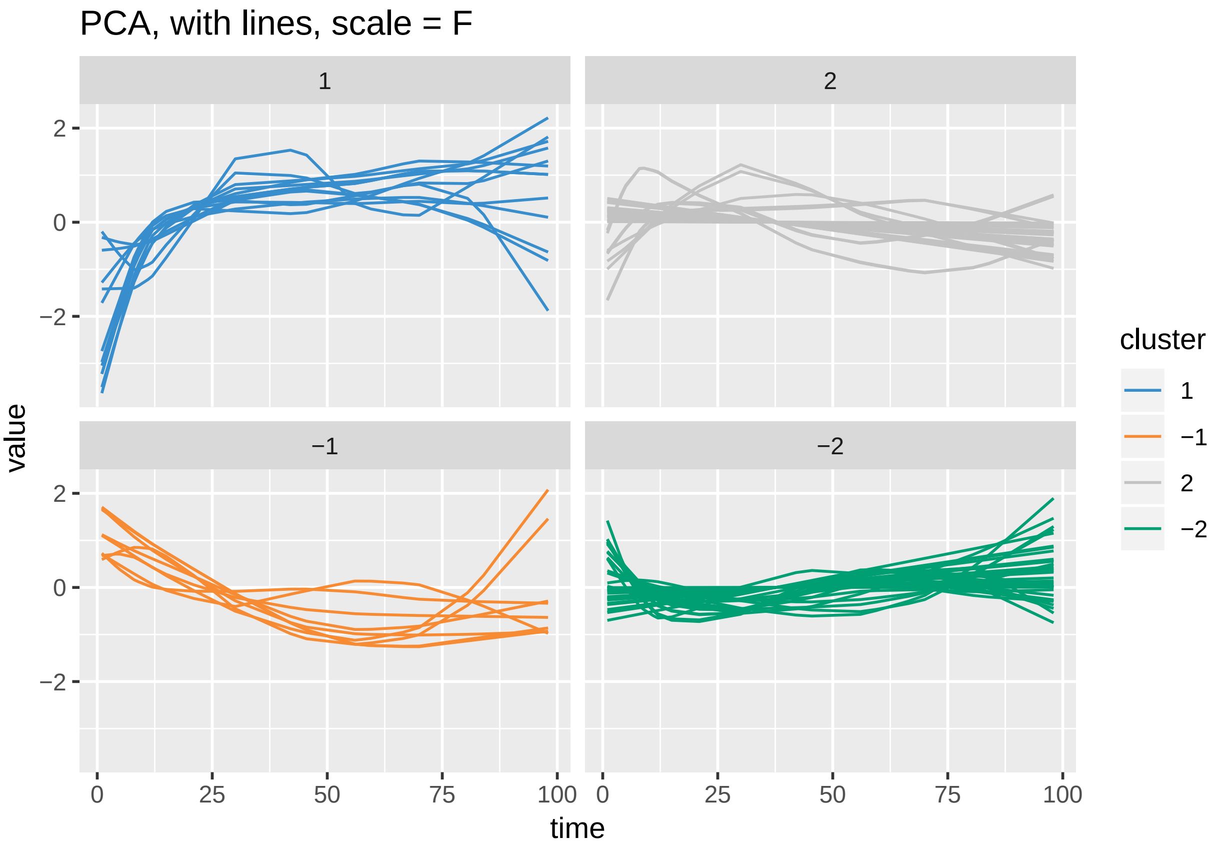
```
spline.data <- spline.MILK.pspline@predSpline[index.filter,] %>% t %>% as.data.frame()
spline.data %>% rownames_to_column("time") %>%
  gather(Features, value, - time) %>% mutate(time =as.numeric(time)) %>%
  ggplot(aes(x=time, y = value, col = Features)) + geom_line() + theme_bw() +
  theme(legend.position = "none") + ggtitle("Modelled OTU evolution")
```



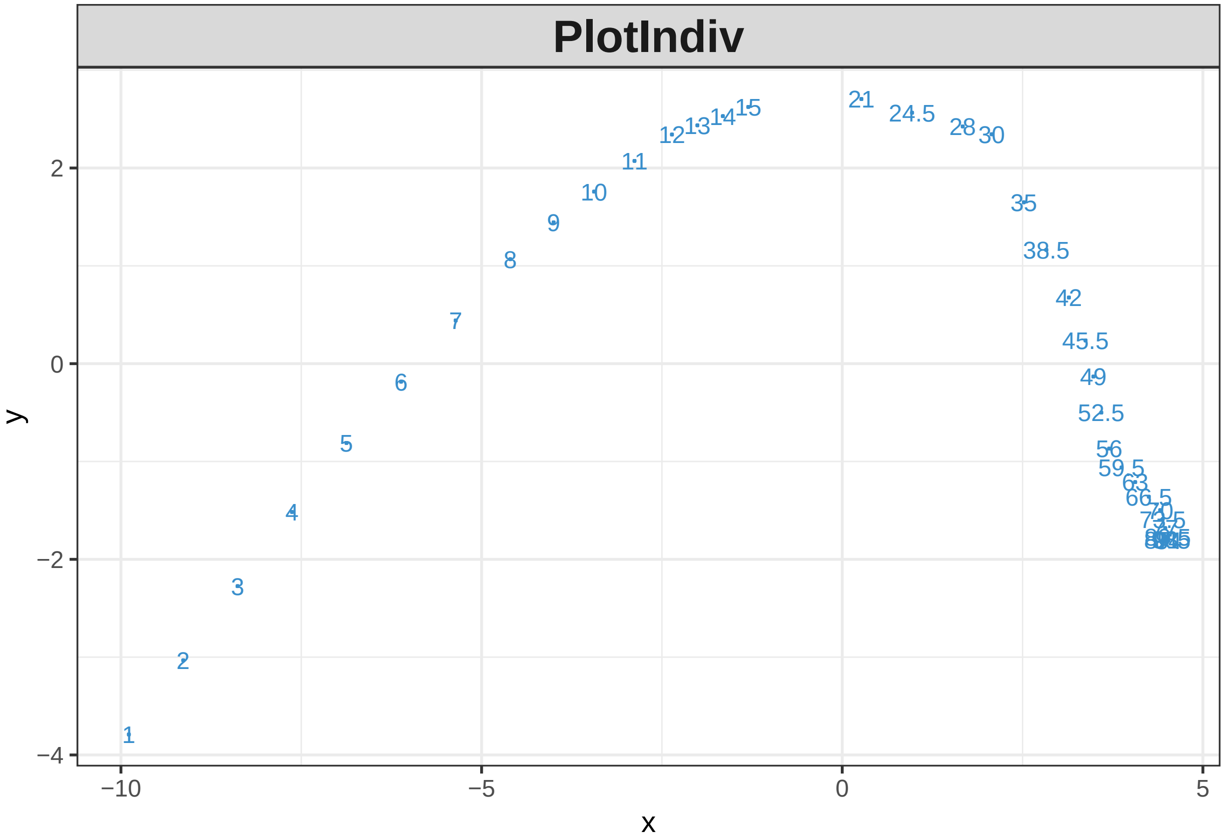
timecourse PCA

With lines

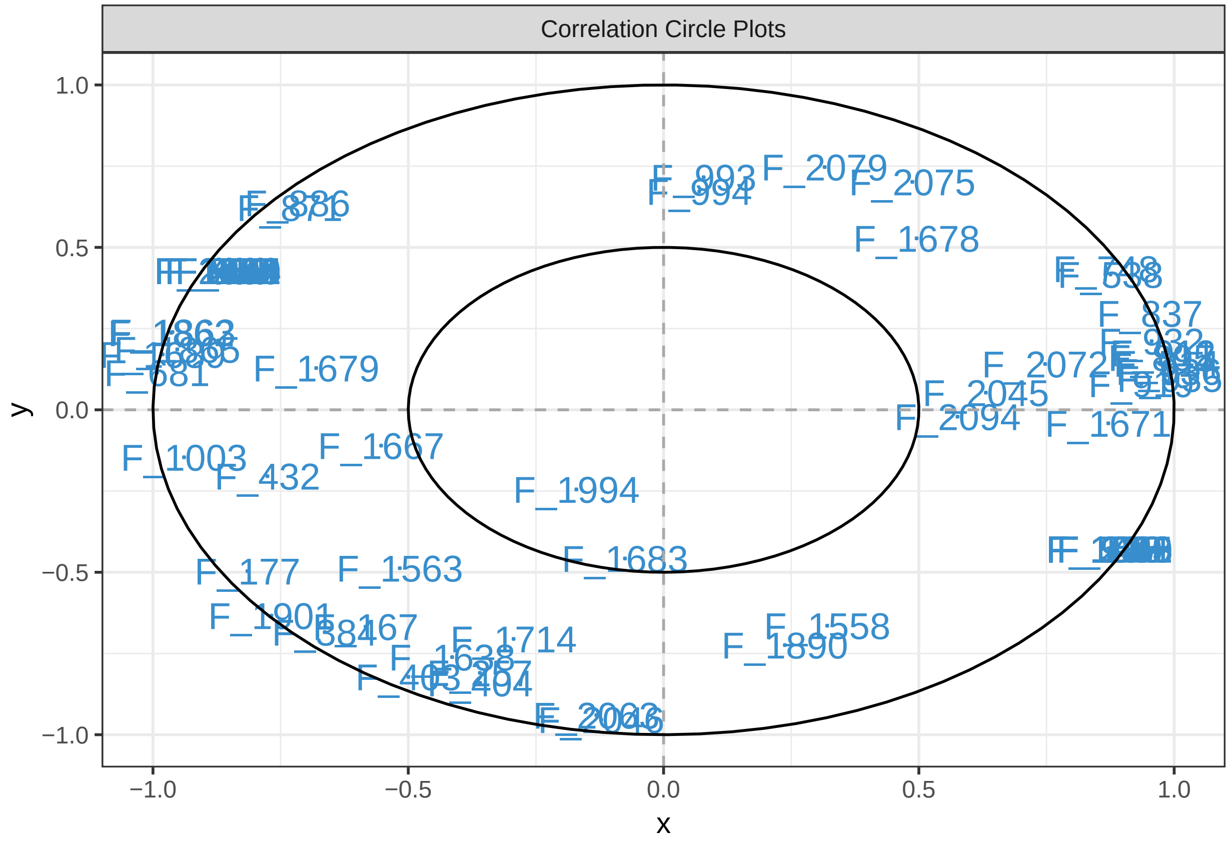
```
pca.res <- pca(spline.data, ncomp = 2, scale = F, center = T)
pca.plot(pca.res, title = "PCA, with lines, scale = F")
```



```
plotIndiv(pca.res)
```



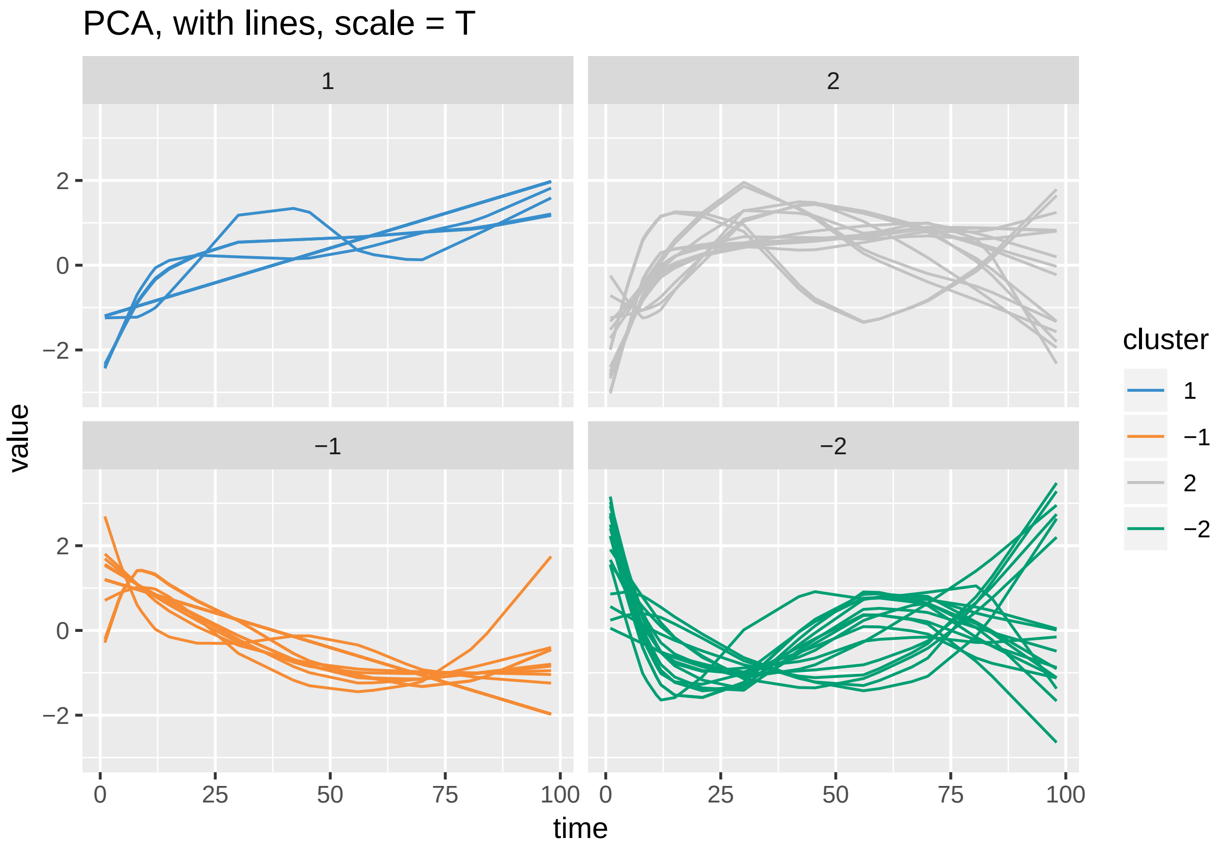
```
plotVar(pca.res)
```



```r
pca.res <- pca(spline.data, ncomp = 2, scale = T, center = T)
pca.get_cluster(pca.res) %>% pull(cluster) %>% table
```

```
## .
## -2 -1  1  2
## 18 43 24 16
```

```r
pca.plot(pca.res, title = "PCA, with lines, scale = T")
```

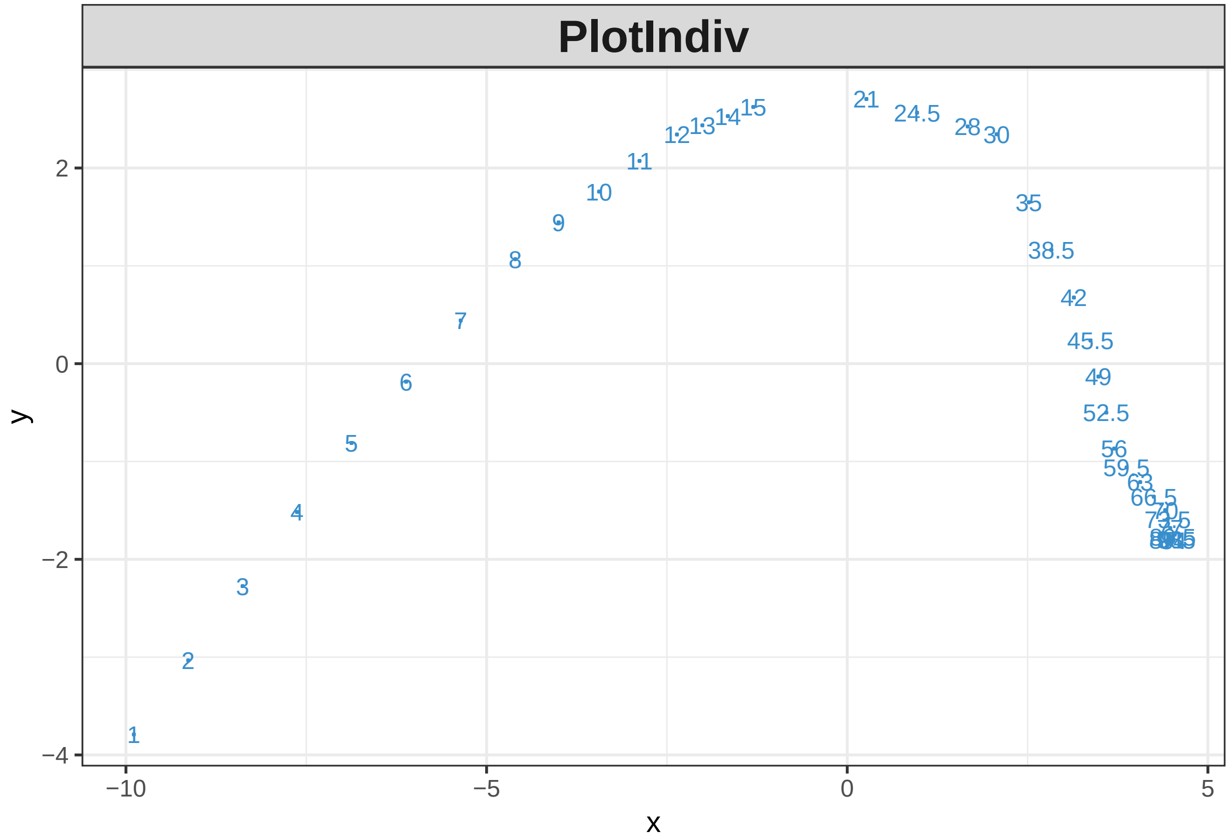


```
pca.res <- pca(spline.data, ncomp = 2, scale = F, center = T)

# silhouette coefficient for this clustering
wrapper.silhouette.pca(spline.data, ncomp = 2, scale = T, center=T)
```

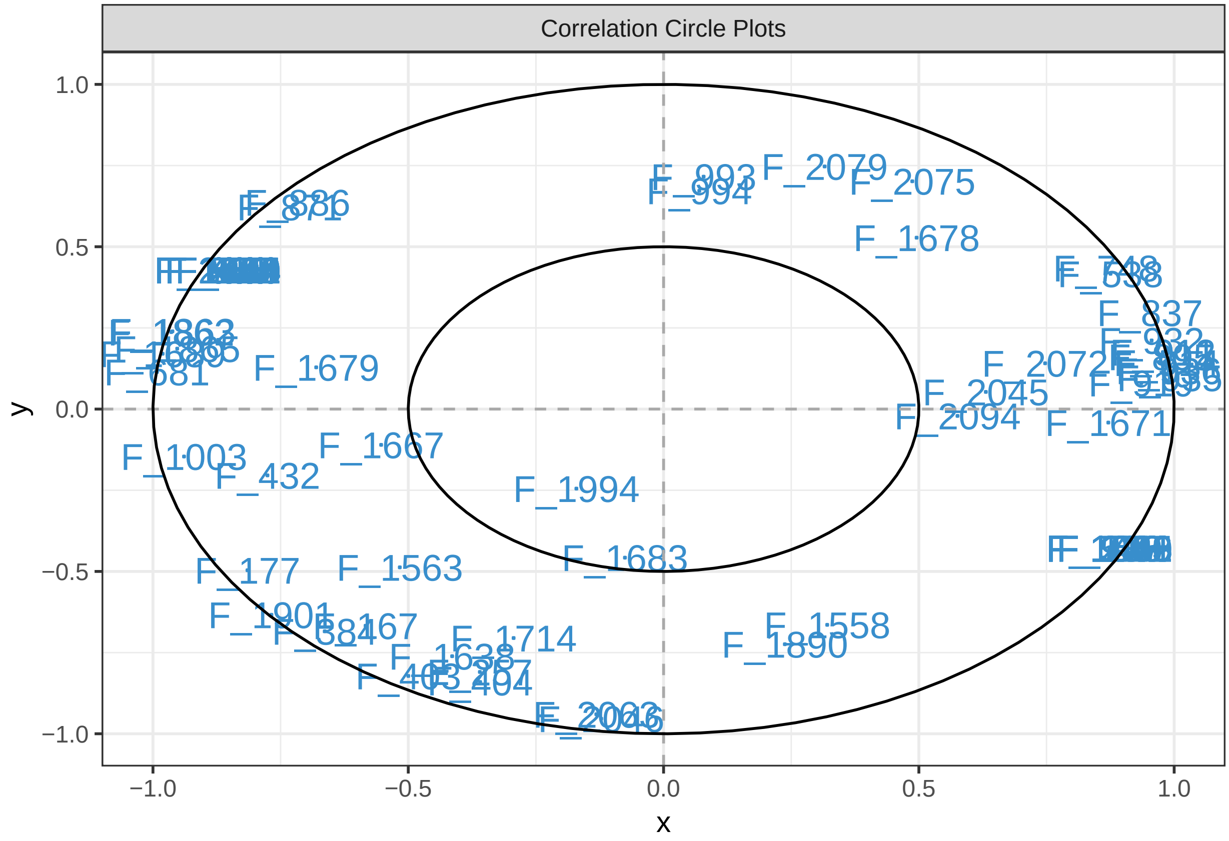
```
## [1] 0.8294685
```

without lines

```
spline.0 <- spline.MILK.pspline@predSpline[spline.MILK.pspline@modelsUsed != 0, ] %>%
  t %>% as.data.frame()
# no filter needed

pca.res.0 <- pca(spline.0, ncomp = 2, scale = T, center = T)

plotIndiv(pca.res)
```



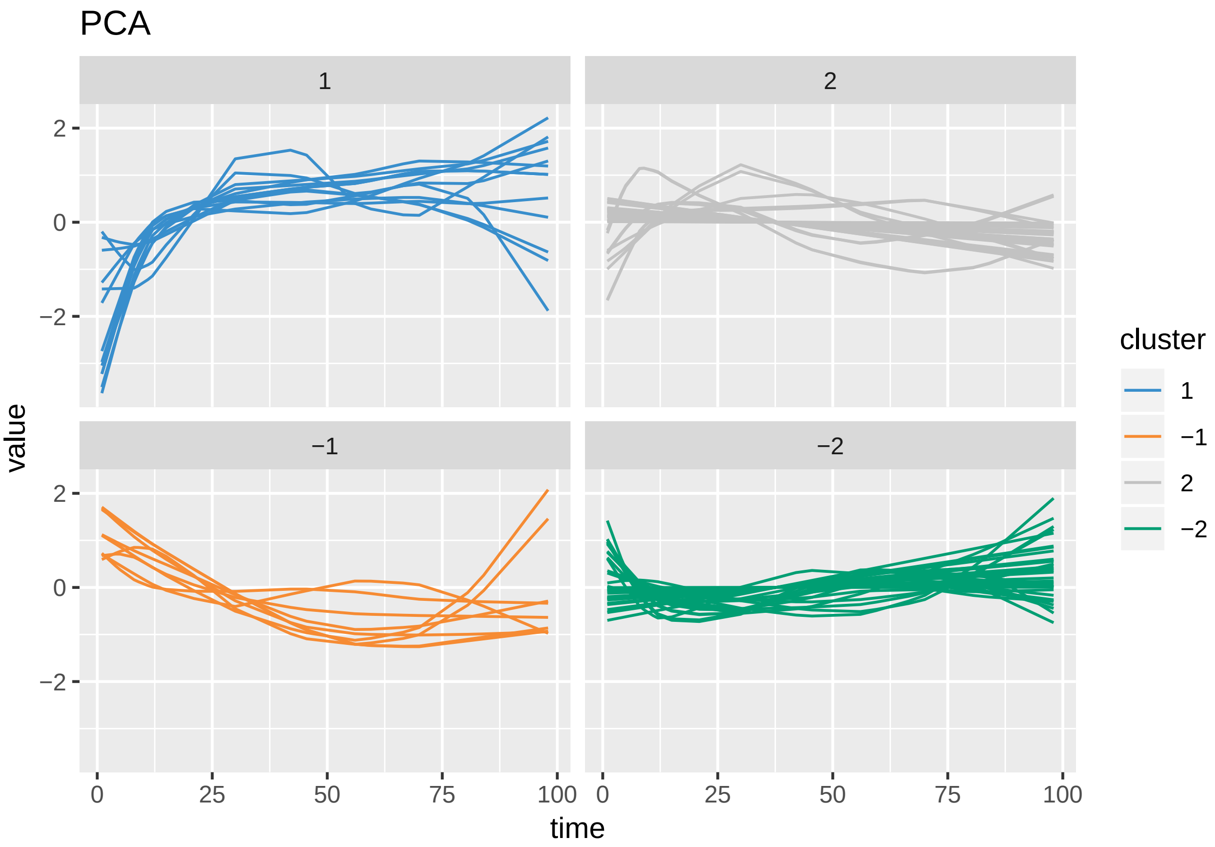
```
plotVar(pca.res)
```



```
pca.get_cluster(pca.res) %>% pull(cluster) %>% table
```

```
## .
## -2 -1  1  2
## 36  9 13 43
```

```
pca.plot(pca.res)
```



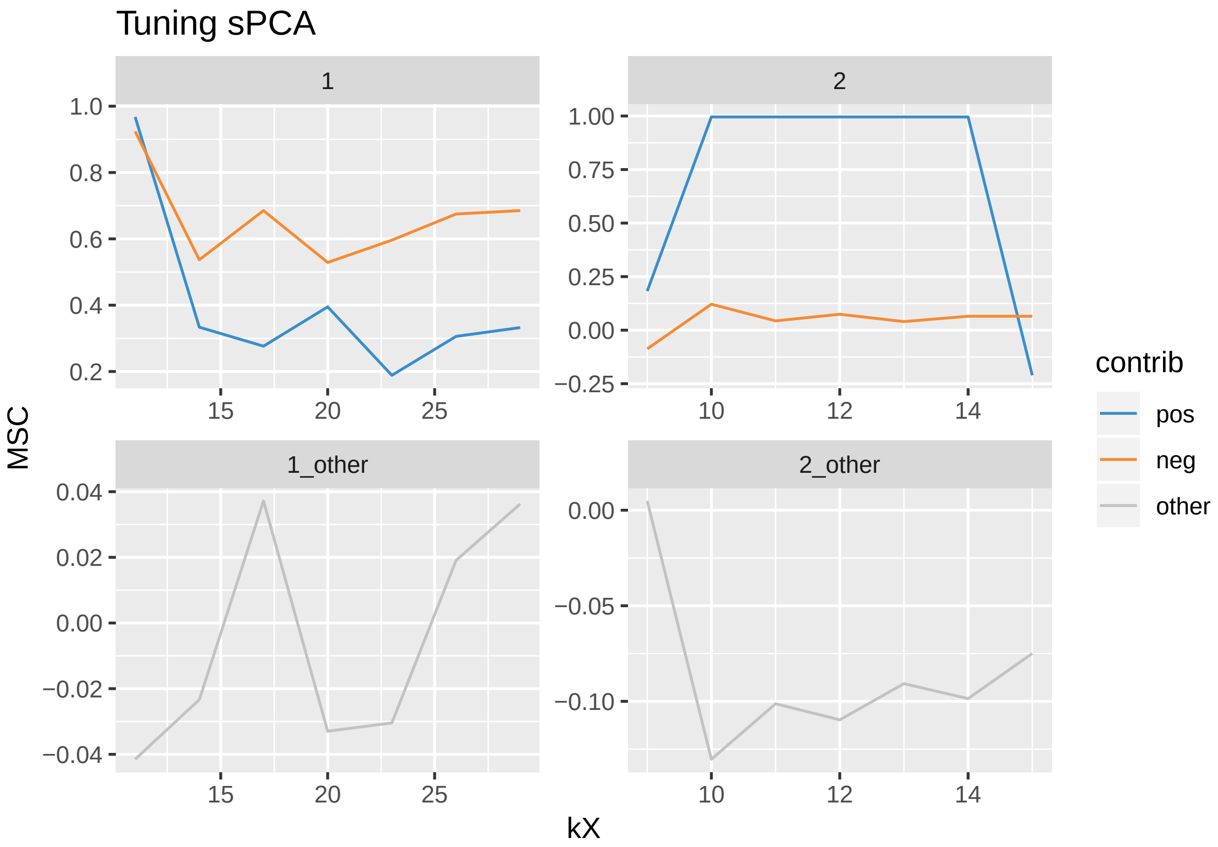
```
# silhouette coefficient for this clustering
wrapper.silhouette.pca(spline.0, ncomp = 2, scale = T, center=T)
```

```
## [1] 0.6624703
```

timecourse sPCA

with lines

```
keepX = list(seq(11,29, 3), seq(9,15,1))
res.tune.spca <- tune.spca(X = spline.data, ncomp = 2, keepX = keepX)
tune.spca.choice.keepX(res.tune.spca, draw = T)
```

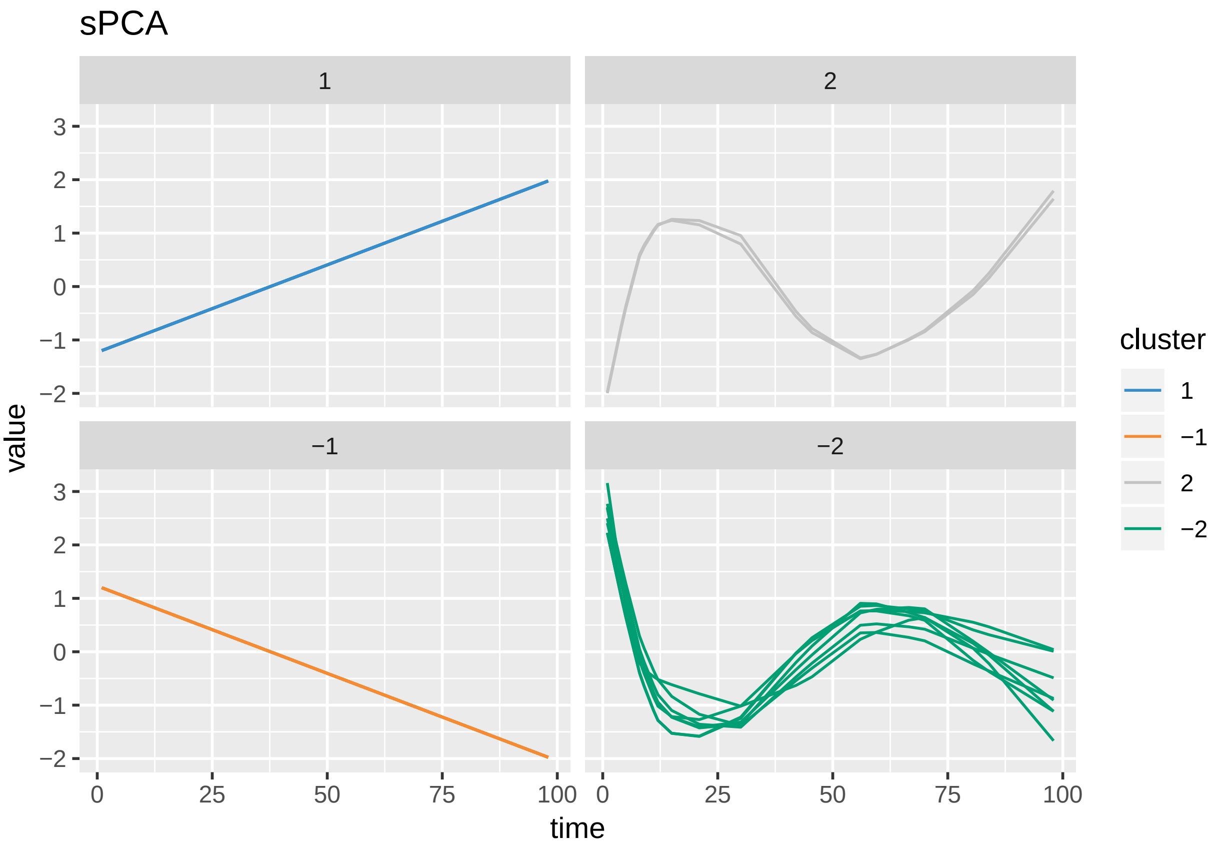


```
## [1] 11 NA
```

```
spca.res_f <- spca(spline.data, ncomp = 2, keepX = c(17,10))

wrapper.silhouette.spca(spline.data, keepX = c(17,10),  ncomp = 2, scale = T, center=T)
```

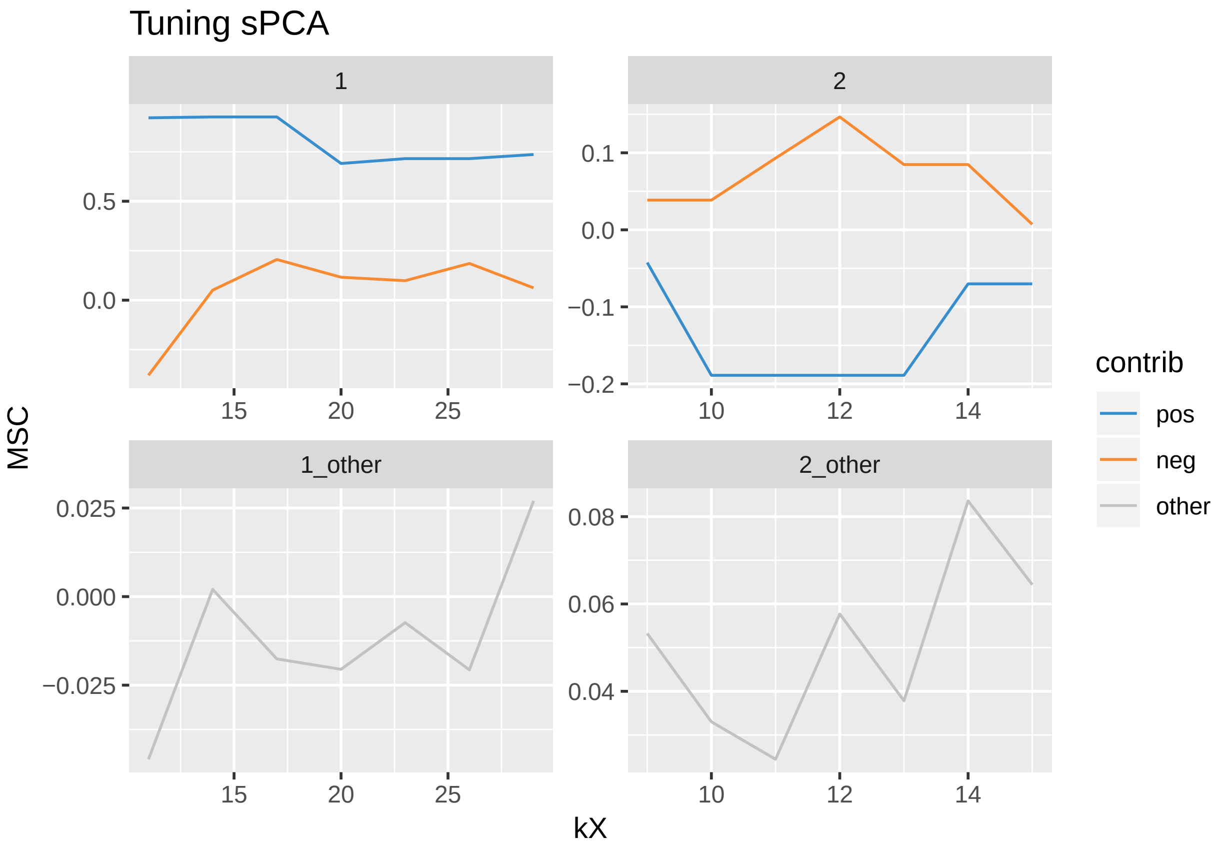
```
## [1] 0.9902438
```

```
spca.plot(spca.res_f)
```



without lines

```
keepX = list(seq(11,29, 3), seq(9,15,1))
res.tune.spca <- tune.spca(X = spline.0, ncomp = 2, keepX = keepX)
tune.spca.choice.keepX(res.tune.spca, draw = T)
```

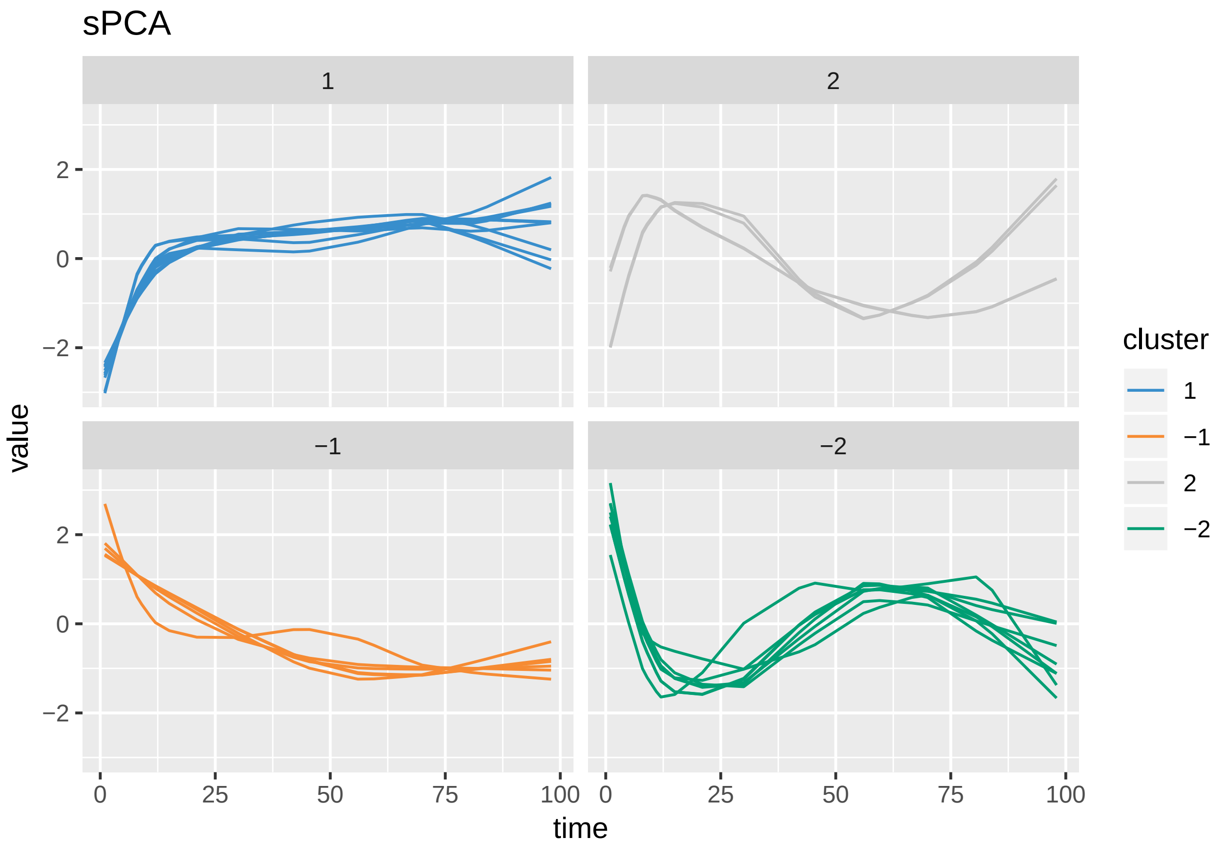


```
## [1] NA NA
```

```r
spca.res_f <- spca(spline.0, ncomp = 2, keepX = c(17,12))

wrapper.silhouette.spca(spline.0, keepX = c(17,12), ncomp = 2, scale = T, center=T)
```

```
## [1] 0.8915965
```

```r
spca.plot(spca.res_f)
```



Interpretation / Conclusion

```
pca.get_cluster(spca.res_f) %>%
  filter(cluster != 0) %>% arrange(cluster) %>%
  left_join(OTUref, by =c("molecule"="Feature")) %>%
  dplyr::select(cluster, Name) %>%
  knitr::kable()
```

molecule	cluster	Name
F_1638	-2	2.30.2.6.3 TAB.ETHANOLICUS
F_167	-2	2.15.1.2.7.089 Bacteroides merdae
F_1890	-2	2.30.7.17 LACTOBACILLI
F_2046	-2	2.30.9.1 C.LEPTUM
F_2063	-2	2.30.9.1.3 C.LEPTUM
F_257	-2	2.15.6 CY.AURANTIACA
F_403	-2	2.21.2 CHLOROPLASTS_AND_CYANELLES
F_404	-2	2.21.2.1 PSEUDANABAENA
F_1003	-1	2.28.4 DELTA_SUBDIVISION
F_1689	-1	2.30.3.2 HELIOBACTERIUM
F_1862	-1	2.30.7.12 STAPHYLOCOCCUS
F_1863	-1	2.30.7.12.1 STAPHYLOCOCCUS
F_1865	-1	2.30.7.12.1.017 Staphylococcus sp.
F_681	-1	2.28.2 BETA_SUBDIVISION
F_538	1	2.28 PROTEOBACTERIA

molecule	cluster	Name
F_748	1	2.28.3 GAMMA_SUBDIVISION
F_837	1	2.28.3.23 VIBRIO
F_895	1	2.28.3.27 ENTERICS_AND_RELATIVES
F_911	1	2.28.3.27.11 KLEBSIELLA
F_912	1	2.28.3.27.11.2 ENB.ASBURIAE
F_914	1	2.28.3.27.11.2.017 Enterobacter sp.
F_919	1	2.28.3.27.13 XENORHABDUS
F_932	1	2.28.3.27.8 WIGGLESWORTHIA_SYMBIONT
F_935	1	2.28.3.4 THIOBACILLUS
F_936	1	2.28.3.4.1 THB.FERROOXIDANS
F_871	2	2.28.3.26 HAEMOPHILUS-PASTEURELLA
F_886	2	2.28.3.26.19 H.ACTINOMYCETEMCOMITANS
F_993	2	2.28.3.9 LEGIONELLA
F_994	2	2.28.3.9.1 LEGIONELLA

Like the original paper, we found microbiome composition tends to converge towards an "adult-like" state.

Despite inter-individual variations, we are able to provide a more or less complex modelisation for the most abundant taxa. Some features were modelled with a straight line which can be explain by a too high inter-individual variation.

We are also able to identify the main patterns of the dynamic that occurs during the first 100 years of life.

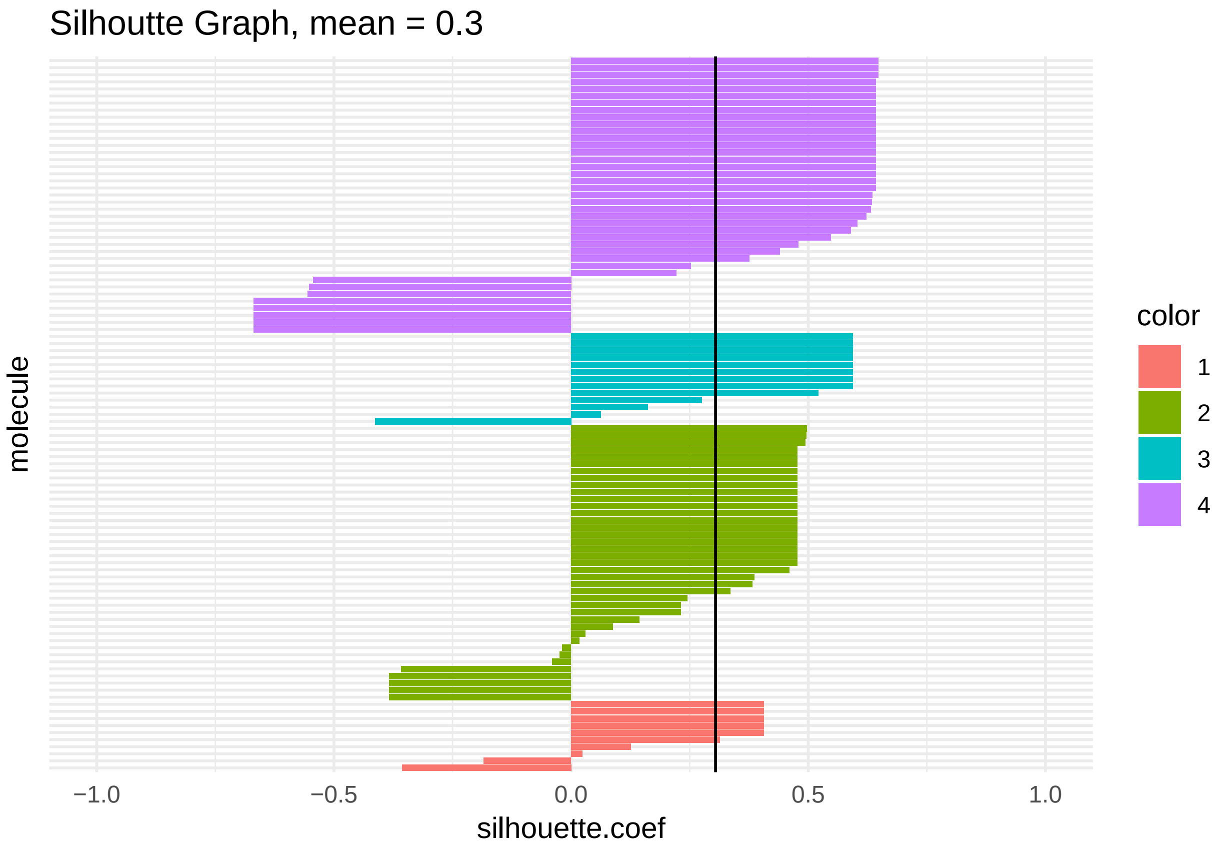
Use of fPCA.

To compare our results

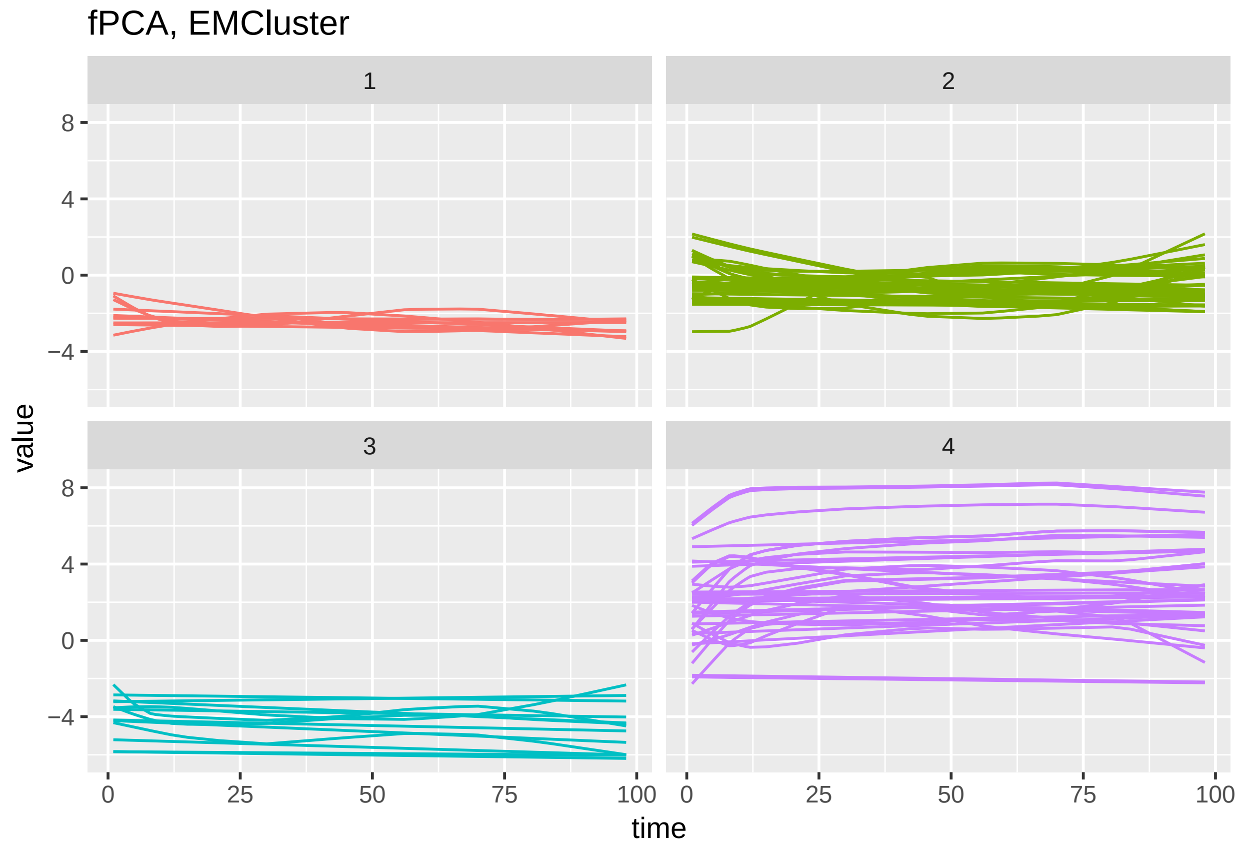
```r
library(fdapace)

data <- as.matrix(spline.data)

# prepare fclust input
FPCA_input <- MakeFPCAInputs(IDs = colnames(data) %>% rep(each=dim(data)[1]),
                             tVec = rep(rownames(data) %>% as.numeric(),dim(data)[2]),
                             yVec = data)

fclust.res <- FClust(FPCA_input$Ly, FPCA_input$Lt,
                     optnsFPCA = list(userBwCov= 2, FVEthreshold = 0.90),
                     k = 4, cmethod = "EMCluster")

tmp <- bind_cols(as.data.frame(colnames(data)),
                 as.data.frame(as.character(fclust.res$cluster))) %>%
    set_names(c("molecule", "cluster"))

DF <- Spearman_distance(data)
B <- Add_Cluster_metadata(DF, tmp)
SC.fpca.1 <- SIlhouette_coef_df(B)
mean(SC.fpca.1$silhouette.coef)
```

```
## [1] 0.3044677
```

```r
plot_silhouette_order_color(SC.fpca.1)
```



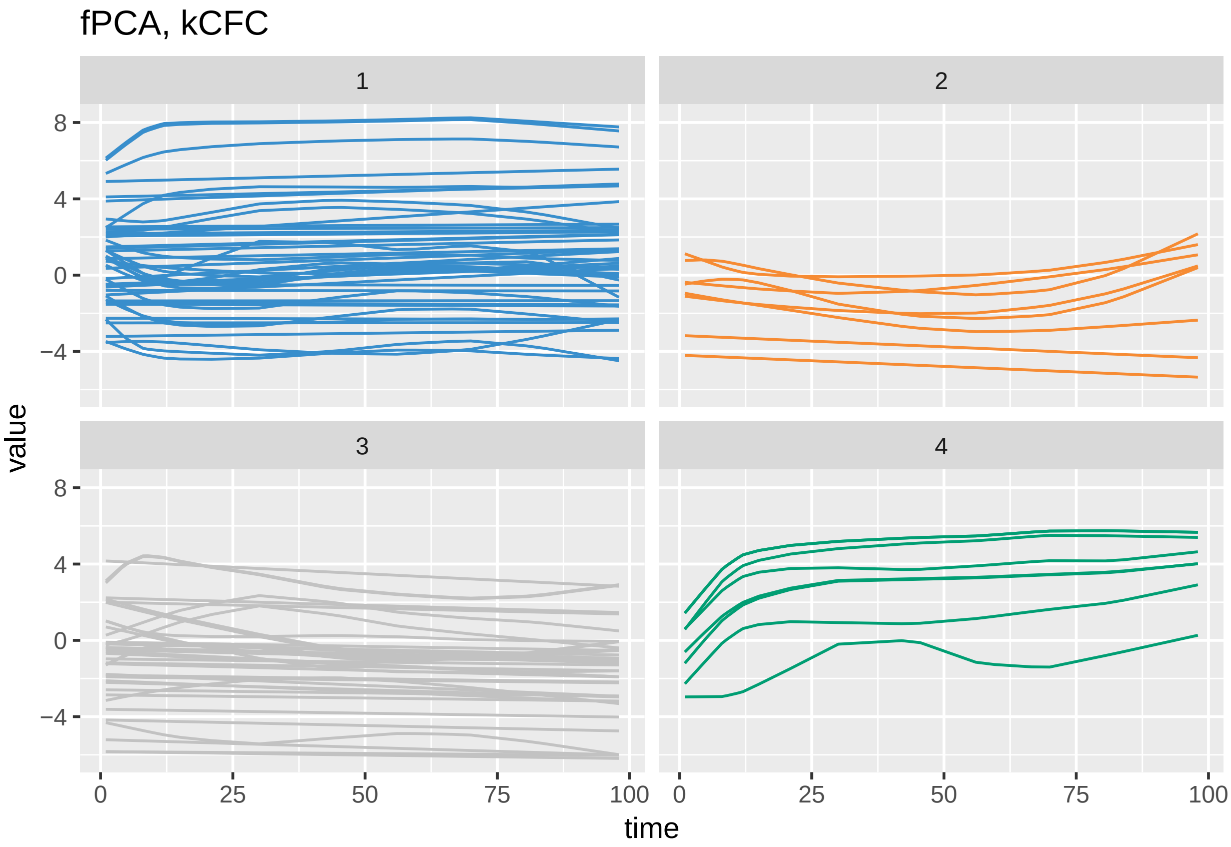
```
## plot clusters
data %>% as.data.frame() %>% rownames_to_column("time") %>%
  gather(molecule, value, -time) %>%
  left_join(tmp) %>%    # add cluster metadata %>%
  mutate(time = as.numeric(time)) %>%
  ggplot(aes(x=time, y=value, group=molecule, color = as.factor(cluster))) +
  geom_line() + facet_wrap(~as.factor(cluster)) + theme(legend.position="none") +
  ggtitle("fPCA, EMCluster")
```



```
# idem with
fclust.res.2 <- FClust(FPCA_input$Ly, FPCA_input$Lt,
                           optnsFPCA = list(userBwCov= 2, FVEthreshold = 0.90),
                           k = 4, cmethod = "kCFC")

tmp <- bind_cols(as.data.frame(colnames(data)),
                 as.data.frame(as.character(fclust.res.2$cluster))) %>%
    set_names(c("molecule", "cluster"))

B <- Add_Cluster_metadata(DF, tmp)
SC.fpca.2 <- SIlhouette_coef_df(B)
mean(SC.fpca.2$silhouette.coef)
```

```
## [1] 0.5300476
```

```
## plot clusters
data %>% as.data.frame() %>% rownames_to_column("time") %>%
  gather(molecule, value, -time) %>%
  left_join(tmp) %>%   # add cluster metadata %>%
  mutate(time = as.numeric(time)) %>%
  ggplot(aes(x=time, y=value, group=molecule, color = as.factor(cluster))) +
  geom_line() + facet_wrap(~as.factor(cluster)) + theme(legend.position="none") +
  scale_color_manual(values=color.mixo(1:4)) + ggtitle("fPCA, kCFC")
```



result silhouette summary

Method	Mean silhouette coef.
PCA (w lines)	.8295
PCA (w/o lines)	.6625
sPCA (line)	.9902
sPCA (w/o line)	.8916
fPCA (kcfc)	.5300
fPCA (GMM)	.3045