Reproducible Analysis: The *Two-Wing* admixed structure of environmental microbial communities

Yushi Tang

2022-09-05

Preface

This document provides reproducible research records for our manuscript about the *Two-Wing* admixed structure of environmental microbial communities. We provide step-by-step instructions for main results in both the original article and the supplemental materials.

Data and Directories

We would suggest to set up the same directories as the *Two-Wing* paper repository at https://github.com/YushiFT/two-wing-mic. Open this markdown file under `your/path/to/docs`. Unzip the two data sets under their original directory, which include

- HZ data: https://github.com/YushiFT/two-wing-mic/tree/main/data/HZ/TwoWing_hz_asv_2018.txt.zip
- AO data: https://github.com/YushiFT/two-wing-mic/tree/main/data/AO/TwoWing_ao_asv.txt.zip

Packages

Main package

Our analysis relies on the `R` package `PMCosm` (Version 0.1.5) developed by our lab. Software handbook and quick instructions about using `PMCosm` are available at https://github.com/YushiFT/PMCosm. We first install and load the main package `PMCosm`.

```
# change to path/to/docs
# setwd(your/path/to/docs)
install.packages("devtools")
devtools::install_github("YushiFT/PMCosm")
```

```
library(PMCosm)
```

Other packages

```
library(ggplot2)       # for generating plots
library(latex2exp)     # for plot text latex
```

```
library(cowplot)        # for merging plots
library(gridExtra)      # for griding plots
library(grid)           # for griding plots
library(stringr)        # for uppercase first letter
library(ggh4x)          # for grid plot with free axis
library(vegan)          # for example data in pcoa analysis
```

Community Structure Inference

Estimate MLE trio

This step can take several hours. Cluster computing recommended.

```
# hz data
mic <- read.table(file='../data/HZ/hz_asv_2018.txt',
                  header=TRUE,row.names=1) # 51441 taxa in total
# construct column id
bay_loci <- c('HB1.1','HB1.2','HB1.3','HB2.1','HB2.2','HB2.3',
              'HB3.1','HB3.2','HB3.3','HB4.1','HB4.2','HB4.3',
              'HB5.1','HB5.2','HB5.3','HB6.1','HB6.2','HB6.3',
              'HB7.1','HB7.2','HB7.3','HB8.1','HB8.2','HB8.3',
              'HB9.1','HB9.2','HB9.3','HB10.1','HB10.2','HB10.3')
era_sy_loci <- c('SY1.1','SY1.2','SY1.3','SY2.1','SY2.2','SY2.3',
                 'SY3.1','SY3.2','SY3.3','SY4.1','SY4.2','SY4.3',
                 'SY5.1','SY5.2','SY5.3','SY6.1','SY6.2','SY6.3')
era_jx_loci <- c('JX1.1','JX1.2','JX1.3','JX2.1','JX2.2','JX2.3',
                 'JX3.1','JX3.2','JX3.3','JX4.1','JX4.2','JX4.3',
                 'JX5.1','JX5.2','JX5.3','JX6.1','JX6.2','JX6.3')

# for hz bay
mic_bay <- mic[,bay_loci] # 51441 taxa
# filter records with zero counts across all sample sites
mic_bay <- mic_bay[rowSums(mic_bay)>0,] # 24383 taxa
# save log records to ../output/HZ
sink(file='../output/HZ/mle_trio_trace_bay.txt')
# estimate mle trio
param_trio <- calc_mle_trio(mic_bay, n_sample=10, replicates=3)
# end log file
sink()
save(param_trio, file='../output/HZ/param_trio_bay.RData')

# for hz era
mic_era <- mic[,c(era_sy_loci,era_jx_loci)] # 51441 taxa
# filter records with zero counts across all sample sites
mic_era <- mic_era[rowSums(mic_era)>0,] # 33606 taxa
# save log records to ../output/HZ
sink(file='../output/HZ/mle_trio_trace_era.txt')
# estimate mle trio
param_trio <- calc_mle_trio(mic_era, n_sample=12, replicates=3)
# end log file
sink()
save(param_trio, file='../output/HZ/param_trio_era.RData')
```

```r
# ao data
# import the whole data set
mic <- read.table('../data/AO/ao_asv.txt')
# industrial factory names
id_dye <- c('LS','SF','CZ','BA','YF')
id_med <- c('GB','YTSW','XHC','ZC')
id_pes <- c('YT','YN')
lis_ind <- list(id_dye, id_med, id_pes)
names(lis_ind) <- c('Dye','Medicine','Pesticide')
# ao procedures id
id_inf <- c('Inf1','Inf2','Inf3')
id_axi <- c('Ax1','Ax2','Ax3')
id_oxi <- c('Ox1','Ox2','Ox3')
id_eff <- c('Eff1','Eff2','Eff3')
lis_pro <- list(id_inf, id_axi, id_oxi, id_eff)
names(lis_pro) <- c('Influent','Anoxic','Oxic','Effluent')

# ao influent
region <- 1
id_col <- c()
# extract samples in target procedure/region
for(i in 1:length(lis_ind)){
    id_ind <- lis_ind[[i]]
    id_pro <- lis_pro[[region]]
    id_col <- c(id_col,
                paste0(rep(id_ind, each=length(id_pro)),'_',rep(id_pro, length(id_ind))))
}
mic_sub <- mic[,id_col] # 111981 taxa
mic_sub <- mic_sub[rowSums(mic_sub)>0,] # 16763 taxa
# save log records to ../output/AO
sink(file='../output/AO/mle_trio_trace_influent.txt')
# estimate mle trio
param_trio <- calc_mle_trio(mic_sub, n_sample=11, replicates=3)
# end log file
sink()
save(param_trio, file='../output/AO/param_trio_influent.RData')


# ao anoxic
region <- 2
id_col <- c()
# extract samples in target procedure/region
for(i in 1:length(lis_ind)){
    id_ind <- lis_ind[[i]]
    id_pro <- lis_pro[[region]]
    id_col <- c(id_col,
                paste0(rep(id_ind, each=length(id_pro)),'_',rep(id_pro, length(id_ind))))
}
mic_sub <- mic[,id_col] # 111981 taxa
mic_sub <- mic_sub[rowSums(mic_sub)>0,] # 31922 taxa
# save log records to ../output/AO
sink(file='../output/AO/mle_trio_trace_anoxic.txt')
```

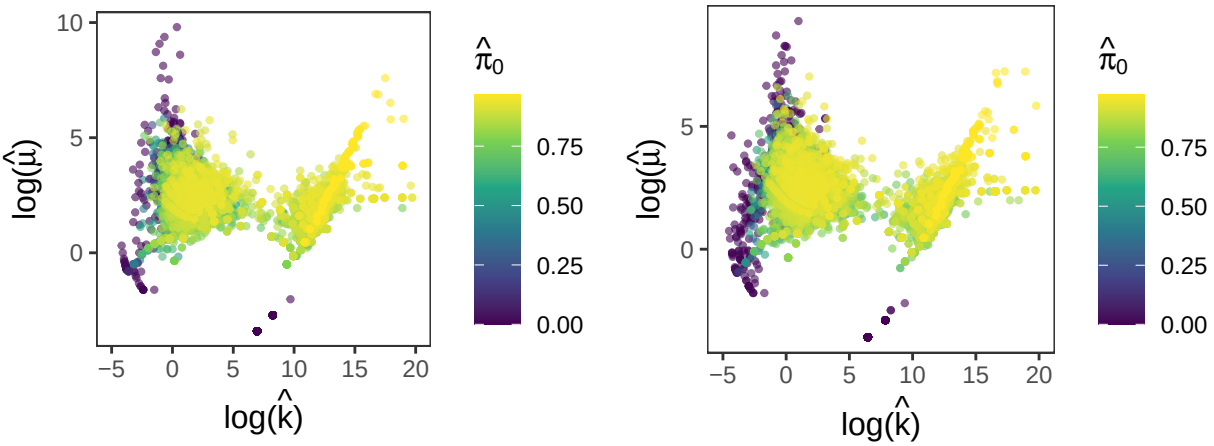
```
# estimate mle trio
param_trio <- calc_mle_trio(mic_sub, n_sample=11, replicates=3)
# end log file
sink()
save(param_trio, file='../output/AO/param_trio_anoxic.RData')


# ao oxic
region <- 3
id_col <- c()
# extract samples in target procedure/region
for(i in 1:length(lis_ind)){
    id_ind <- lis_ind[[i]]
    id_pro <- lis_pro[[region]]
    id_col <- c(id_col,
                paste0(rep(id_ind, each=length(id_pro)),'_',rep(id_pro, length(id_ind))))
}
mic_sub <- mic[,id_col] # 111981 taxa
mic_sub <- mic_sub[rowSums(mic_sub)>0,] # 36071 taxa
# save log records to ../output/AO
sink(file='../output/AO/mle_trio_trace_oxic.txt')
# estimate mle trio
param_trio <- calc_mle_trio(mic_sub, n_sample=11, replicates=3)
# end log file
sink()
save(param_trio, file='../output/AO/param_trio_oxic.RData')


# ao effluent
region <- 4
id_col <- c()
# extract samples in target procedure/region
for(i in 1:length(lis_ind)){
    id_ind <- lis_ind[[i]]
    id_pro <- lis_pro[[region]]
    id_col <- c(id_col,
                paste0(rep(id_ind, each=length(id_pro)),'_',rep(id_pro, length(id_ind))))
}
mic_sub <- mic[,id_col] # 111981 taxa
mic_sub <- mic_sub[rowSums(mic_sub)>0,] #  38330 taxa
# save log records to ../output/AO
sink(file='../output/AO/mle_trio_trace_effluent.txt')
# estimate mle trio
param_trio <- calc_mle_trio(mic_sub, n_sample=11, replicates=3)
# end log file
sink()
save(param_trio, file='../output/AO/param_trio_effluent.RData')
```

Visualize the *Two-Wing* admixed structure by MLE trio estimates

HZ data

The left panel below is HZ Bay, right panel is HZ ERA.

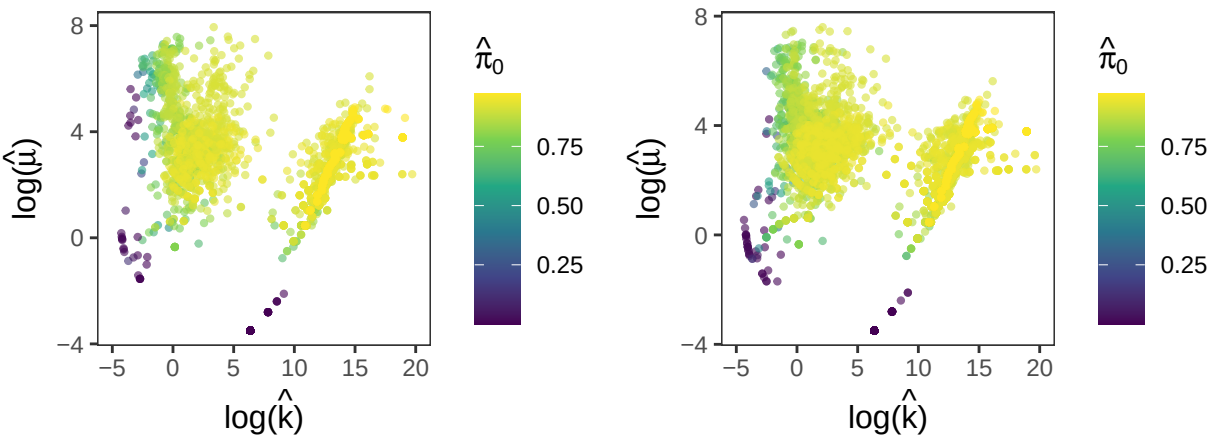
```
load('../output/HZ/param_trio_bay.RData')
g1 <- plot_trio(param_trio, zoom_in=TRUE)

load('../output/HZ/param_trio_era.RData')
g2 <- plot_trio(param_trio, zoom_in=TRUE)

plot_grid(g1, g2, nrow=1, align='h')
```



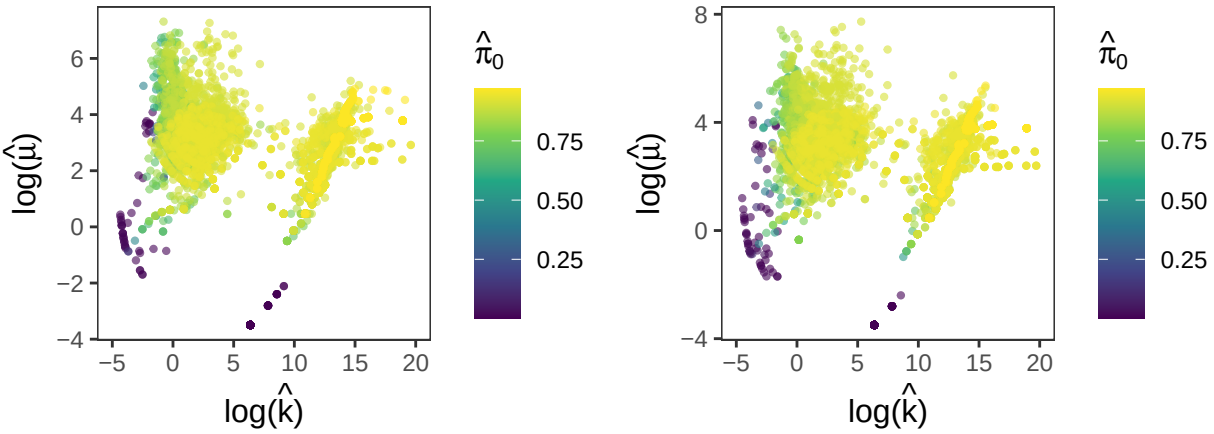
AO data

From left top to right bottom, the order is *Influent*, *Anoxic*, *Oxic*, and *Effluent*.

```
load('../output/AO/param_trio_influent.RData')
g1 <- plot_trio(param_trio, zoom_in=TRUE)

load('../output/AO/param_trio_anoxic.RData')
g2 <- plot_trio(param_trio, zoom_in=TRUE)

load('../output/AO/param_trio_oxic.RData')
g3 <- plot_trio(param_trio, zoom_in=TRUE)

load('../output/AO/param_trio_effluent.RData')
g4 <- plot_trio(param_trio, zoom_in=TRUE)

plot_grid(g1, g2, nrow=1, align='h')
```




```
plot_grid(g3, g4, nrow=1, align='h')
```



Dispersal Vanguards and Laggards

Clarify the model-driven testable boundary

HZ data

```
# hz data
mic <- read.table(file='../data/HZ/hz_asv_2018.txt',
                  header=TRUE,row.names=1) # 51441 taxa in total
# construct column id
bay_loci <- c('HB1.1','HB1.2','HB1.3','HB2.1','HB2.2','HB2.3',
              'HB3.1','HB3.2','HB3.3','HB4.1','HB4.2','HB4.3',
              'HB5.1','HB5.2','HB5.3','HB6.1','HB6.2','HB6.3',
              'HB7.1','HB7.2','HB7.3','HB8.1','HB8.2','HB8.3',
              'HB9.1','HB9.2','HB9.3','HB10.1','HB10.2','HB10.3')
era_sy_loci <- c('SY1.1','SY1.2','SY1.3','SY2.1','SY2.2','SY2.3',
                 'SY3.1','SY3.2','SY3.3','SY4.1','SY4.2','SY4.3',
                 'SY5.1','SY5.2','SY5.3','SY6.1','SY6.2','SY6.3')
era_jx_loci <- c('JX1.1','JX1.2','JX1.3','JX2.1','JX2.2','JX2.3',
                 'JX3.1','JX3.2','JX3.3','JX4.1','JX4.2','JX4.3',
                 'JX5.1','JX5.2','JX5.3','JX6.1','JX6.2','JX6.3')
```

```
# for hz bay
mic_bay <- mic[,bay_loci] # 51441 taxa
# filter records with zero counts across all sample sites
mic_bay <- mic_bay[rowSums(mic_bay)>0,] # 24383 taxa
load('../output/HZ/param_trio_bay.RData')
id_vag_lag <- classify_vag_lag(mic_bay, param_trio)
save(id_vag_lag, file='../output/HZ/id_vag_lag_bay.RData')
# hz era
# for hz era
mic_era <- mic[,c(era_sy_loci,era_jx_loci)] # 51441 taxa
# filter records with zero counts across all sample sites
mic_era <- mic_era[rowSums(mic_era)>0,] # 33606 taxa
load('../output/HZ/param_trio_era.RData')
id_vag_lag <- classify_vag_lag(mic_era, param_trio)
save(id_vag_lag, file='../output/HZ/id_vag_lag_era.RData')
```

AO data

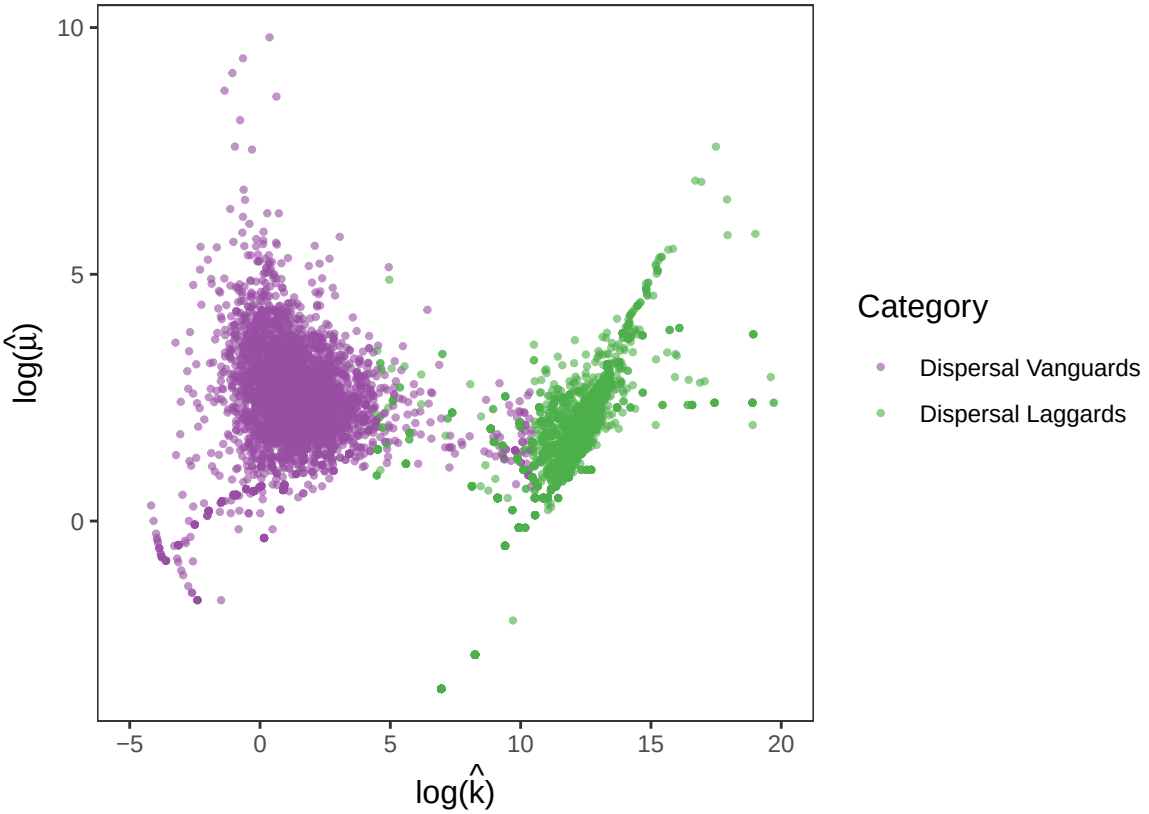
```
# ao data
# import the whole data set
mic <- read.table('../data/AO/ao_asv.txt')
# industrial factory names
id_dye <- c('LS','SF','CZ','BA','YF')
id_med <- c('GB','YTSW','XHC','ZC')
id_pes <- c('YT','YN')
lis_ind <- list(id_dye, id_med, id_pes)
names(lis_ind) <- c('Dye','Medicine','Pesticide')
# ao procedures id
id_inf <- c('Inf1','Inf2','Inf3')
id_axi <- c('Ax1','Ax2','Ax3')
id_oxi <- c('Ox1','Ox2','Ox3')
id_eff <- c('Eff1','Eff2','Eff3')
lis_pro <- list(id_inf, id_axi, id_oxi, id_eff)
names(lis_pro) <- c('influent','anoxic','oxic','effluent')

# ao influent
for(region in 1:4){
  id_col <- c()
  for(i in 1:length(lis_ind)){
    id_ind <- lis_ind[[i]]
    id_pro <- lis_pro[[region]]
    id_col <- c(id_col,
                paste0(rep(id_ind, each=length(id_pro)),'_',rep(id_pro, length(id_ind))))
  }
  mic_sub <- mic[,id_col]
  mic_sub <- mic_sub[rowSums(mic_sub)>0,]

  load(paste0('../output/AO/param_trio_',names(lis_pro)[region],'.RData'))
  id_vag_lag <- classify_vag_lag(mic_sub, param_trio)
  save(id_vag_lag, file=paste0('../output/AO/id_vag_lag_',names(lis_pro)[region],'.RData'))

}
```

Visualize the community structure as the admixture of dispersal vanguards and laggards

HZ data

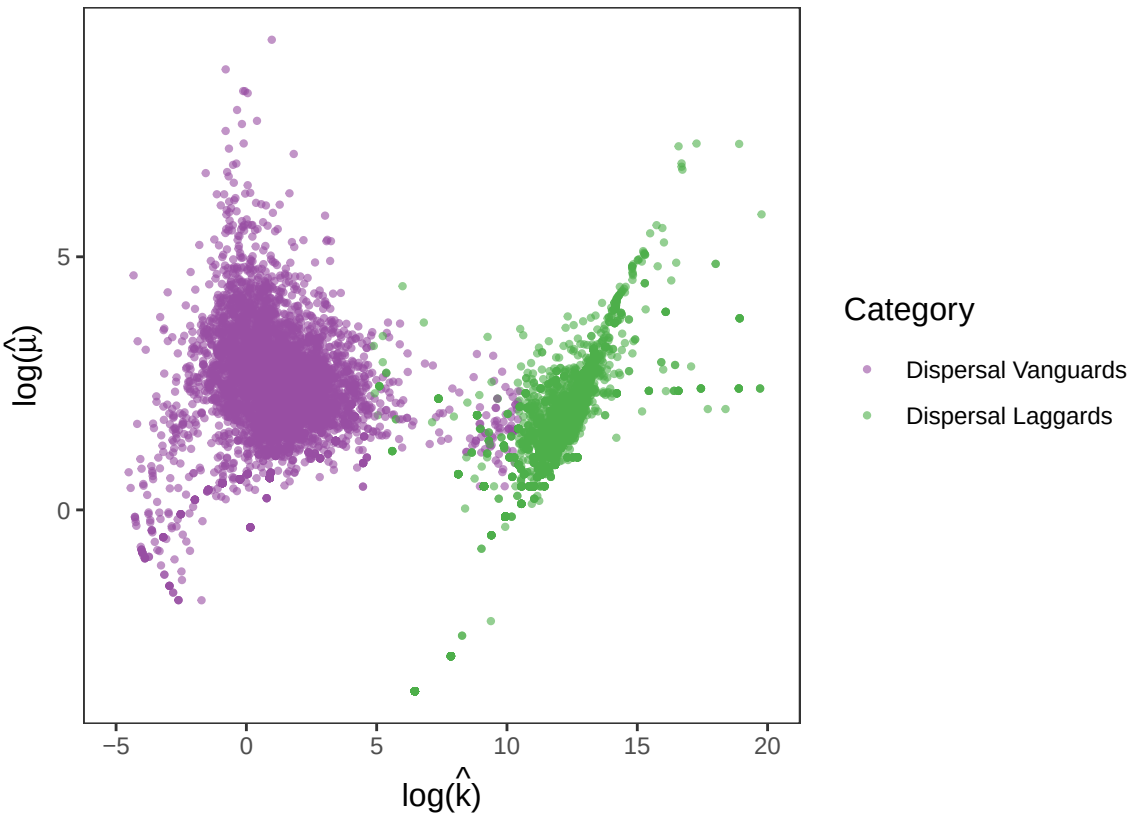
HZ Bay

```
load('../output/HZ/param_trio_bay.RData')
load('../output/HZ/id_vag_lag_bay.RData')
plot_vag_lag(param_trio, id_vag_lag, zoom_in=TRUE)
```



HZ ERA

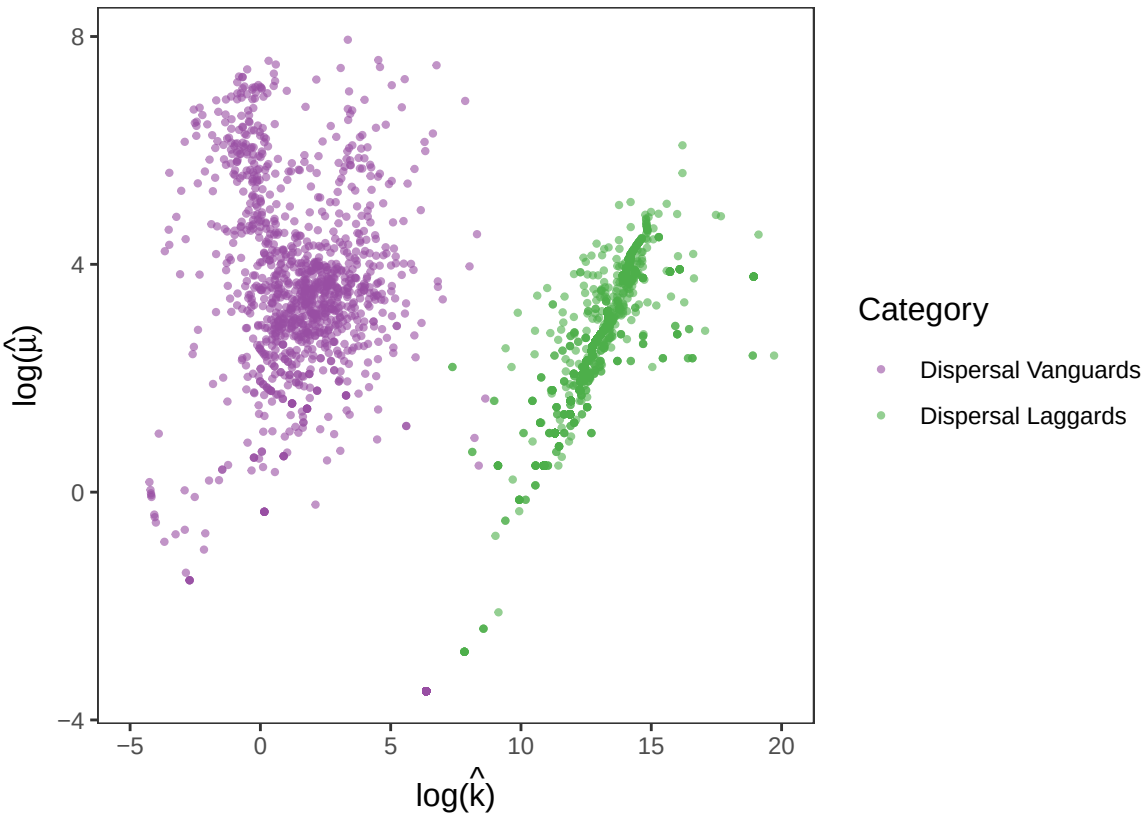
```
load('../output/HZ/param_trio_era.RData')
load('../output/HZ/id_vag_lag_era.RData')
plot_vag_lag(param_trio, id_vag_lag, zoom_in=TRUE)
```



AO data

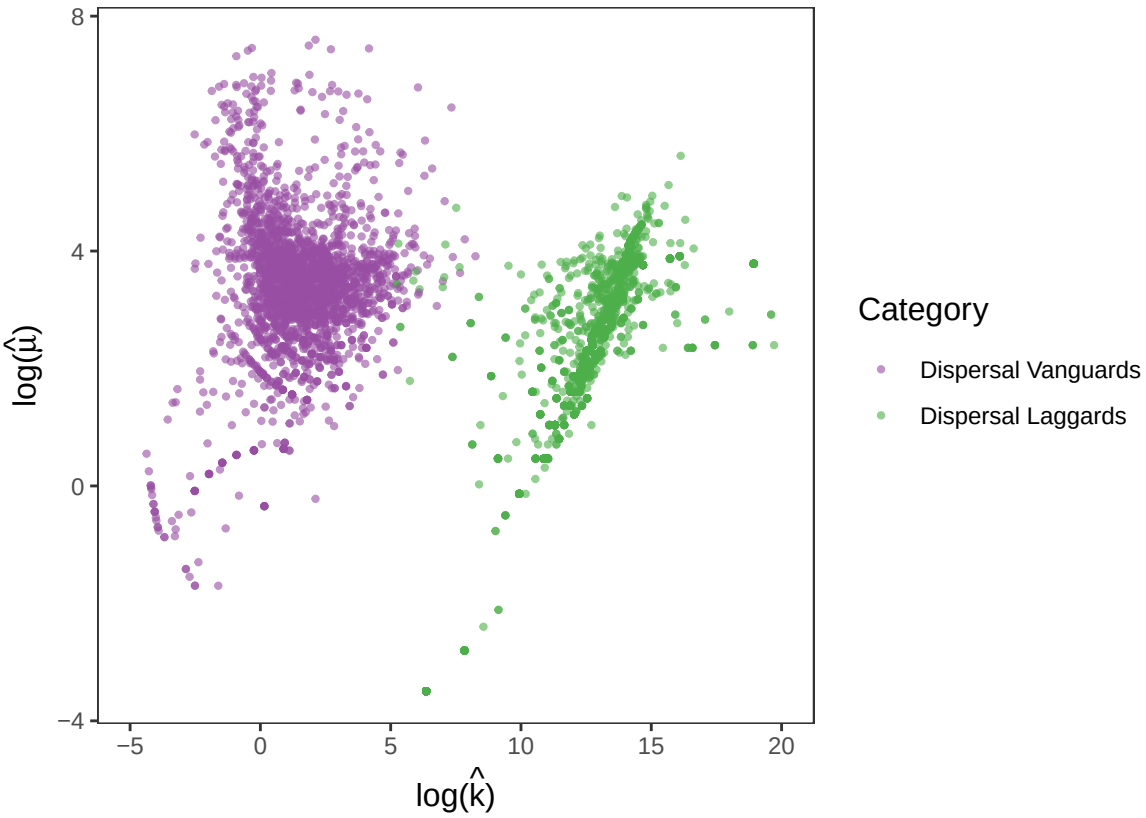
AO *Influent*

```
load('../output/AO/param_trio_influent.RData')
load('../output/AO/id_vag_lag_influent.RData')
plot_vag_lag(param_trio, id_vag_lag, zoom_in=TRUE)
```



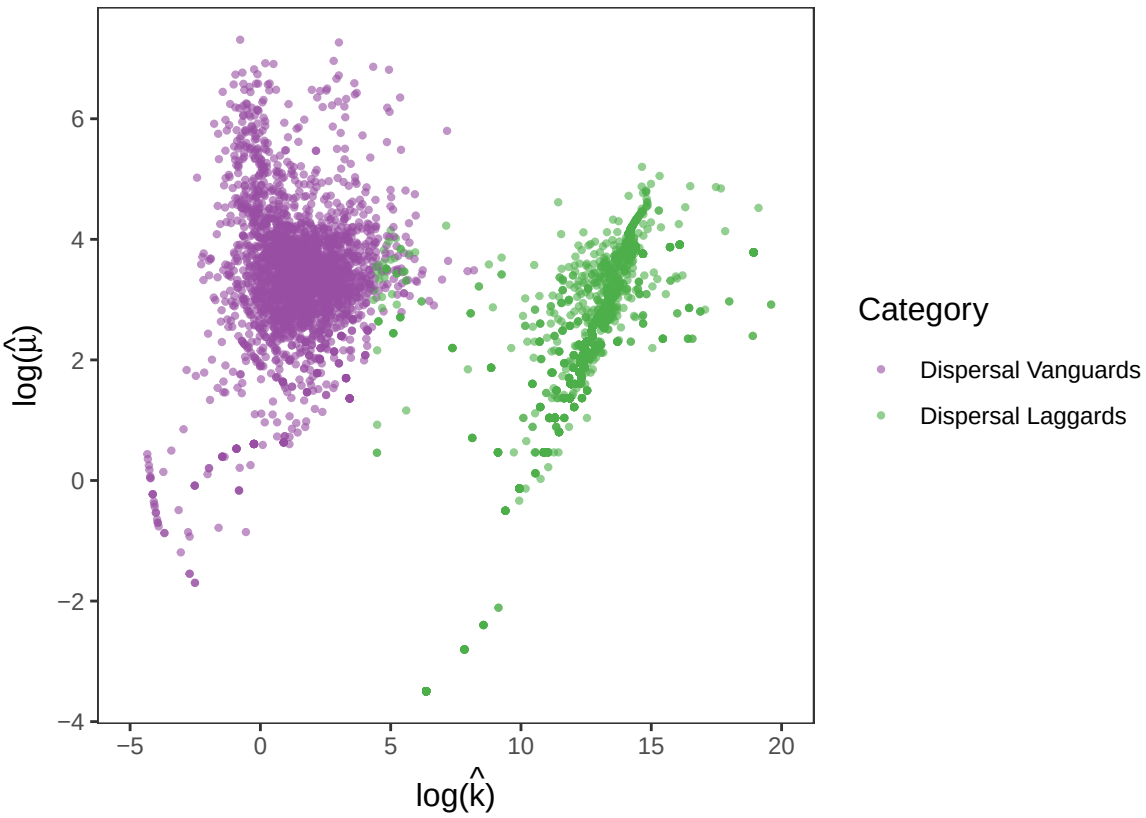
AO *Anoxic*

```
load('../output/AO/param_trio_anoxic.RData')
load('../output/AO/id_vag_lag_anoxic.RData')
plot_vag_lag(param_trio, id_vag_lag, zoom_in=TRUE)
```



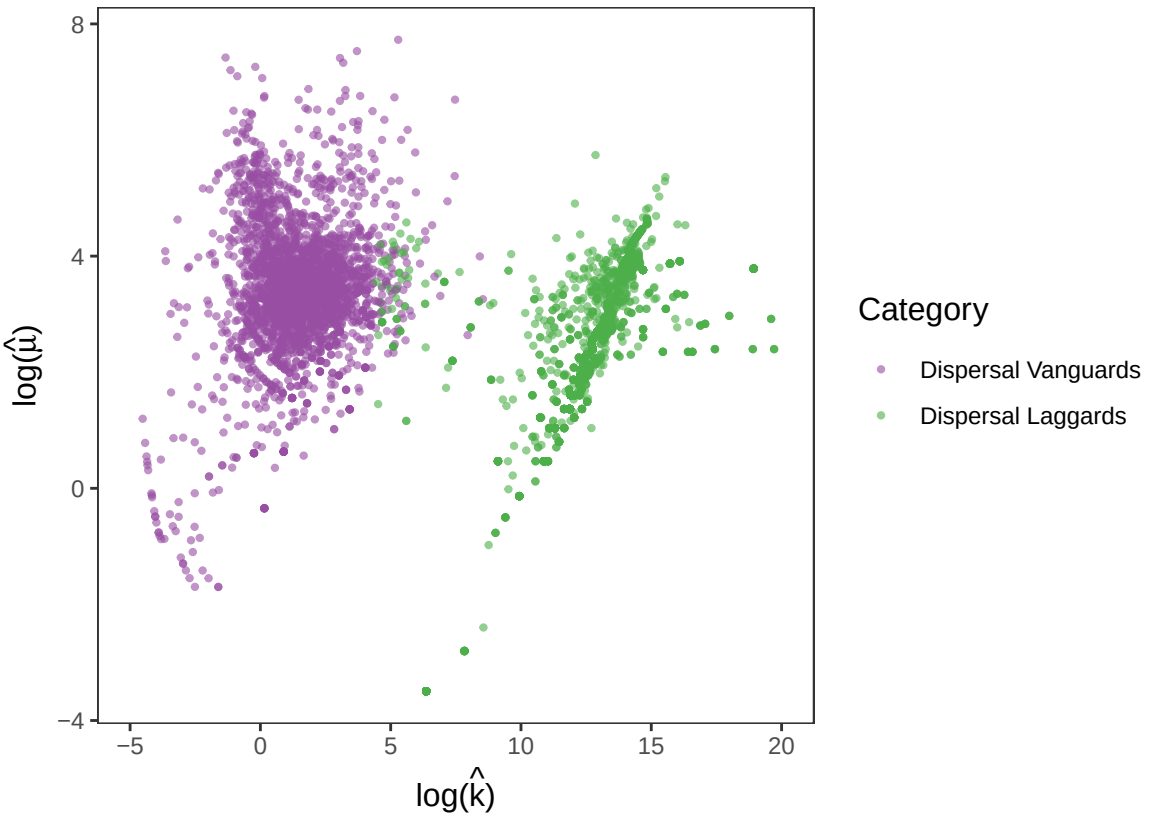
AO *Oxic*

```
load('../output/AO/param_trio_oxic.RData')
load('../output/AO/id_vag_lag_oxic.RData')
plot_vag_lag(param_trio, id_vag_lag, zoom_in=TRUE)
```



AO *Effluent*

```
load('../output/AO/param_trio_effluent.RData')
load('../output/AO/id_vag_lag_effluent.RData')
plot_vag_lag(param_trio, id_vag_lag, zoom_in=TRUE)
```



Abundant and Rare Biospheres in the *Two-Wing* Admixed Structure

The *Two-Wing* admixed structure is a generalization and extension of relative-abundance-based categorizing methods. The former one is more sufficient since it delivers all information that abundant and rare biospheres can provide while the *Two-Wing* admixed structure further provides unbiased estimation of the over-dispersion, thereby characterizing species dispersal.

Classify abundant and rare taxa by following relative-abundance-based methods

Categorize taxa for HZ data

```
# hz data
mic <- read.table(file='../data/HZ/hz_asv_2018.txt',
                  header=TRUE,row.names=1) # 51441 taxa in total
# construct column id
bay_loci <- c('HB1.1','HB1.2','HB1.3','HB2.1','HB2.2','HB2.3',
              'HB3.1','HB3.2','HB3.3','HB4.1','HB4.2','HB4.3',
              'HB5.1','HB5.2','HB5.3','HB6.1','HB6.2','HB6.3',
              'HB7.1','HB7.2','HB7.3','HB8.1','HB8.2','HB8.3',
              'HB9.1','HB9.2','HB9.3','HB10.1','HB10.2','HB10.3')
era_sy_loci <- c('SY1.1','SY1.2','SY1.3','SY2.1','SY2.2','SY2.3',
                 'SY3.1','SY3.2','SY3.3','SY4.1','SY4.2','SY4.3',
                 'SY5.1','SY5.2','SY5.3','SY6.1','SY6.2','SY6.3')
```

```
era_jx_loci <- c('JX1.1','JX1.2','JX1.3','JX2.1','JX2.2','JX2.3',
                 'JX3.1','JX3.2','JX3.3','JX4.1','JX4.2','JX4.3',
                 'JX5.1','JX5.2','JX5.3','JX6.1','JX6.2','JX6.3')

# for hz bay
mic_bay <- mic[,bay_loci] # 51441 taxa
# filter records with zero counts across all sample sites
mic_bay <- mic_bay[rowSums(mic_bay)>0,] # 24383 taxa
# method 0, a, b, and c
taxa_cat_0_h <- classify_taxa(mic_bay, t_lower=1e-3, method='0')
taxa_cat_0_l <- classify_taxa(mic_bay, t_lower=1e-4, method='0')
taxa_cat_a <- classify_taxa(mic_bay, method='a')
taxa_cat_b <- classify_taxa(mic_bay, method='b')
taxa_cat_c <- classify_taxa(mic_bay, method='c')
save(taxa_cat_0_h, taxa_cat_0_l, taxa_cat_a, taxa_cat_b, taxa_cat_c,
     file='../output/HZ/abundant_rare_taxa_bay.RData')

# for hz era
mic_era <- mic[,c(era_sy_loci,era_jx_loci)] # 51441 taxa
# filter records with zero counts across all sample sites
mic_era <- mic_era[rowSums(mic_era)>0,] # 33606 taxa
# method 0, a, b, and c
taxa_cat_0_h <- classify_taxa(mic_era, t_lower=1e-3, method='0')
taxa_cat_0_l <- classify_taxa(mic_era, t_lower=1e-4, method='0')
taxa_cat_a <- classify_taxa(mic_era, method='a')
taxa_cat_b <- classify_taxa(mic_era, method='b')
taxa_cat_c <- classify_taxa(mic_era, method='c')
save(taxa_cat_0_h, taxa_cat_0_l, taxa_cat_a, taxa_cat_b, taxa_cat_c,
     file='../output/HZ/abundant_rare_taxa_era.RData')
```

Categorize taxa for AO data

```
mic <- read.table('../data/AO/ao_asv.txt')
# industrial factory names
id_dye <- c('LS','SF','CZ','BA','YF')
id_med <- c('GB','YTSW','XHC','ZC')
id_pes <- c('YT','YN')
lis_ind <- list(id_dye, id_med, id_pes)
names(lis_ind) <- c('Dye','Medicine','Pesticide')
# ao procedures id
id_inf <- c('Inf1','Inf2','Inf3')
id_axi <- c('Ax1','Ax2','Ax3')
id_oxi <- c('Ox1','Ox2','Ox3')
id_eff <- c('Eff1','Eff2','Eff3')
lis_pro <- list(id_inf, id_axi, id_oxi, id_eff)
names(lis_pro) <- c('influent','anoxic','oxic','effluent')

for(region in 1:4){
  id_col <- c()
  for(i in 1:length(lis_ind)){
    id_ind <- lis_ind[[i]]
    id_pro <- lis_pro[[region]]
```

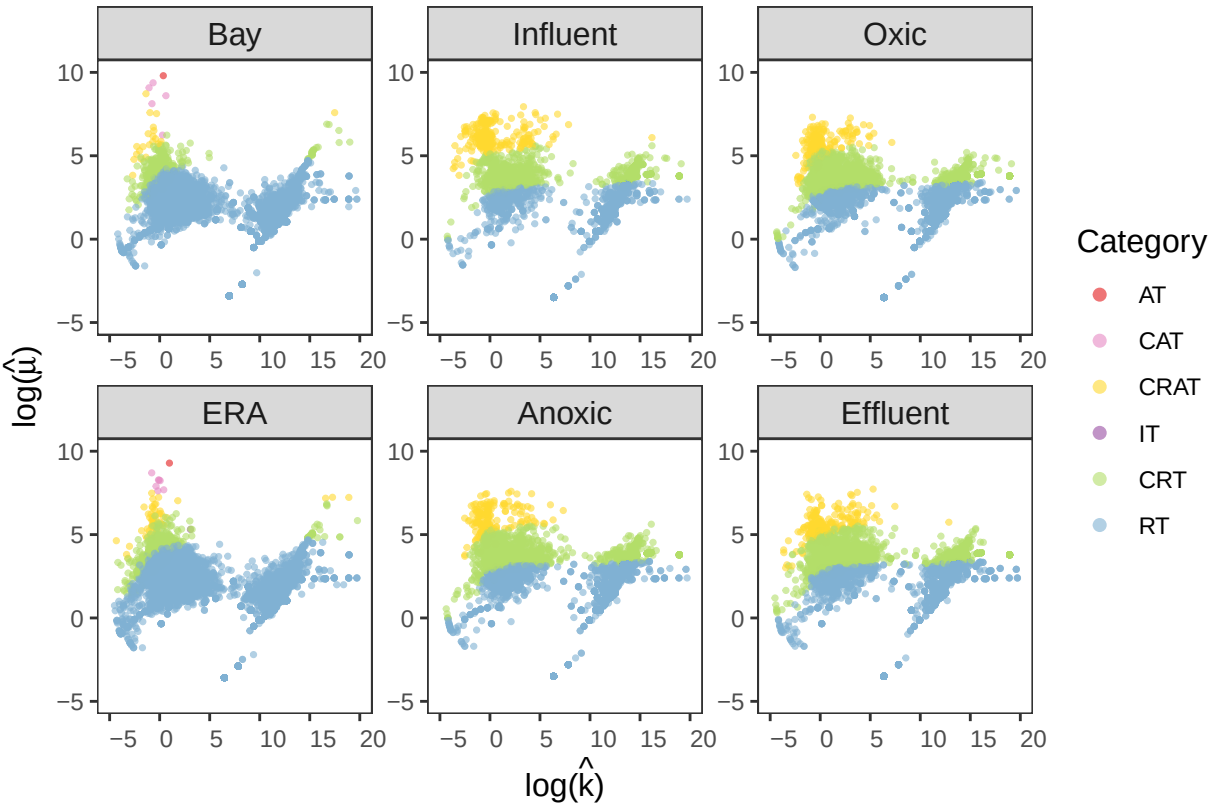
```
    id_col <- c(id_col,
                paste0(rep(id_ind, each=length(id_pro)),'_',rep(id_pro, length(id_ind))))
  }
  mic_sub <- mic[,id_col]
  mic_sub <- mic_sub[rowSums(mic_sub)>0,]

  # method 0, a, b, and c
  taxa_cat_0_h <- classify_taxa(mic_sub, t_lower=1e-3, method='0')
  taxa_cat_0_l <- classify_taxa(mic_sub, t_lower=1e-4, method='0')
  taxa_cat_a <- classify_taxa(mic_sub, method='a')
  taxa_cat_b <- classify_taxa(mic_sub, method='b')
  taxa_cat_c <- classify_taxa(mic_sub, method='c')
  save(taxa_cat_0_h, taxa_cat_0_l, taxa_cat_a, taxa_cat_b, taxa_cat_c,
       file=paste0('../output/AO/abundant_rare_taxa_',names(lis_pro)[region],'.RData'))
}
```

Visualize abundant and rare biospheres in the *Two-Wing* admixed structure

Abundant and rare taxa are defined by *Method 0* with a higher threshold 0.01% for rare taxa

```
# load parameters mle
vec_process <- c('influent','anoxic','oxic','effluent')
param_plt <- data.frame()
for(item in vec_process){
  load(paste0('../output/AO/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/AO/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$AT, 'AT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$CRT, 'CRT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$CAT, 'CAT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$CRAT, 'CRAT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

vec_region <- c('bay','era')
for(item in vec_region){
  load(paste0('../output/HZ/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/HZ/abundant_rare_taxa_',item,'.RData'))
```

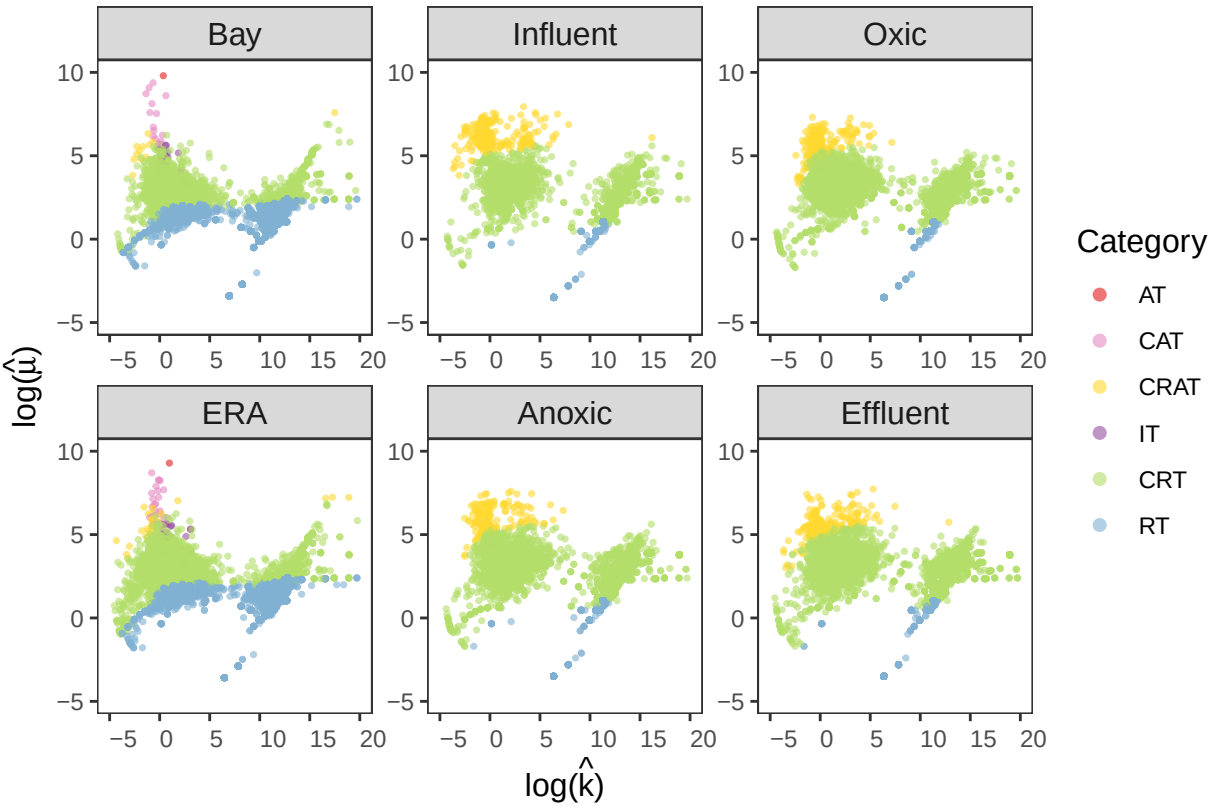
```
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$AT, 'AT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$CRT, 'CRT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$CAT, 'CAT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_h$CRAT, 'CRAT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

rownames(param_plt) <- c(1:nrow(param_plt))
param_plt$Procedure <- ifelse(param_plt$Procedure=='Era','ERA',param_plt$Procedure)

param_plt$Procedure <- factor(param_plt$Procedure,
                              levels=c('Bay','Influent','Oxic',
                                       'ERA','Anoxic','Effluent'))
param_plt$Category <- factor(param_plt$Category,
                             levels=c('AT','CAT','CRAT',
                                      'IT','CRT','RT'))

ggplot() +
  geom_point(data=param_plt,
             aes(x=log(k), y=log(mu), color=Category), alpha=0.6,size=0.6) +
  ylab(TeX('\\log{(\\hat{\\mu})}')) +
  xlab(TeX('\\log{(\\hat{k})}')) +
  ggtitle(TeX('')) +
  xlim(-5,20) +
  ylim(-5,10) +
  theme_bw() +
  theme(aspect.ratio=1) +
  facet_wrap(Procedure~., scale='free') +
  scale_colour_manual(values = c('#e41a1c','#e78ac3','#ffd92f',
                                 '#984ea3','#b3de69','#80b1d3')) +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12),
        axis.text.x = element_text(size = 9, vjust = 0.5, hjust = 0),
        axis.text.y = element_text(size=9),
        strip.text = element_text(size = 12),
        legend.title = element_text(color = "black", size = 12),
        legend.text = element_text(color = "black", size = 9)) +
  guides(color = guide_legend(override.aes = list(size = 1.8)))
```



Abundant and rare taxa are defined by *Method 0* with a lower threshold 0.001% to identify rare taxa

```
# load parameters mle
vec_process <- c('influent','anoxic','oxic','effluent')
param_plt <- data.frame()
for(item in vec_process){
  load(paste0('../output/AO/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/AO/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$AT, 'AT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$CRT, 'CRT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$CAT, 'CAT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$CRAT, 'CRAT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

vec_region <- c('bay','era')
for(item in vec_region){
  load(paste0('../output/HZ/param_trio_',item,'.RData'))
```

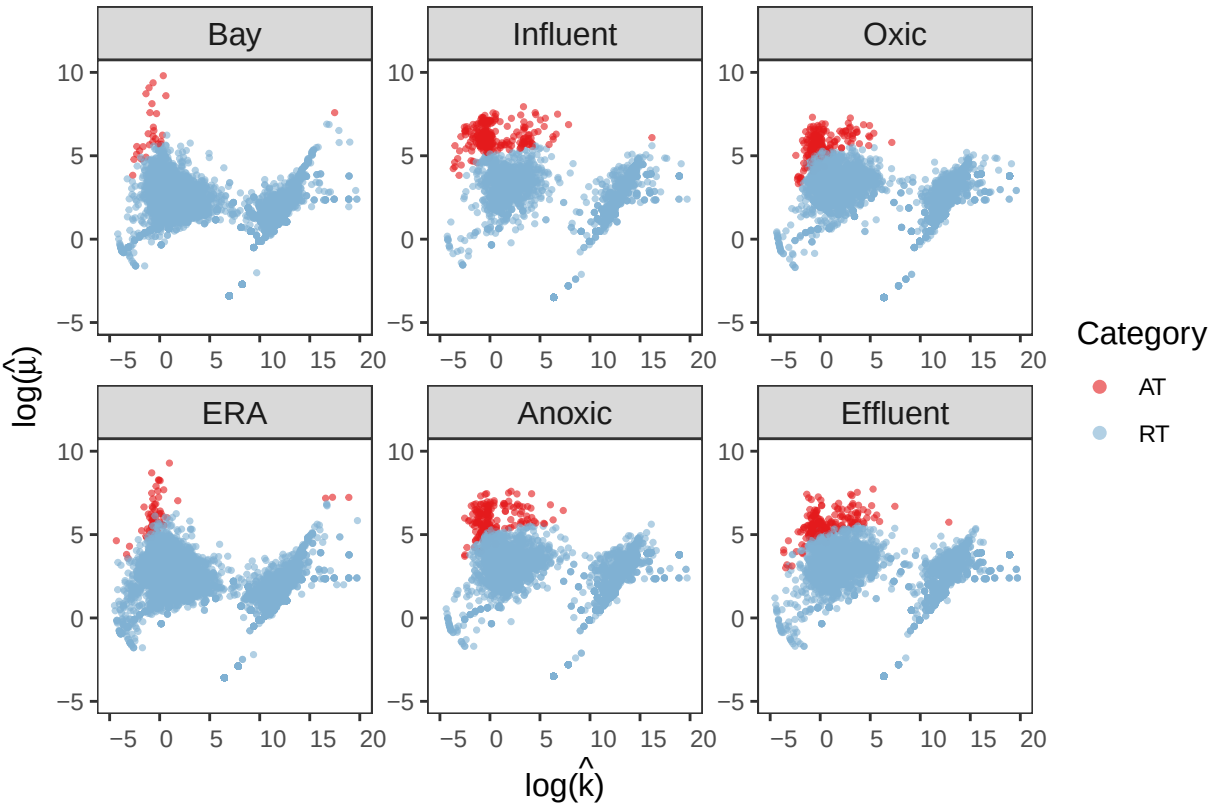
```
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/HZ/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$AT, 'AT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$CRT, 'CRT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$CAT, 'CAT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_0_1$CRAT, 'CRAT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

rownames(param_plt) <- c(1:nrow(param_plt))
param_plt$Procedure <- ifelse(param_plt$Procedure=='Era','ERA',param_plt$Procedure)

param_plt$Procedure <- factor(param_plt$Procedure,
                              levels=c('Bay','Influent','Oxic',
                                       'ERA','Anoxic','Effluent'))
param_plt$Category <- factor(param_plt$Category,
                             levels=c('AT','CAT','CRAT',
                                      'IT','CRT','RT'))

ggplot() +
  geom_point(data=param_plt,
             aes(x=log(k), y=log(mu), color=Category), alpha=0.6,size=0.6) +
  ylab(TeX('\\log{(\\hat{\\mu})}')) +
  xlab(TeX('\\log{(\\hat{k})}')) +
  ggtitle(TeX('')) +
  xlim(-5,20) +
  ylim(-5,10) +
  theme_bw() +
  theme(aspect.ratio=1) +
  facet_wrap(Procedure~., scale='free') +
  scale_colour_manual(values = c('#e41a1c','#e78ac3','#ffd92f',
                                 '#984ea3','#b3de69','#80b1d3')) +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12),
        axis.text.x = element_text(size = 9, vjust = 0.5, hjust = 0),
        axis.text.y = element_text(size=9),
        strip.text = element_text(size = 12),
        legend.title = element_text(color = "black", size = 12),
        legend.text = element_text(color = "black", size = 9)) +
  guides(color = guide_legend(override.aes = list(size = 1.8)))
```



Abundant and rare taxa are defined by *Method a*

```
# load parameters mle
vec_process <- c('influent','anoxic','oxic','effluent')
param_plt <- data.frame()
for(item in vec_process){
  load(paste0('../output/AO/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/AO/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_a$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_a$AT, 'AT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

vec_region <- c('bay','era')
for(item in vec_region){
  load(paste0('../output/HZ/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
```

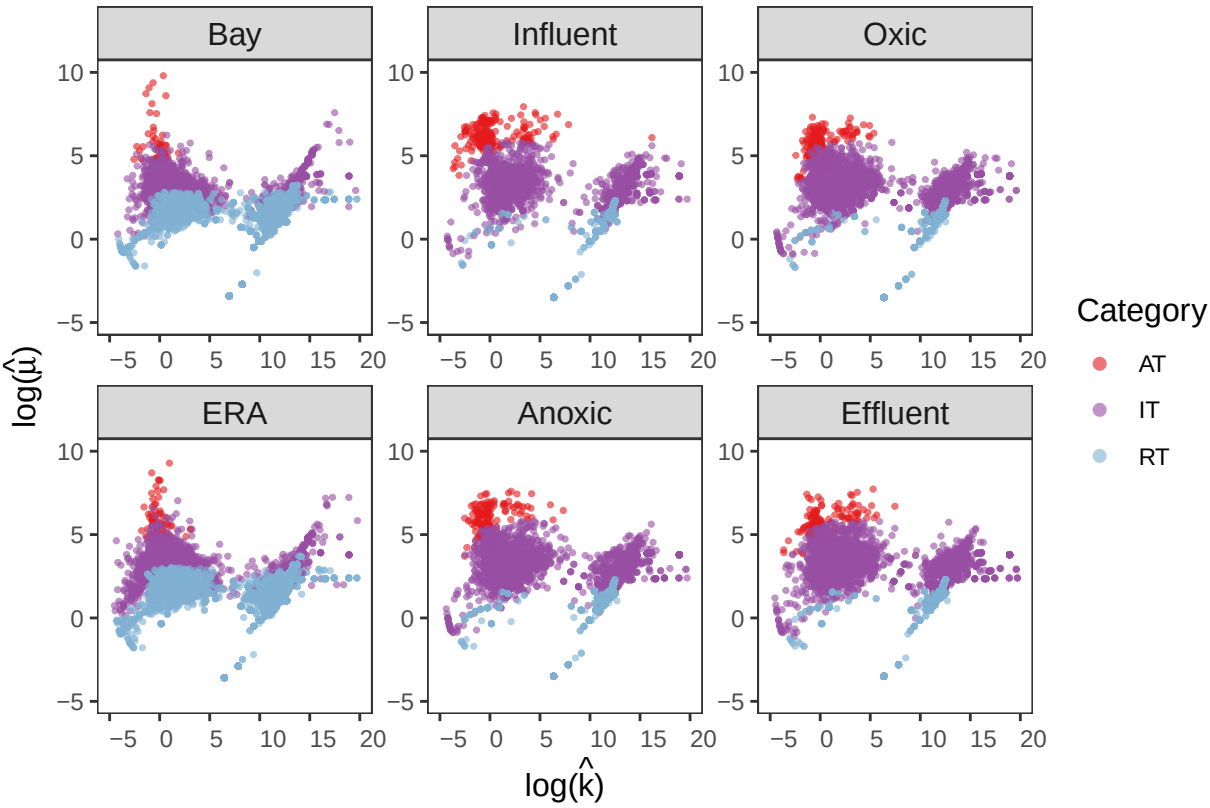
```
  # annotate taxa category
  load(paste0('../output/HZ/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_a$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_a$AT, 'AT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

rownames(param_plt) <- c(1:nrow(param_plt))
param_plt$Procedure <- ifelse(param_plt$Procedure=='Era','ERA',param_plt$Procedure)

param_plt$Procedure <- factor(param_plt$Procedure,
                              levels=c('Bay','Influent','Oxic',
                                       'ERA','Anoxic','Effluent'))
param_plt$Category <- factor(param_plt$Category,
                             levels=c('AT','RT'))

ggplot() +
  geom_point(data=param_plt,
             aes(x=log(k), y=log(mu), color=Category), alpha=0.6,size=0.6) +
  ylab(TeX('\\log{(\\hat{\\mu})}')) +
  xlab(TeX('\\log{(\\hat{k})}')) +
  ggtitle(TeX('')) +
  xlim(-5,20) +
  ylim(-5,10) +
  theme_bw() +
  theme(aspect.ratio=1) +
  facet_wrap(Procedure~., scale='free') +
  scale_colour_manual(values = c('#e41a1c','#80b1d3')) +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12),
        axis.text.x = element_text(size = 9, vjust = 0.5, hjust = 0),
        axis.text.y = element_text(size=9),
        strip.text = element_text(size = 12),
        legend.title = element_text(color = "black", size = 12),
        legend.text = element_text(color = "black", size = 9)) +
  guides(color = guide_legend(override.aes = list(size = 1.8)))
```



Abundant and rare taxa are defined by *Method b*

```
# load parameters mle
vec_process <- c('influent','anoxic','oxic','effluent')
param_plt <- data.frame()
for(item in vec_process){
  load(paste0('../output/AO/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/AO/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_b$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_b$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_b$AT, 'AT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

vec_region <- c('bay','era')
for(item in vec_region){
  load(paste0('../output/HZ/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
```

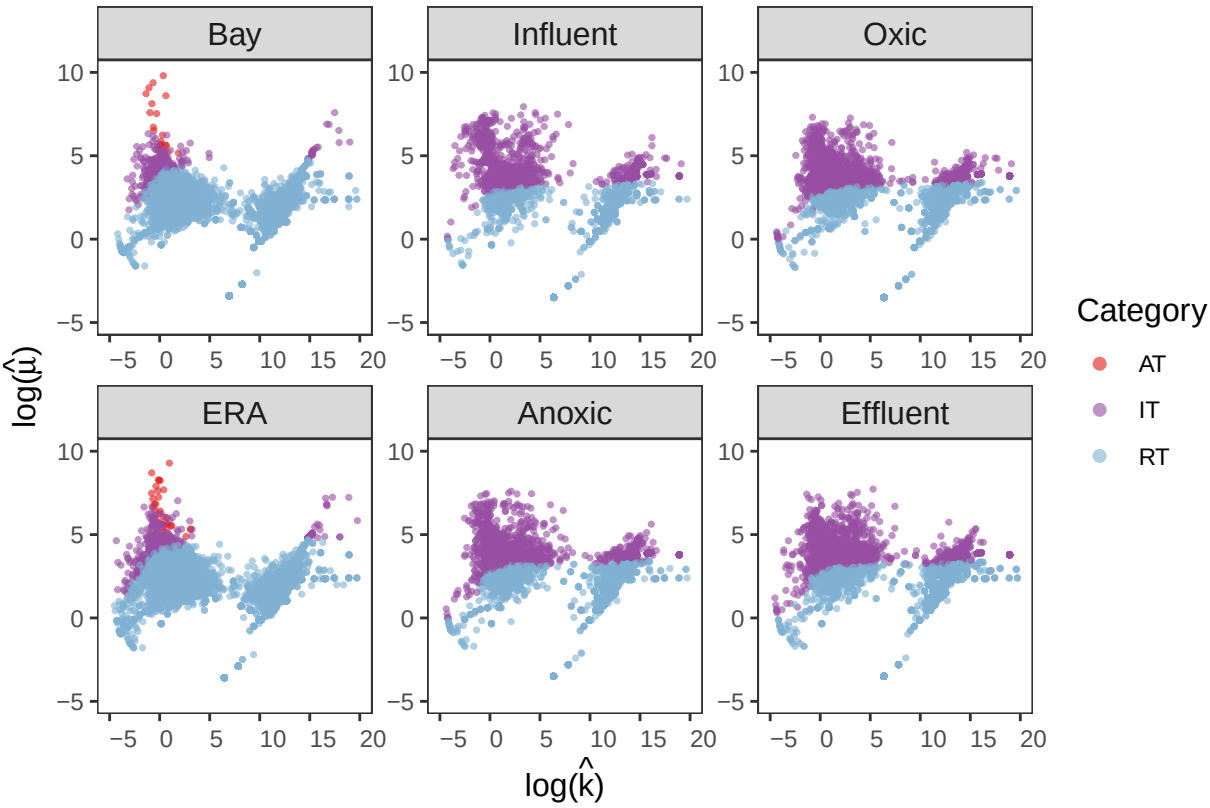
```
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/HZ/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_b$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_b$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_b$AT, 'AT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

rownames(param_plt) <- c(1:nrow(param_plt))
param_plt$Procedure <- ifelse(param_plt$Procedure=='Era','ERA',param_plt$Procedure)

param_plt$Procedure <- factor(param_plt$Procedure,
                              levels=c('Bay','Influent','Oxic',
                                       'ERA','Anoxic','Effluent'))
param_plt$Category <- factor(param_plt$Category,
                             levels=c('AT','IT','RT'))

ggplot() +
  geom_point(data=param_plt,
             aes(x=log(k), y=log(mu), color=Category), alpha=0.6,size=0.6) +
  ylab(TeX('\\log{(\\hat{\\mu})}')) +
  xlab(TeX('\\log{(\\hat{k})}')) +
  ggtitle(TeX('')) +
  xlim(-5,20) +
  ylim(-5,10) +
  theme_bw() +
  theme(aspect.ratio=1) +
  facet_wrap(Procedure~., scale='free') +
  scale_colour_manual(values = c('#e41a1c','#984ea3','#80b1d3')) +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12),
        axis.text.x = element_text(size = 9, vjust = 0.5, hjust = 0),
        axis.text.y = element_text(size=9),
        strip.text = element_text(size = 12),
        legend.title = element_text(color = "black", size = 12),
        legend.text = element_text(color = "black", size = 9)) +
  guides(color = guide_legend(override.aes = list(size = 1.8)))
```



Abundant and rare taxa are defined by *Method c*

```
# load parameters mle
vec_process <- c('influent','anoxic','oxic','effluent')
param_plt <- data.frame()
for(item in vec_process){
  load(paste0('../output/AO/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/AO/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_c$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_c$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_c$AT, 'AT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

vec_region <- c('bay','era')
for(item in vec_region){
  load(paste0('../output/HZ/param_trio_',item,'.RData'))
  param_trio$ID <- rownames(param_trio)
  # extract gamma-poisson distributed microbes
  param_gpm <- param_trio[param_trio$k!=Inf,]
  # annotate procedure
```

```
  param_gpm$Procedure <- str_to_title(item)
  # annotate taxa category
  load(paste0('../output/HZ/abundant_rare_taxa_',item,'.RData'))
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_c$RT, 'RT', 'Other')
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_c$IT, 'IT', param_gpm$Category)
  param_gpm$Category <- ifelse(param_gpm$ID %in% taxa_cat_c$AT, 'AT', param_gpm$Category)

  param_plt <- rbind(param_plt, param_gpm)

}

rownames(param_plt) <- c(1:nrow(param_plt))
param_plt$Procedure <- ifelse(param_plt$Procedure=='Era','ERA',param_plt$Procedure)

param_plt$Procedure <- factor(param_plt$Procedure,
                              levels=c('Bay','Influent','Oxic',
                                       'ERA','Anoxic','Effluent'))
param_plt$Category <- factor(param_plt$Category,
                             levels=c('AT','IT','RT'))

ggplot() +
  geom_point(data=param_plt,
             aes(x=log(k), y=log(mu), color=Category), alpha=0.6,size=0.6) +
  ylab(TeX('\\log{(\\hat{\\mu})}')) +
  xlab(TeX('\\log{(\\hat{k})}')) +
  ggtitle(TeX('')) +
  xlim(-5,20) +
  ylim(-5,10) +
  theme_bw() +
  theme(aspect.ratio=1) +
  facet_wrap(Procedure~., scale='free') +
  scale_colour_manual(values = c('#e41a1c','#984ea3','#80b1d3')) +
  theme(panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12),
        axis.text.x = element_text(size = 9, vjust = 0.5, hjust = 0),
        axis.text.y = element_text(size=9),
        strip.text = element_text(size = 12),
        legend.title = element_text(color = "black", size = 12),
        legend.text = element_text(color = "black", size = 9)) +
  guides(color = guide_legend(override.aes = list(size = 1.8)))
```




```
sessionInfo()
```

```
## R version 4.2.1 (2022-06-23)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur ... 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.2/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.2/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] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] vegan_2.6-2     lattice_0.20-45 permute_0.9-7   ggh4x_0.2.2
##  [5] stringr_1.4.1   gridExtra_2.3   cowplot_1.1.1   latex2exp_0.9.4
##  [9] ggplot2_3.3.6   PMCosm_0.1.5
##
## loaded via a namespace (and not attached):
##  [1] highr_0.9         pillar_1.8.1      compiler_4.2.1    tools_4.2.1
##  [5] digest_0.6.29     viridisLite_0.4.1 nlme_3.1-158      evaluate_0.16
##  [9] lifecycle_1.0.1   tibble_3.1.8      gtable_0.3.1      mgcv_1.8-40
## [13] pkgconfig_2.0.3   rlang_1.0.4       Matrix_1.4-1      cli_3.3.0
## [17] rstudioapi_0.13   parallel_4.2.1    yaml_2.3.5        xfun_0.32
```

```
## [21] fastmap_1.1.0     cluster_2.1.3     withr_2.5.0       dplyr_1.0.9
## [25] knitr_1.40        generics_0.1.3    vctrs_0.4.1       tidyselect_1.1.2
## [29] glue_1.6.2        R6_2.5.1          fansi_1.0.3       rmarkdown_2.14
## [33] farver_2.1.1      purrr_0.3.4       magrittr_2.0.3    splines_4.2.1
## [37] MASS_7.3-58.1     scales_1.2.1      htmltools_0.5.3   colorspace_2.0-3
## [41] labeling_0.4.2    utf8_1.2.2        stringi_1.7.8     munsell_0.5.0
```