

MetaDAG experiment example

Version beta 0.2

2023-10-03

Table of contents

1 Load data MetaDag experiment

This is an example of an experiment of [metaDag](#) data with results.

Hash: bb261b6e-95c6-3e39-b82b-b68eea80e30b

results_ff15c187-62e7-37c2-96a7-c824f7eab671

URL:

```
library(tidyverse)
library(igraph)
library(ComplexHeatmap)
library(viridis)
library(circlize)
library(plotly)
library(randomcoloR)
library(factoextra)
library(RColorBrewer)
library(kableExtra)
library(igraph)
library(GGally)
```

Metadag Tool Experiment id: ff15c187-62e7-37c2-96a7-c824f7eab671
Web Page Experiment name: Eukaryotes
Experiment creator: Pere Palmer

[Home](#) [Experiment](#) [Results](#)

Experiment definition
Give the values needed to perform the desired experiment.

General data Here you can select the files containing the calculations made for your experiment.

	MBB	mDAG	Results	similarities
Select All	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

MetaDAG files Here, you can select what components have to be added to the downloaded data. The amount of options can be variable depending on the experiment and its configurations.

↑ ↓

Metadag Tool Experiment id: ff15c187-62e7-37c2-96a7-c824f7eab671
Web Page Experiment name: Eukaryotes
Experiment creator: Pere Palmer

[Home](#) [Experiment](#) [Results](#)

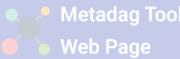
Experiment definition
Give the values needed to perform the desired experiment.

General data Here you can select the files containing the calculations made for your experiment.

MetaDAG files Here, you can select what components have to be added to the downloaded data. The amount of options can be variable depending on the experiment and its configurations.

type	format			representation			Select row		
	core	pan	image	svg	graphml	raw		Essential	Reaction
Global	<input checked="" type="checkbox"/>								
Categories	<input checked="" type="checkbox"/>								
Individuals	<input checked="" type="checkbox"/>								
Select column	<input checked="" type="checkbox"/>								

Delete ↑ →


Metadag Tool Experiment id: ff15c187-62e7-37c2-96a7-c824f7eab671

Web Page Experiment name: Eukaryotes
 Experiment creator: Pere Palmer

-  Home
-  Experiments
-  Results

Viewing the results

Once an experiment has been calculated it's possible to view the results. To do this, it's necessary to give the experiment unique identifier supplied in the email sent when the experiment was submitted. This unique identifier is also reported in response to the submission.

In the email sent, an url is given to access directly to the experiment results page.
 But here, you can also access to the experiment results with the unique identifier.

Enter the experiment identification code *

ff15c187-62e7-37c2-96a7-c824f7eab671






 ACGTCC 110101

 CGGC 1100

 GACCGT 2001011

 **UIB** | Universitat
 de les Illes Balears | Departament
 de Ciències Matemàtiques i Informàtica

```

experiment=
"result_bb261b6e-95c6-3e39-b82b-b68eea80e30b"
path_exp=paste0("data/",experiment,"/data/")
knitr:::kable(data.frame(
  Directory_files_and_folders=dir(path_exp),
  Type=c(rep("Data file",2),
  rep("Directory",2),
  rep("Data file",5),
  "Directory")))

```

Directory_files_and_folders	Type
Different_MBB.csv	Data file
Different_mDAG.csv	Data file
Global	Directory
Individuals	Directory
Results.csv	Data file
Similarities_MBB_MSAMethod.csv	Data file
Similarities_MBB_MunkresMethod.csv	Data file
Similarities_mDAG_MSAMethod.csv	Data file
Similarities_mDAG_MunkresMethod.csv	Data file
TaxonomyLevels	Directory

```

MBB=read_csv(paste0(path_exp,"Different_MBB.csv"),
             show_col_types = FALSE)

```

```

mDAG=read_csv(paste0(path_exp,"Different_mDAG.csv"),
               show_col_types = FALSE)
Results=read_csv(paste0(path_exp,"Results.csv"),
                 show_col_types = FALSE)
save(MBB,mDAG,Results,
      file="MBB_mDag_Results.Rdata")

```

1.1 MBB

In this experiment MBB is a table with 5112 rows and 4106 columns.

```

#100
knitr::kable(MBB[1:20,1:10]) %>%
  scroll_box(width = "100%", height = "200px")

```

MBB Id	natural	#pathways	Protists	Fungi	Plants	Animals	Alveolates	Amoebozoa	Annelids
0	0	0	0	0	0	0	0	0	0
0.0	0	0	0	0	0	0	0	0	0
0.0.0	1	1	1	0	0	0	0	0	0
0.0.0.0	1	1	1	0	0	0	0	0	0
0.0.1	1	1	1	0	0	0	0	0	0
0.0.1.0	1	86	2	77	1	6	1	0	0
0.0.1.1	1	86	2	77	1	6	1	0	0
0.0.1.2	1	15	15	0	0	0	12	0	0
0.0.1.3	1	5	5	0	0	0	0	4	0
0.0.1.4	0	0	0	0	0	0	0	0	0
0.0.1.4.0	1	609	14	128	128	339	1	3	1
0.0.1.4.1	1	609	14	128	128	339	1	3	1
0.0.1.4.2	1	322	6	92	133	91	1	0	0
0.0.1.4.3	1	322	6	92	133	91	1	0	0
0.0.1.4.4	1	1	0	0	1	0	0	0	0
0.0.1.4.5	1	1	1	0	0	0	0	1	0
0.0.1.5	0	0	0	0	0	0	0	0	0
0.0.1.5.0	1	5	5	0	0	0	0	4	0
0.0.2	1	1	1	0	0	0	0	0	0
0.0.3	1	1	1	0	0	0	0	1	0

1.2 mDAG

Abstract/unique mDAG's in this experiment

```
dim(mDAG)
```

```
[1] 884 5224
```

```
#100
```

```
kable(mDAG[1:20,1:10]) %>% scroll_box(width = "100%", height = "200px")
```

mDAG Id	#Categories	Animals	Plants	Fungi	Protists	Alveolates	Amoebozoa	Annelids	Arth
0001	3	1	0	0	0	0	0	0	
0002	2	0	0	1	0	0	0	0	
0003	2	1	0	0	0	0	0	0	
0004	3	1	0	0	0	0	0	0	
0005	3	1	0	0	0	0	0	0	
0006	3	0	1	0	0	0	0	0	
0007	2	0	1	0	0	0	0	0	
0008	3	0	1	0	0	0	0	0	
0009	3	0	1	0	0	0	0	0	
0010	3	1	0	0	0	0	0	0	
0011	3	1	0	0	0	0	0	0	
0012	3	0	0	0	1	0	0	0	
0013	3	1	0	0	0	0	0	0	
0014	3	0	0	0	1	1	0	0	
0015	2	0	0	1	0	0	0	0	
0016	3	0	0	0	1	0	0	1	0
0017	3	1	0	0	0	0	0	0	
0018	3	1	0	0	0	0	0	0	
0019	3	1	0	0	0	0	0	0	
0020	3	1	0	0	0	0	0	0	

```
dim(mDAG)
```

```
[1] 884 5224
```

```
names(mDAG)[1:6]
```

```
[1] "mDAG_Id"      "#Categories"  "Animals"       "Plants"       "Fungi"       
```

```
[6] "Protists"
```

```
head(names(mDAG)[7:(dim(mDAG)[2]-1150)])
```

```
[1] "Alveolates"      "Amoebozoa"      "Annelids"      
```

```
[4] "Arthropods"      "Ascomycetes"    "Basal_angiosperms"
```

```
# 28 to 1213 code MBB: 1 if MBB in mDAG 0
```

1.3 Results

Tabular data **Results** for this experiment

```
#100
#1:20
kable(Results[1:20,1:10])%>%
  row_spec(0, angle = 0) %>%
  scroll_box(width = "300%", height = "1000px")
```

organism	Categories	mDAG Id	Full Name
aaf	Protists Stramenopiles Pelagophytes	0036	Aureococcus anophagefferens
aag	Animals Arthropods Insects	0035	Aedes aegypti (yellow fever mosquito)
aalb	Animals Arthropods Insects	0276	Aedes albopictus (Asian tiger mosquito)
aali	Animals Arthropods Insects	0267	Anopheles albimanus
aalt	Fungi Ascomycetes Dothideomycetes	0240	Alternaria alternata
aam	Animals Vertebrates Birds	0040	Apteryx mantelli mantelli (North Island brown kiwi)
aamp	Animals Vertebrates Mammals	0313	Arvicola amphibius (Eurasian water vole)
aang	Animals Vertebrates Fishes	0317	Anguilla anguilla (European eel)
aara	Animals Arthropods Insects	0362	Anopheles arabiensis
abe	Fungi Ascomycetes Eurotiomycetes	0060	Trichophyton benhamiae
abp	Fungi Basidiomycetes	0068	Agaricus bisporus var. burnettii JB137-S8
abv	Fungi Basidiomycetes	0073	Agaricus bisporus var. bisporus H97
acan	Protists Amoebozoa Acanthamoeba	0873	Acanthamoeba castellanii
acar	Animals Vertebrates Birds	0884	Antrostomus carolinensis (chuck-will's-widow)
acep	Animals Arthropods Insects	0054	Atta cephalotes (leaf cutting ant)
acer	Animals Arthropods Insects	0057	Apis cerana (Asiatic honeybee)
achc	Animals Vertebrates Birds	0103	Aquila chrysaetos chrysaetos (golden eagle)
ache	Fungi Ascomycetes Eurotiomycetes	0106	Aspergillus chevalieri
achl	Animals Vertebrates Birds	0081	Acanthisitta chloris (rifleman)
acoz	Animals Arthropods Insects	0255	Anopheles coluzzii

```
dim(Results)
```

```
[1] 884 3997
```

```
names(Results)[1] # organisms kegg id class representant of mDAG
```

```
[1] "organism"
```

```
names(Results)[2] # taxonomy separate by /
```

```

[1] "Categories"

names(Results) [3] # mDAG_Id

[1] "mDAG_Id"

names(Results) [4] # Full name representant

[1] "Full Name"

names(Results) [6:36]# columns 6 to 2726 Variable Reactions name id: This reaction belongs to

[1] "R00710_rev(1.2.1.3)" "R00711(1.2.1.5)" "R00711_rev(1.2.1.5)"
[4] "R00755(4.1.1.1)" "R00746(1.1.1.2)" "R00746_rev(1.1.1.2)"
[7] "R00754(1.1.1.1)" "R00754_rev(1.1.1.1)" "R00014(4.1.1.1)"
[10] "R00014(1.2.4.1)" "R03270(1.2.4.1)" "R02569(2.3.1.12)"
[13] "R02569_rev(2.3.1.12)" "R00703(1.1.1.27)" "R00703_rev(1.1.1.27)"
[16] "R00200(2.7.1.40)" "R00658(4.2.1.11)" "R00658_rev(4.2.1.11)"
[19] "R01518(5.4.2.11)" "R01518_rev(5.4.2.11)" "R01061(1.2.1.12)"
[22] "R01061_rev(1.2.1.12)" "R01015(5.3.1.1)" "R01015_rev(5.3.1.1)"
[25] "R01070(4.1.2.13)" "R01070_rev(4.1.2.13)" "R04779(2.7.1.11)"
[28] "R04780(3.1.3.11)" "R02740(5.3.1.9)" "R02740_rev(5.3.1.9)"
[31] "R00959(5.4.2.2)"

reactions=names(Results)[-c(1:5)]
reverse_reactions=stringr::str_detect(reactions, "rev")
reverse_reactions=table(reverse_reactions)
dimnames(reverse_reactions)$reverse_reactions=c("One direction",
                                              "Reverse reaction")
reverse_reactions

reverse_reactions
  One direction Reverse reaction
      3398           594

```

2 Graphs

Read graphs in graphml format

2.1 Individuals graph for each organism

Read individual graph for Homo sapiens (Keg id: hsa) at directory(Individuals/hsa)

```
files_hsa=dir(paste0(path_exp,"Individuals/hsa"))
files_hsa
```

```
[1] "hsa_mDAG.graphml"      "hsa_mDAG_adj.csv"      "hsa_mDAG_biggerDAG.pdf"
[4] "hsa_mDAG_biggerDAG.svg" "hsa_mDAG_essential.pdf" "hsa_mDAG_essential.svg"
[7] "hsa_mDAG_nl.csv"       "hsa_mDAG_structure.csv" "hsa_R_adj.csv"
[10] "hsa_R_nl.csv"          "hsa_RC.graphml"        "hsa_RC.pdf"
[13] "hsa_RC.svg"
```

files_Individual_hsa	Description
hsa_mDAG.graphml	graph mDAG graphml format
hsa_mDAG_adj.csv	csv graph adjacency matrix of mDAG
hsa_mDAG_biggerDAG.pdf	plot bigger connected component mDAG pdf format
hsa_mDAG_biggerDAG.svg	plot bigger connected component mDAG svg format
hsa_mDAG_essential.pdf	plot essential MBB mDAG pdf format
hsa_mDAG_essential.svg	plot essential MBB mDAG svg format
hsa_mDAG_nl.csv	csv names labels MBB vertex of mDAG
hsa_mDAG_structure.csv	csv matrix membership of reactions of each MBB of mDAG ordered by decre
hsa_R_adj.csv	csv graph adjacency matrix of RC graph
hsa_R_nl.csv	csv names labels MBB vertex of RC
hsa_RC.graphml	graph RC graphml format
hsa_RC.pdf	plot RC pdf format
hsa_RC.svg	plot RC svg format

```
graph_mDAG=read.graph(
  paste0(path_exp,
  "Individuals/hsa/hsa_mDAG.graphml"),
```

```

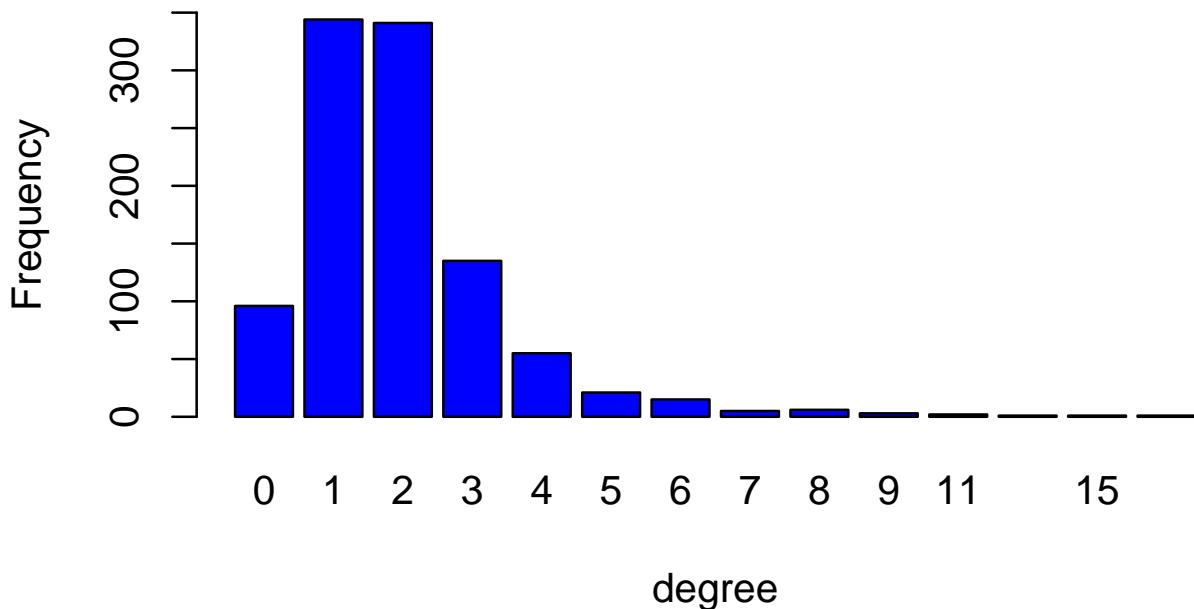
format= "graphml")
summary(graph_mDAG)

IGRAPH 3572354 D--- 1026 1086 --
+ attr: color (v/c), label (v/c), id (v/c), id (e/c)

barplot(table(igraph::degree(graph_mDAG, mode="all")),
        ylim=c(0,350), col="blue",
        main="Frequency of nodes degree",
        ylab="Frequency", xlab="degree")

```

Frequency of nodes degree



2.2 mDag Global core for eukaryotes

Note that the mDAG core is empty as it does not contain any reactions.

2.2.1 Core mDAG

```

graph_core_mDAG=read.graph(
  paste0(path_exp,"Global/core/core_mDAG.graphml"),

```

```

format = "graphml")
summary(graph_core_mDAG)

IGRAPH 36f0ee2 D--- 0 0 --
+ attr: color (v/c), label (v/c), id (v/c)

# file missing
knitr::include_graphics(paste0(path_exp,
                                "Global/core/core_mDAG.pdf"))

```

The graph core mDAG have 0 vertex and 0, is an empty graph.

2.2.2 Core reaction graph (RC)

```

graph_core_RC=read.graph(
  paste0(path_exp,
         "Global/core/core_RC.graphml"),
  format = "graphml")
summary(graph_core_mDAG)

IGRAPH 36f0ee2 D--- 0 0 --
+ attr: color (v/c), label (v/c), id (v/c)

knitr::include_graphics(
  paste0(path_exp,"Global/core/core_RC.pdf"))

```

Figure 2.1: Core mDAG is empty

The graph core reaction graph have 0 vertex and 0, is an empty graph.

2.3 mDag Global pan for eukaryotes

2.3.1 Pan mDAG

```
graph_pan_mDAG=read.graph(
  paste0(path_exp,
    "TaxonomyLevels/Kingdom/Animals/pan/Animals_pan_mDAG.graphml"),
  format = "graphml")
summary(graph_pan_mDAG)
```

```
IGRAPH 36fac37 D--- 1184 1261 --
+ attr: color (v/c), label (v/c), id (v/c), id (e/c)
```

The graph pan mDAG have 1184 vertex and 1261.

2.3.2 Pan Reaction graph (RC)

```
graph_pan_RC=read.graph(
  paste0(path_exp,
    "TaxonomyLevels/Kingdom/Animals/pan/Animals_pan_RC.graphml"),
  format = "graphml")
summary(graph_pan_RC)
```

```
IGRAPH 37046cb D--- 4556 5798 --
+ attr: color (v/c), label (v/c), id (v/c), id (e/c)
```

The graph pan reaction graph have 4556 vertex and 5798.

```
compo=components(graph_mDAG, mode = "weak")
str(compo)

List of 3
$ membership: num [1:1026] 1 1 1 1 1 1 1 1 1 ...
$ csize      : num [1:167] 589 1 1 1 1 1 4 3 4 3 ...
$ no         : int 167

compo$csize
```

```

[1] 589  1  1  1  1  1  1  4  3  4  3  2  3  3  1  1  1  1  2  6
[19]   3  1  3  6  1  1  1  1  1  3  1  6  2  1  1  1  1  1  1  2  1
[37]   1 14  1 16  1  6  2  2  4  1  1  1  1  1  1  1  1  1  1  1  1
[55]  13  1  1  1  1  2  6  5  5  2  2  2  10  1  1  1  1  2  2  1
[73]   1  1 62  6  2  1  2  1  1  1  2  1  2  14  3  1  1  1  1  2  1
[91]   1  1  1  1  1  1  3  6  1  3  1  3  2  1  1  2  3  4  1  1  2
[109]   3  1  1  2  5  1  1  2  3  2  1  1  2  3  4  1  1  1  2  2
[127]   1  1  2  1  1  1  3  1  2  2  1  6  1  1  1  2  1  3  1  1  2
[145]   1  1  1  1  1  2  7  1 15  3  1  1  1  1  2  1  1  3  1  1
[163]   1  1  1  1  1  2

```

```

k=which.max(compo$csizes==max(compo$csizes))
k

```

```
[1] 1
```

```
table(compo$membership)
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
589	1	1	1	1	1	4	3	4	3	2	3	3	1	1	1	1	2	6	3	1
21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	
3	6	1	1	1	1	1	3	1	6	2	1	1	1	2	1	1	14	1	16	
41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	
1	6	2	2	4	1	1	1	1	1	1	1	1	1	13	1	1	1	1	2	
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	
6	5	5	2	2	10	1	1	1	2	2	1	1	1	62	6	2	1	2	1	
81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	
1	1	2	1	2	14	3	1	1	1	1	1	1	1	1	1	1	3	6	1	3
101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	
1	3	2	1	1	1	2	2	3	1	1	2	5	1	1	2	3	2	1	1	
121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	
2	3	4	1	1	2	1	1	2	1	1	1	1	1	3	1	2	2	1	6	
141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	160	
1	1	1	2	1	1	1	1	1	2	7	1	15	3	1	1	1	1	2	1	
161	162	163	164	165	166	167														
3	1	1	1	1	1	1	2													

```

vertex=which(compo$membership==k)
length(vertex)

```

```
[1] 589
```

```
Big_Component=induced_subgraph(graph_mDAG, vids=vertex)
igraph::vcount(Big_Component)
```

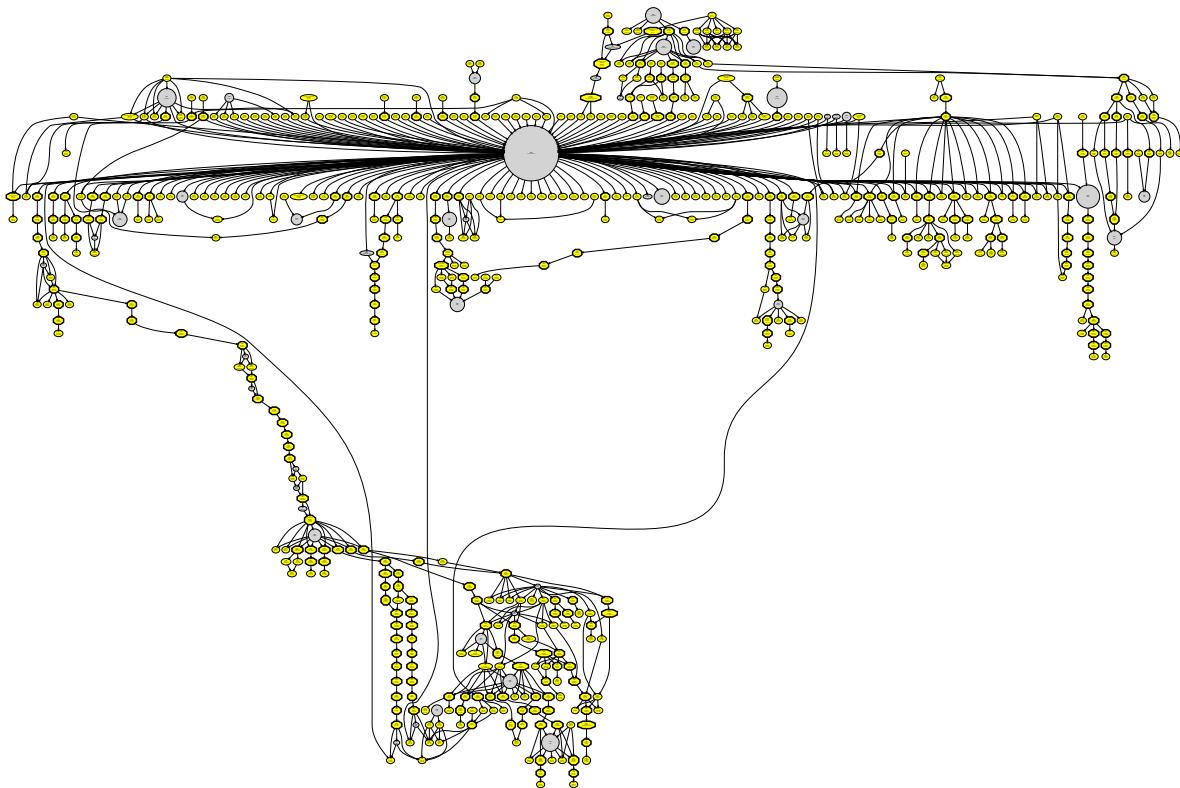
```
[1] 589
```

```
igraph::ecount(Big_Component)
```

```
[1] 774
```

The curated plot of the bigger component of hsa mDAG

```
knitr:::include_graphics(paste0(path_exp,
"Individuals/hsa/hsa_mDAG_biggerDAG.pdf"))
```



3 Similarities and metadata for an experiment

We will first load the metadata and adapt them to the structure of the similarities to facilitate the creation of the graphs and statistics.

Remember de path of the experiment:

```
path_exp
```

```
[1] "data/result_bb261b6e-95c6-3e39-b82b-b68eea80e30b/data/"
```

3.1 Load meta data from eukariotes experimet

Meta data mDa_Id and taxonomy sort by Kingdom,Filum,Class,mDAG_Id

```
path_exp
```

```
[1] "data/result_bb261b6e-95c6-3e39-b82b-b68eea80e30b/data/"

Results=read_csv(paste0(path_exp,"Results.csv"))
names(Results)[c(1,3,4)]=c("Organism","mDAG_Id","Full_Name")
taxo=Results %>% select(Organism:Full_Name)
taxo=taxo %>% separate(Categories,into=c("Kingdom","Phylum","Class"))
index=which(is.na(taxo$Class))
taxo$Class[index]=paste(taxo$Phylum[index])
meta_taxo=taxo
aux=table(meta_taxo$Phylum)
Freq_Phylum=tibble(Phylum=names(aux),Freq_Phylum=aux)
names(Freq_Phylum)=c("Phylum","Freq_Phylum")
aux=table(meta_taxo$Class)
Freq_Class=tibble(Class=names(aux),Freq_Class=aux)
names(Freq_Class)=c("Class","Freq_Class")

meta_taxo = meta_taxo %>%
  left_join(Freq_Phylum) %>%
```

```

  left_join(Freq_Class)
meta_taxo = meta_taxo %>%
  arrange(Kingdom,desc(Freq_Phylum),Phylum,
         desc(Freq_Class),Class)
head(meta_taxo)

# A tibble: 6 x 8
  Organism Kingdom Phylum      Class   mDAG_Id Full_Name Freq_Phylum Freq_Class
  <chr>     <chr>   <chr>       <chr>    <chr>      <table[1d]> <table[1d>
1 aamp      Animals Vertebrates Mammals 0313   Arvicola ~ 331      139
2 afz       Animals Vertebrates Mammals 0143   Antechinu~ 331      139
3 ajm       Animals Vertebrates Mammals 0221   Artibeus ~ 331      139
4 aju       Animals Vertebrates Mammals 0224   Acinonyx ~ 331      139
5 aml       Animals Vertebrates Mammals 0279   Ailuropod~ 331      139
6 anu       Animals Vertebrates Mammals 0310   Arvicanth~ 331      139

table(meta_taxo$Kingdom) %>% kable %>%
  kable_styling("striped", full_width = F,position="left")%>%
  scroll_box(width = "400px", height = "200px")



| Var1     | Freq |
|----------|------|
| Animals  | 535  |
| Fungi    | 154  |
| Plants   | 139  |
| Protists | 56   |



table(meta_taxo$Phylum,meta_taxo$Kingdom) %>% kable %>%
  kable_styling("striped", full_width = F,position="left")%>%
  scroll_box(width = "500px", height = "500px")

```

	Animals	Fungi	Plants	Protists
Alveolates	0	0	0	25
Amoebozoa	0	0	0	7
Annelids	1	0	0	0
Arthropods	158	0	0	0
Ascomycetes	0	113	0	0
Basal	0	0	2	0
Basidiomycetes	0	36	0	0
Brachiopoda	1	0	0	0
Cephalochordates	2	0	0	0
Choanoflagellates	0	0	0	2
Cnidarians	10	0	0	0
Cryptomonads	0	0	0	1
Echinoderms	3	0	0	0
Eudicots	0	0	98	0
Euglenozoa	0	0	0	9
Ferns	0	0	1	0
Flatworms	4	0	0	0
Green	0	0	11	0
Haptophyta	0	0	0	1
Hemichordates	1	0	0	0
Heterolobosea	0	0	0	1
Metamonada	0	0	0	2
Microsporidians	0	5	0	0
Mollusks	14	0	0	0
Monocots	0	0	23	0
Mosses	0	0	1	0
Nematodes	6	0	0	0
Placozoans	1	0	0	0
Poriferans	1	0	0	0
Red	0	0	3	0
Stramenopiles	0	0	0	8
Tunicates	2	0	0	0
Vertebrates	331	0	0	0

3.2 Similarities MSA,Munkres methods

In this section we will show the similarities between mDAG's using different methods.

The experiment data set consists of 884 eukaryotes from the animal, plant, fungus, and protist kingdoms.

Kingdom	Abs. Freq.
Animals	535
Fungi	154
Plants	139
Protists	56

```
list_Sim=dir(path_exp,pattern="^Similarities")
list_Sim

[1] "Similarities_MBB_MSAMethod.csv"      "Similarities_MBB_MunkresMethod.csv"
[3] "Similarities_mDAG_MSAMethod.csv"     "Similarities_mDAG_MunkresMethod.csv"
```

Load MDAG similarities

```
Sim_MSA_mDAG=read_csv(paste0(path_exp,
                               "Similarities_mDAG_MSAMethod.csv"))
Sim_MSA_mDAG=as.matrix(Sim_MSA_mDAG[,-1])
rownames(Sim_MSA_mDAG)=colnames(Sim_MSA_mDAG)
Sim_MSA_mDAG=Sim_MSA_mDAG[meta_taxo$mDAG_Id,meta_taxo$mDAG_Id]

Sim_Mun_mDAG=read_csv(paste0(path_exp,
                               "Similarities_mDAG_MunkresMethod.csv"))
Sim_Mun_mDAG=as.matrix(Sim_Mun_mDAG[,-1])
rownames(Sim_Mun_mDAG)=colnames(Sim_Mun_mDAG)
Sim_Mun_mDAG=Sim_Mun_mDAG[meta_taxo$mDAG_Id,meta_taxo$mDAG_Id]
```

3.3 Heatmaps

3.3.1 Heatmap Similarity MSA and Munkres method

```
dff<-meta_taxo %>% select(Kingdom)  %>% as.data.frame()
colorsK <- list(Kingdom= c("Animals"="red",
                           "Plants"="green",
                           "Fungi"="yellow",
                           "Protists"="black"))
annotationK <- HeatmapAnnotation(df=dff, col = colorsK,show_legend = TRUE)

MSA_heat_1 <- Heatmap(matrix = Sim_MSA_mDAG,
                       column_title=
                           "m-DAGs MSA-similarity Eukaryotes by Kingdoms",
                           heatmap_legend_param=list(
```

```

        title="Similarity",
        at = seq(0,1,by=0.1)),
col=rev(viridis(256)),
cluster_rows = FALSE,
cluster_columns = FALSE,
top_annotation = annotationK,
show_column_names = FALSE,
show_row_names = FALSE,
left_annotation =
  rowAnnotation(df = diff,
    col = colorsK,
    show_annotation_name=FALSE,
    show_legend=FALSE
  ))
}

Mun_heat_1<- Heatmap(matrix = Sim_Mun_mDAG,
  column_title="mDAGs Munkres-similarity Eukaryotes by Kingdoms",
  name = "Munkres Similarity",
  heatmap_legend_param=list(
    title="Similarity",
    at = seq(0,1,by=0.1)),
col=rev(viridis(256)),
cluster_rows = FALSE,
cluster_columns = FALSE,
top_annotation = annotationK,
show_column_names = FALSE,
show_row_names = FALSE,
left_annotation =
  rowAnnotation(df = diff,
    col = colorsK,
    show_annotation_name=FALSE,
    show_legend=FALSE
  ))
}

meta_animals= meta_taxo %>% filter(Kingdom=="Animals")

diff<-meta_taxo %>%

```

```

filter(Kingdom=="Animals") %>%
select(Phylum,Freq_Phylum) %>%
as.data.frame() %>% select(Phylum)

namesP=dff %>% distinct( Phylum, .keep_all = TRUE)
namesP=namesP$Phylum
dff$Phylum=ordered(dff$Phylum,labels=namesP)
col=rainbow(length(namesP))
colorsP=list(Phylum=col)
names(colorsP$Phylum)=namesP
annotation_H2 <- HeatmapAnnotation(df=dff, col = colorsP)
MSA_heat_2 <- Heatmap(matrix =
                        Sim_MSA_mDAG[1:nrow(dff),1:nrow(dff)],
                        column_title="mDAGs MSA-similarity Animals by Phyla",
                        col=rev(viridis(256)),
                        cluster_rows = FALSE,
                        show_heatmap_legend=FALSE,
                        cluster_columns = FALSE,
                        top_annotation = annotation_H2,
                        show_column_names = FALSE,
                        show_row_names = FALSE,
                        left_annotation =
                            rowAnnotation(df = dff,
                                          col = colorsP,
                                          show_annotation_name=FALSE,
                                          show_legend =FALSE
                                         ))
                                        

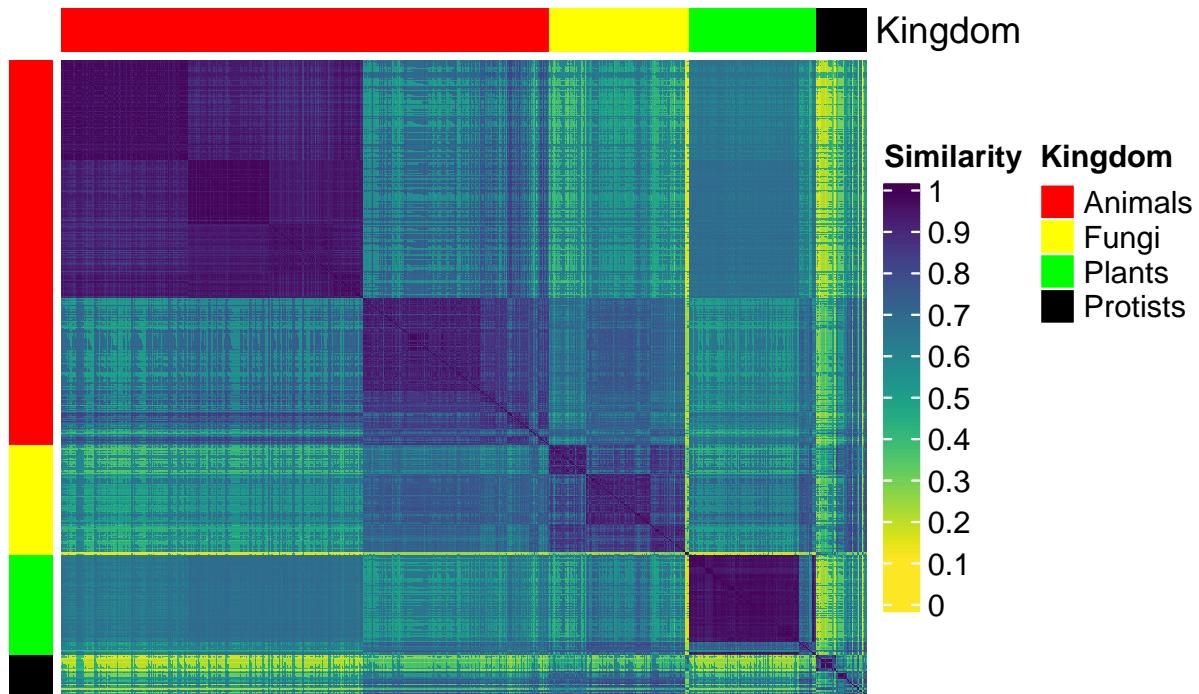
Mun_heat_2 <- Heatmap(matrix = Sim_Mun_mDAG[1:nrow(dff),1:nrow(dff)],
                       column_title="mDAGs Munkres-similarity Animals by Phyla",
                       col=rev(viridis(256)),
                       show_heatmap_legend=FALSE,
                       cluster_rows = FALSE,
                       cluster_columns = FALSE,
                       top_annotation = annotation_H2,
                       show_column_names = FALSE,
                       show_row_names = FALSE,
                       left_annotation = rowAnnotation(df = dff,
                                                       col = colorsP,
                                                       show_annotation_name=FALSE,
                                                       show_legend =FALSE
                                                      )

```

))

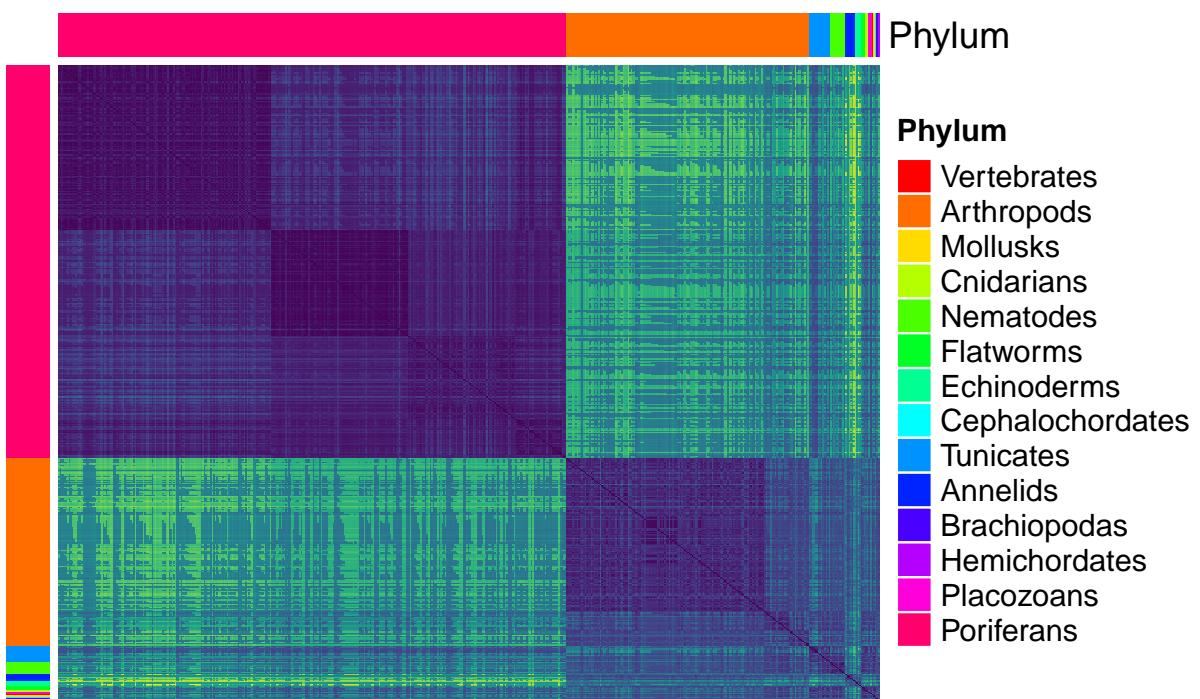
MSA_heat_1

m-DAGs MSA-similarity Eukaryotes by Kingdoms



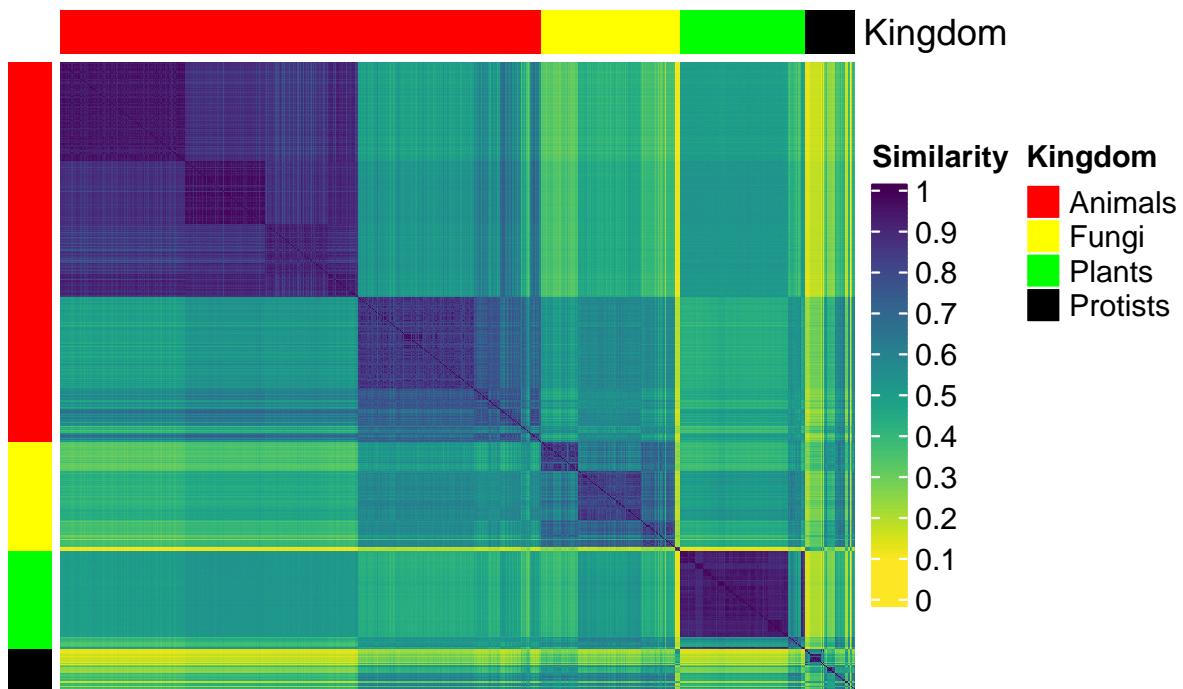
MSA_heat_2

mDAGs MSA–similarity Animals by Phyla



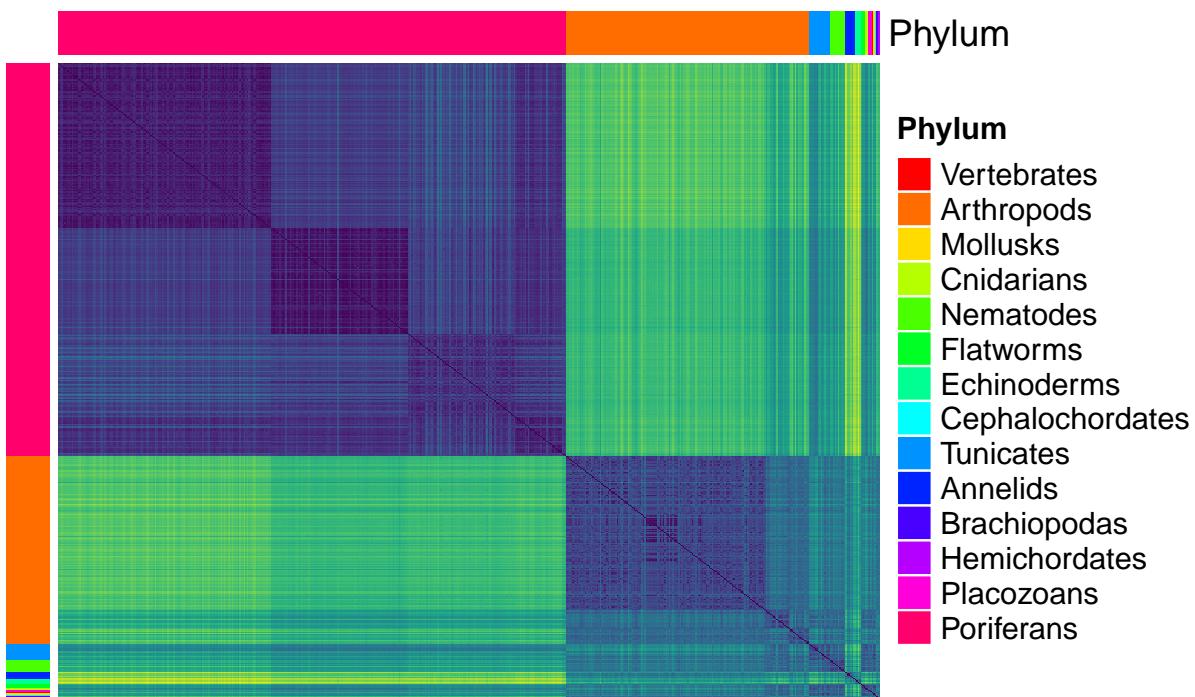
Mun_heat_1

mDAGs Munkres–similarity Eukaryotes by Kingdoms



Mun_heat_2

mDAGs Munkres–similarity Animals by Phyla



3.4 MDS (Multidimensional Scaling) MSA

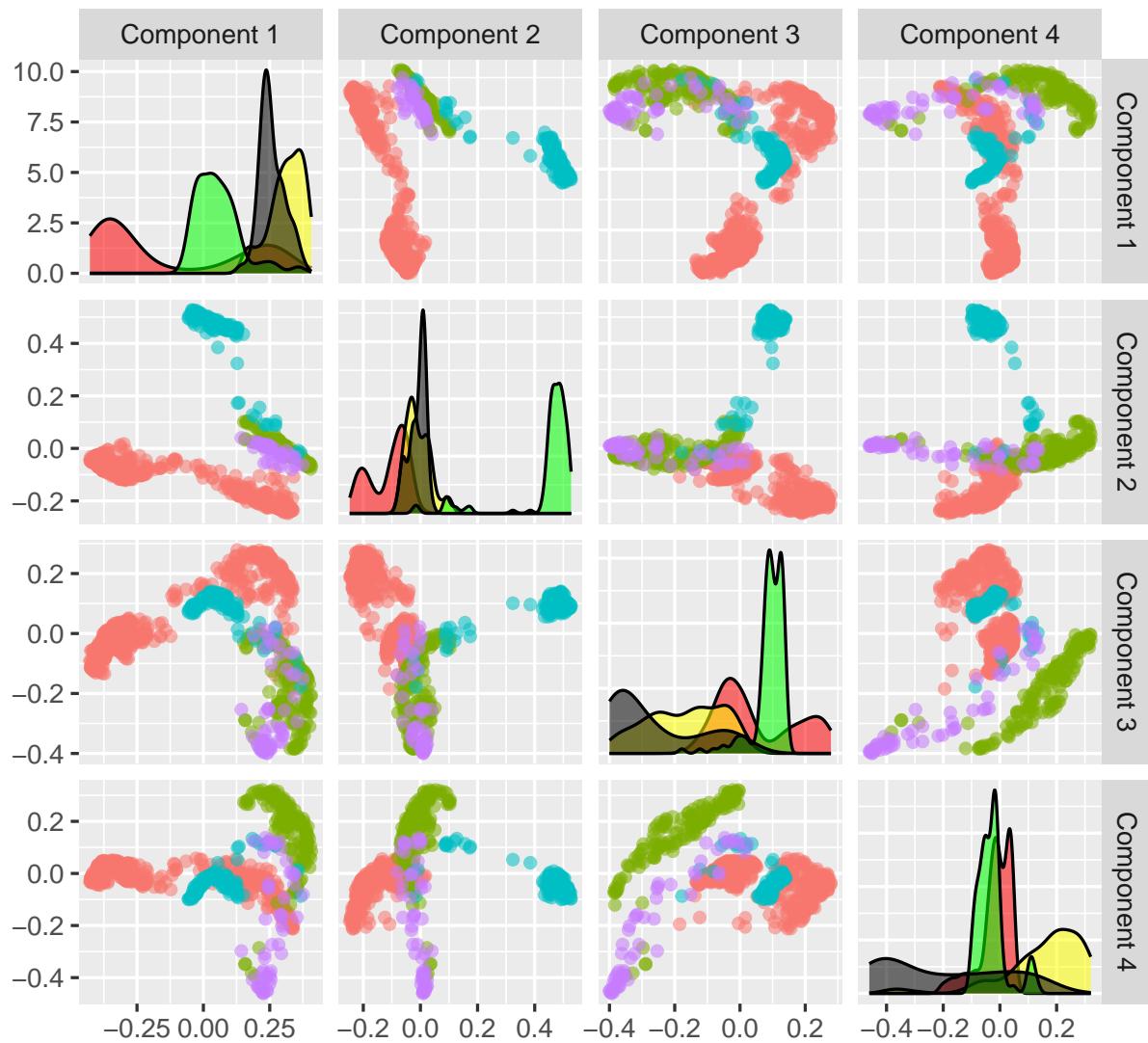
```
## Metric multidimensional scaling (mMDS)
mds7 <- cmdscale(sqrt(1-Sim_MSA_mDAG^2), k=7, eig=TRUE)
#pairs(mds7$points[,1:4])
mds7$GOF

[1] 0.4406568 0.5541042

mds <- mds7$points %>% as_tibble()
colnames(mds) <- paste0("Dim.", 1:dim(mds7$points)[2])

cooordinates=as_tibble(mds7$points)
colnames(cooordinates)=paste("Component", 1:7)
ggpairs(cooordinates, columns=1:4,
        aes(color=meta_taxo$Kingdom, alpha=0.5,
            title="MDS 4 dimensions projection",
            legend=1), upper=list(continuous="points")) +
```

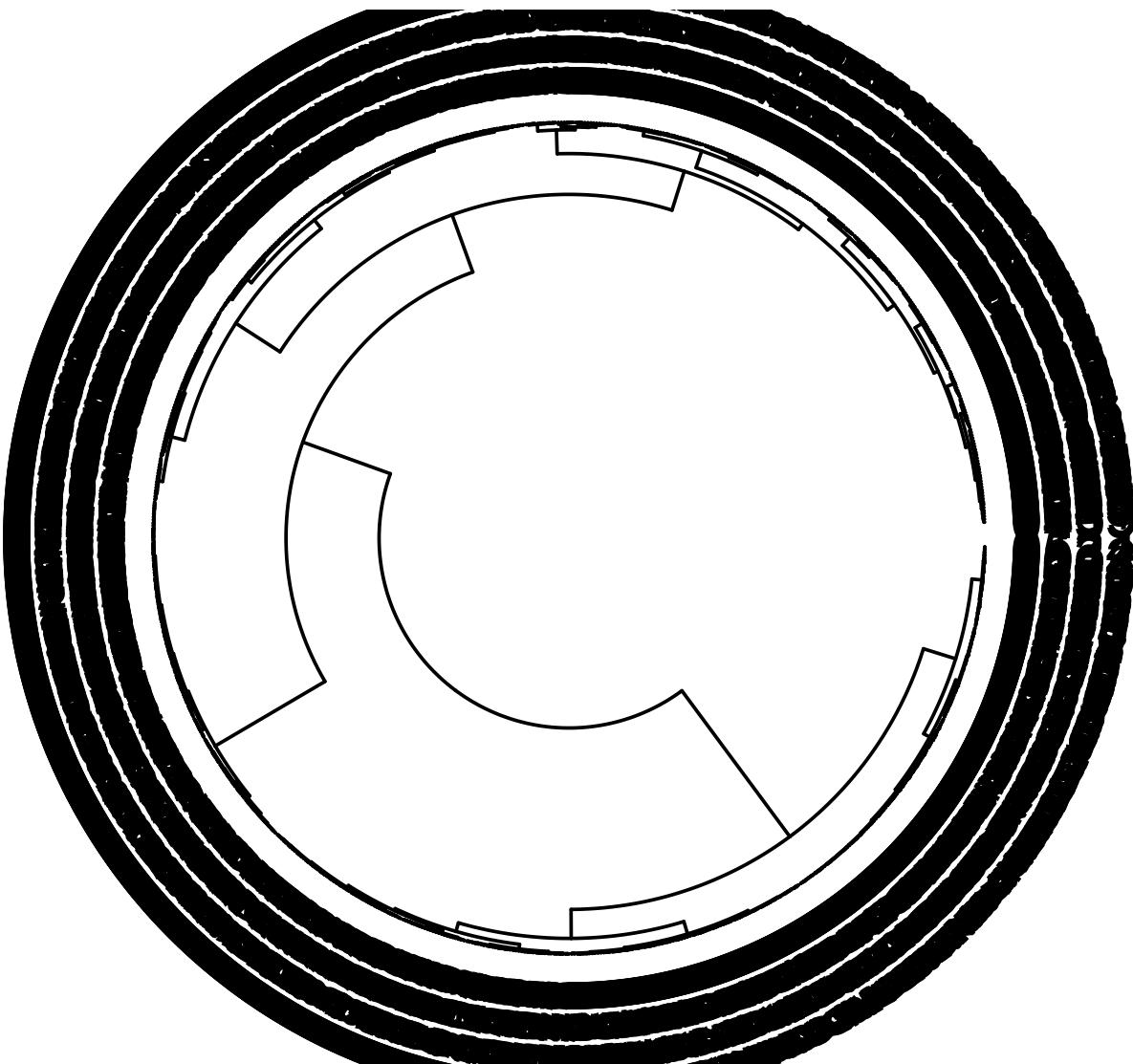
```
scale_fill_manual(values = colorsK$Kingdom) +  
theme(legend.position = "left")
```



4 Hierarchical cluster MSA

```
library(dendextend)
library(ggraph)
library(ape)

D=as.dist(sqrt(1-Sim_MSA_mDAG^2))
hc_MSA=hclust(as.dist(D),method ="ward.D")
library(circlize)
circlize_dendrogram(as.dendrogram(hc_MSA))
```



```
clust4_MSA=cutree(hc_MSA,4)
table(clust4_MSA,meta_taxo$Kingdom)
```

clust4_MSA	Animals	Fungi	Plants	Protists
1	331	0	0	0
2	195	0	0	0
3	9	154	14	56
4	0	0	125	0

```
aux=meta_taxo %>%
  select(Organism,Kingdom,Phylum,Class,Full_Name)
```

```

aux$clust4_MSA=clust4_MSA
aux_Animals_cluster_1_2 = aux %>%
  filter(Kingdom=="Animals",clust4_MSA %in% c(1,2))






```

```

Phylum %in% c("Nematodes", "Flatworms"))
aux_all_Nematodes_Flatworms

# A tibble: 10 x 6
  Organism Kingdom Phylum Class Full_Name      clust4_MSA
  <chr>     <chr>   <chr>  <chr>    <chr>          <int>
1 bmy       Animals Nematodes Nematodes Brugia malayi (filaria) 3
2 cbr       Animals Nematodes Nematodes Caenorhabditis briggsae (nem~ 3
3 cel       Animals Nematodes Nematodes Caenorhabditis elegans (nema~ 3
4 loa       Animals Nematodes Nematodes Loa loa (eye worm)        3
5 nai       Animals Nematodes Nematodes Necator americanus (New Worl~ 2
6 tsp       Animals Nematodes Nematodes Trichinella spiralis        3
7 egl       Animals Flatworms Flatworms Echinococcus granulosus (hyd~ 3
8 ovi       Animals Flatworms Flatworms Opisthorchis viverrini (Sout~ 3
9 shx       Animals Flatworms Flatworms Schistosoma haematobium (uri~ 3
10 smm      Animals Flatworms Flatworms Schistosoma mansoni        3

aux_14_Plants_clust2= filter(aux,clust4_MSA==3,
                               Kingdom=="Plants")
aux_14_Plants_clust2

# A tibble: 14 x 6
  Organism Kingdom Phylum Class Full_Name      clust4_MSA
  <chr>     <chr>   <chr>  <chr>    <chr>          <int>
1 apro      Plants  Green  algae Auxenochlorella protothecoides 3
2 bpg       Plants  Green  algae Bathycoccus prasinos        3
3 cre       Plants  Green  algae Chlamydomonas reinhardtii 3
4 csl       Plants  Green  algae Cocomyxa subellipsoidea 3
5 cvr       Plants  Green  algae Chlorella variabilis        3
6 mis       Plants  Green  algae Micromonas commoda        3
7 mng       Plants  Green  algae Monoraphidium neglectum 3
8 mpp       Plants  Green  algae Micromonas pusilla        3
9 olu       Plants  Green  algae Ostreococcus lucimarinus 3
10 ota      Plants  Green  algae Ostreococcus tauri        3
11 vcn      Plants  Green  algae Volvox carteri f. nagariensis 3
12 ccp      Plants  Red    algae Chondrus crispus (carragheen) 3
13 cme      Plants  Red    algae Cyanidioschyzon merolae 3
14 gsl      Plants  Red    algae Galdieria sulphuraria 3

aux_all_algae_class= aux %>%
  filter(Kingdom=="Plants",
         Class %in% c("algae"))
aux_all_algae_class

```

	Organism	Kingdom	Phylum	Class	Full_Name	clust4_MSA
	<chr>	<chr>	<chr>	<chr>	<chr>	<int>
1	apro	Plants	Green	algae	Auxenochlorella protothecoides	3
2	bpg	Plants	Green	algae	Bathycoccus prasinos	3
3	cre	Plants	Green	algae	Chlamydomonas reinhardtii	3
4	csl	Plants	Green	algae	Coccomyxa subellipsoidea	3
5	cvr	Plants	Green	algae	Chlorella variabilis	3
6	mis	Plants	Green	algae	Micromonas commoda	3
7	mng	Plants	Green	algae	Monoraphidium neglectum	3
8	mpp	Plants	Green	algae	Micromonas pusilla	3
9	olu	Plants	Green	algae	Ostreococcus lucimarinus	3
10	ota	Plants	Green	algae	Ostreococcus tauri	3
11	vcn	Plants	Green	algae	Volvox carteri f. nagariensis	3
12	ccp	Plants	Red	algae	Chondrus crispus (carragheen)	3
13	cme	Plants	Red	algae	Cyanidioschyzon merolae	3
14	gsl	Plants	Red	algae	Galdieria sulphuraria	3

The hierarchical classification by Ward's method recovers the kingdom Animal clusters 1 (all vertebrates) and 2 (invertebrate animals), cluster 4 the Plants and in cluster 3 are all protists and fungi together with 9 animals and 14 plants.

The 9 Animals are all from the Phylum Nematodes or Flatworms, out of the total of the 10 species of these phylums considered in the experiment. Only the Nematode Necator americanus (New World hookworm) is classified in Animals.

The 14 plants in cluster 2 are all algae considered in the experiment.

4.1 MDS (Multidimensional Scaling) Munkres

```
## Metric multidimensional scaling
mds7 <- cmdscale(sqrt(1-Sim_Mun_mDAG^2), k=7, eig=TRUE)
mds7$GOF

[1] 0.5600385 0.5796647

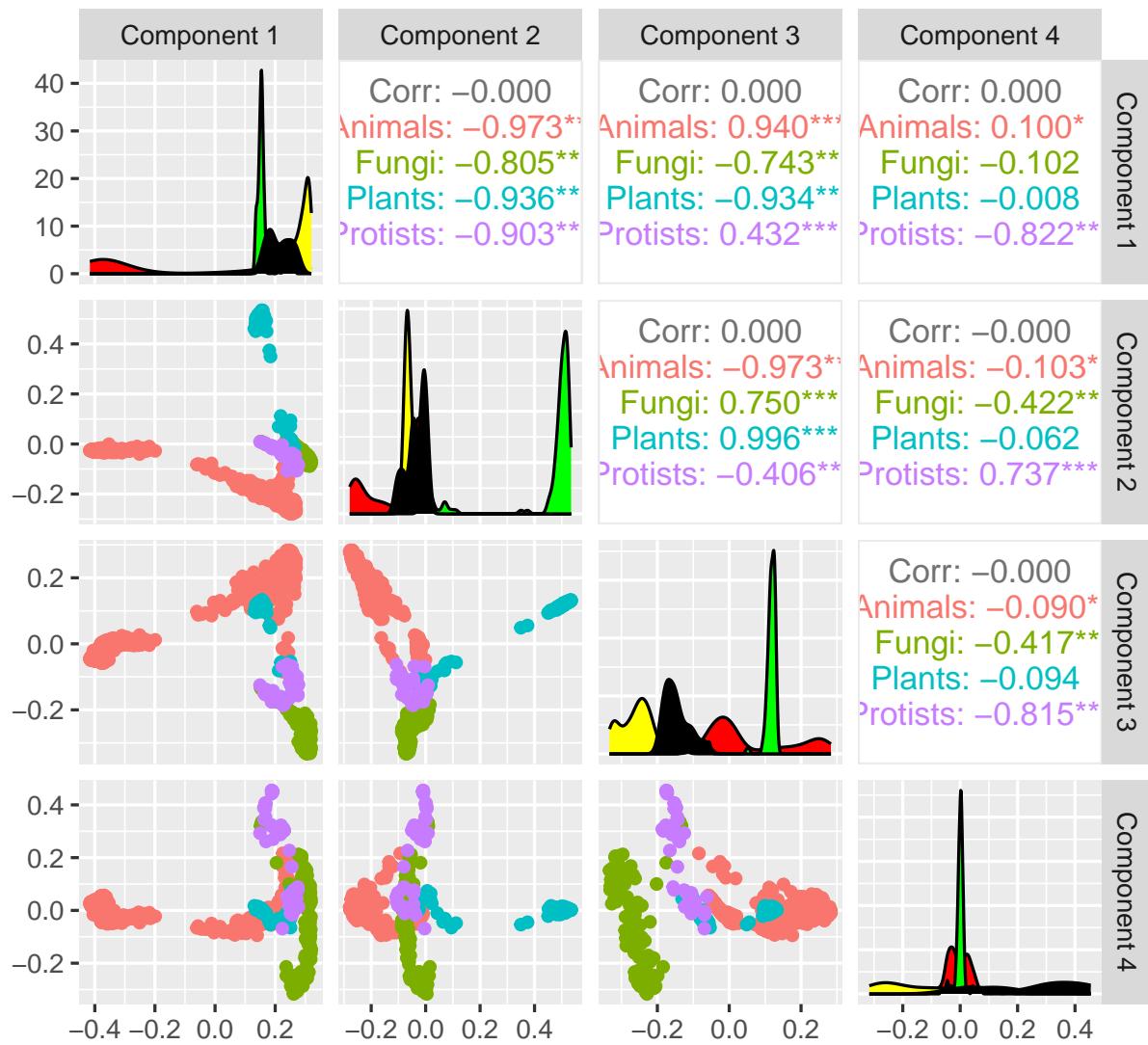
mds <- mds7$points %>% as_tibble()
colnames(mds) <- paste0("Dim.", 1:dim(mds7$points)[2])

coordinates=as_tibble(mds7$points)
colnames(coordinates)=paste("Component", 1:7)
ggpairs(coordinates, columns=1:4,
```

```

aes(color=meta_taxo$Kingdom,
   title="MDS 4 dimensions projection",legend=1),
lower=list(continuous="points")) +
scale_fill_manual(values = colorsK$Kingdom) +
theme(legend.position = "left")

```



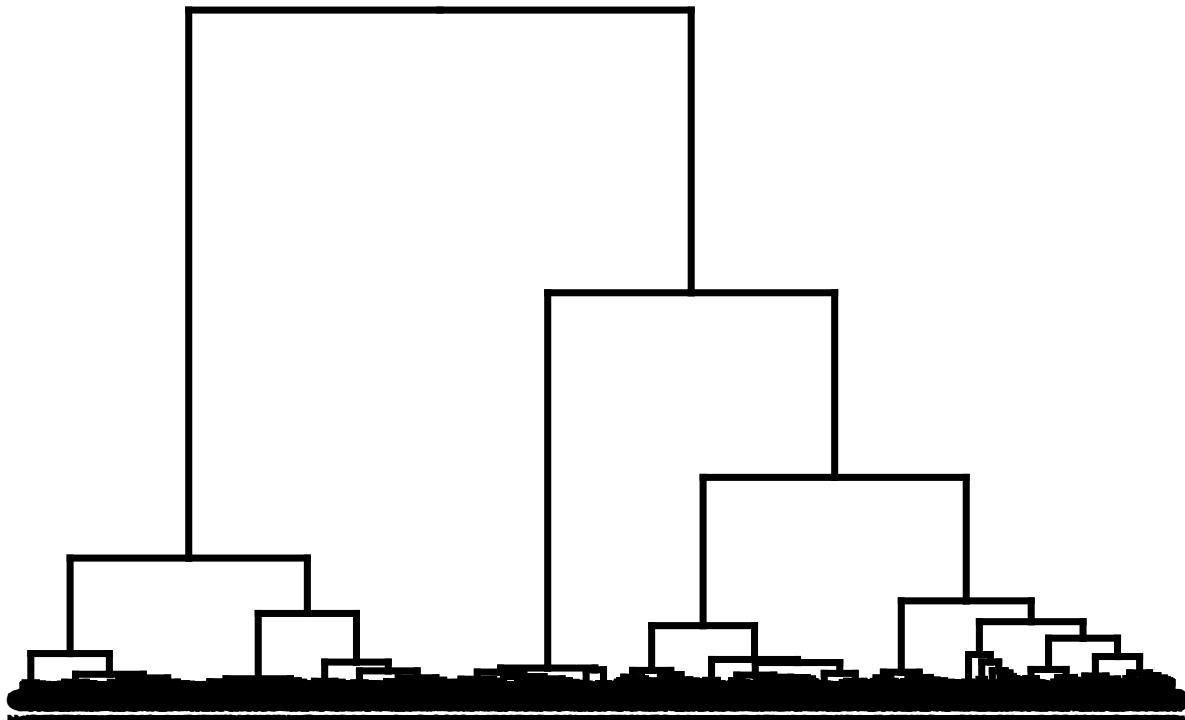
4.2 Hierarchical cluster Munkres

```

D=as.dist(sqrt(1-Sim_Mun_mDAG^2))
hc_Mun=hclust(as.dist(D),method ="ward.D")

```

```
ggplot(as.ggdend(as.dendrogram(hc_Mun)))
```



```
clust4_Mun=cutree(hc_Mun,4)
table(clust4_Mun,meta_taxo$Kingdom)
```

clust4_Mun	Animals	Fungi	Plants	Protists
1	331	0	0	0
2	197	0	0	0
3	7	154	14	56
4	0	0	125	0

Only two nematodes of the 9 return to the invertebrate animal cluster for Munkrest; all algae remain in the fungi and archaea cluster.

```
aux=meta_taxo %>%
  select(Organism,Kingdom,Phylum,Class,Full_Name)
aux$clust4_Mun=clust4_Mun
aux_Animals_cluster_1_2_Mun = aux %>%
  filter(Kingdom=="Animals",clust4_Mun %in% c(1,2))
```

```



```

						<int>
1	bmy	Animals	Nematodes	Nematodes	Brugia malayi (filaria)	3
2	cbr	Animals	Nematodes	Nematodes	Caenorhabditis briggsae (nem~	2
3	cel	Animals	Nematodes	Nematodes	Caenorhabditis elegans (nema~	2
4	loa	Animals	Nematodes	Nematodes	Loa loa (eye worm)	3
5	nai	Animals	Nematodes	Nematodes	Necator americanus (New Worl~	2
6	tsp	Animals	Nematodes	Nematodes	Trichinella spiralis	3
7	egl	Animals	Flatworms	Flatworms	Echinococcus granulosus (hyd~	3
8	ovi	Animals	Flatworms	Flatworms	Opisthorchis viverrini (Sout~	3
9	shx	Animals	Flatworms	Flatworms	Schistosoma haematobium (uri~	3
10	smm	Animals	Flatworms	Flatworms	Schistosoma mansoni	3

```

aux_14_Plants_clust2_Mun= filter(aux,clust4_Mun==3,
                                   Kingdom=="Plants")
aux_14_Plants_clust2_Mun

# A tibble: 14 x 6
#> #> Organism Kingdom Phylum Class Full_Name      clust4_Mun
#> #> <chr>   <chr>   <chr>  <chr> <chr>           <int>
#> 1 apro    Plants  Green  algae Auxenochlorella protothecoides 3
#> 2 bpg     Plants  Green  algae Bathycoccus prasinos       3
#> 3 cre     Plants  Green  algae Chlamydomonas reinhardtii  3
#> 4 csl     Plants  Green  algae Coccomyxa subellipsoidea 3
#> 5 cvr     Plants  Green  algae Chlorella variabilis    3
#> 6 mis     Plants  Green  algae Micromonas commoda     3
#> 7 mng     Plants  Green  algae Monoraphidium neglectum  3
#> 8 mpp     Plants  Green  algae Micromonas pusilla     3
#> 9 olu     Plants  Green  algae Ostreococcus lucimarinus 3
#> 10 ota    Plants  Green  algae Ostreococcus tauri     3
#> 11 vcn    Plants  Green  algae Volvox carteri f. nagariensis 3
#> 12 ccp     Plants  Red    algae Chondrus crispus (carragheen) 3
#> 13 cme    Plants  Red    algae Cyanidioschyzon merolae   3
#> 14 gsl    Plants  Red    algae Galdieria sulphuraria  3

```

```

aux_all_algae_class= aux %>%
  filter(Kingdom=="Plants",
         Class %in% c("algae"))
aux_all_algae_class

# A tibble: 14 x 6
#> #> Organism Kingdom Phylum Class Full_Name      clust4_Mun
#> #> <chr>   <chr>   <chr>  <chr> <chr>           <int>
#> 1 apro    Plants  Green  algae Auxenochlorella protothecoides 3
#> 2 bpg     Plants  Green  algae Bathycoccus prasinos       3
#> 3 cre     Plants  Green  algae Chlamydomonas reinhardtii  3
#> 4 csl     Plants  Green  algae Coccomyxa subellipsoidea 3
#> 5 cvr     Plants  Green  algae Chlorella variabilis    3
#> 6 mis     Plants  Green  algae Micromonas commoda     3
#> 7 mng     Plants  Green  algae Monoraphidium neglectum  3
#> 8 mpp     Plants  Green  algae Micromonas pusilla     3
#> 9 olu     Plants  Green  algae Ostreococcus lucimarinus 3
#> 10 ota    Plants  Green  algae Ostreococcus tauri     3
#> 11 vcn    Plants  Green  algae Volvox carteri f. nagariensis 3
#> 12 ccp     Plants  Red    algae Chondrus crispus (carragheen) 3
#> 13 cme    Plants  Red    algae Cyanidioschyzon merolae   3
#> 14 gsl    Plants  Red    algae Galdieria sulphuraria  3

```

1	apro	Plants	Green	algae	Auxenochlorella protothecoides	3
2	bpg	Plants	Green	algae	Bathycoccus prasinos	3
3	cre	Plants	Green	algae	Chlamydomonas reinhardtii	3
4	csl	Plants	Green	algae	Coccomyxa subellipsoidea	3
5	cvr	Plants	Green	algae	Chlorella variabilis	3
6	mis	Plants	Green	algae	Micromonas commoda	3
7	mng	Plants	Green	algae	Monoraphidium neglectum	3
8	mpp	Plants	Green	algae	Micromonas pusilla	3
9	olu	Plants	Green	algae	Ostreococcus lucimarinus	3
10	ota	Plants	Green	algae	Ostreococcus tauri	3
11	vcn	Plants	Green	algae	Volvox carteri f. nagariensis	3
12	ccp	Plants	Red	algae	Chondrus crispus (carragheen)	3
13	cme	Plants	Red	algae	Cyanidioschyzon merolae	3
14	gsl	Plants	Red	algae	Galdieria sulphuraria	3

4.3 Similarity comparison Eukaryotes

Comparison of two similarities

Load the similarities for pairs and comparison

```
n=length(meta_taxo$mDAG_Id)
n

[1] 884

dim(Sim_MSA_mDAG)

[1] 884 884

#aux1=base::rep(x=1:n, each=c(n:1))

aux=as_tibble(Sim_MSA_mDAG)
aux$mDag=names(aux)
aux=aux %>% pivot_longer(cols=`0313`:`0300`,
                           names_to="mDag_2",
                           values_to="Sim_MSA")

aux_2= aux %>% mutate(i=pmax(as.integer(mDag),
                               as.integer(mDag_2)),
                           j=pmin(as.integer(mDag),
                               as.integer(mDag_2))) %>% unite("ij",i:j) %>%
```

```

filter(duplicated(ij))

aux=as_tibble(Sim_Mun_mDAG)
aux$mDag=names(aux)
aux=aux %>% pivot_longer(cols=`0313`:`0300`,
                           names_to="mDag_2",
                           values_to="Sim_Mun")
aux_2 = aux_2 %>% left_join(aux)

Sim_comp=aux_2
rm(aux,aux_2)

```

Spearman and Pearson correlations

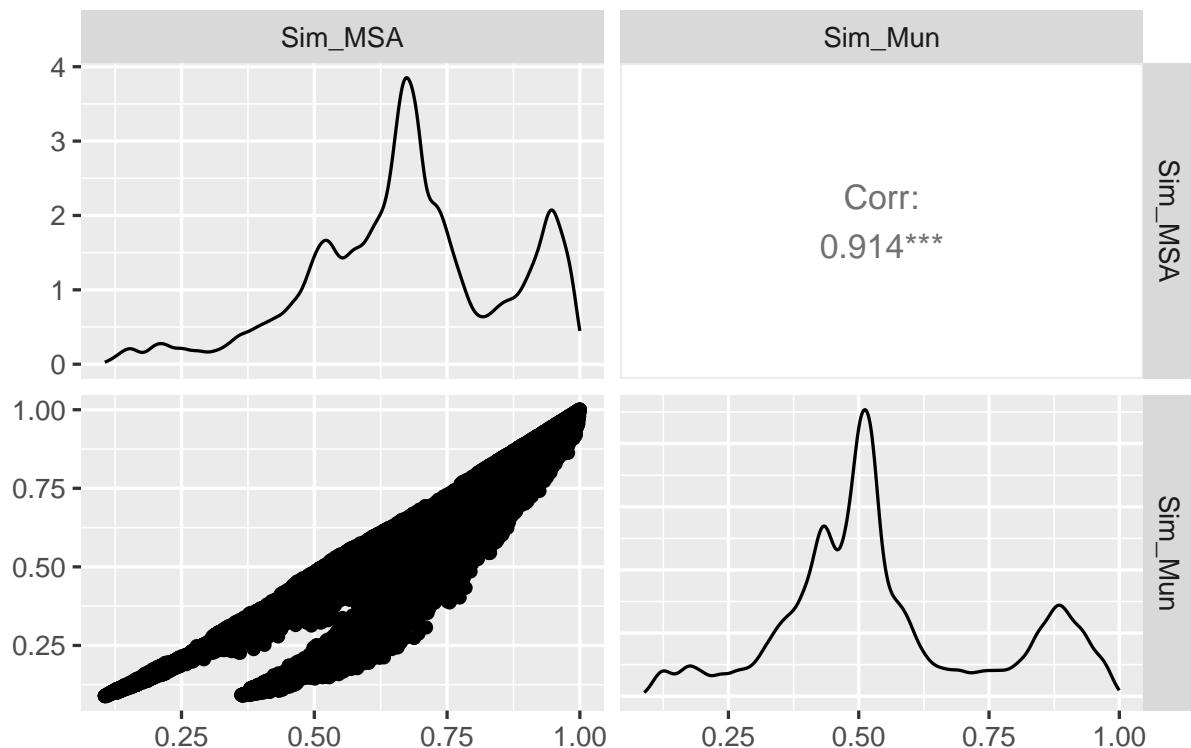
```
cor(Sim_comp[,c(3,5)],method="spearman")
```

	Sim_MSA	Sim_Mun
Sim_MSA	1.0000000	0.8902053
Sim_Mun	0.8902053	1.0000000

```
cor(Sim_comp[,c(3,5)],method="pearson")
```

	Sim_MSA	Sim_Mun
Sim_MSA	1.0000000	0.9138905
Sim_Mun	0.9138905	1.0000000

```
GGally::ggpairs(Sim_comp[,c(3,5)])
```



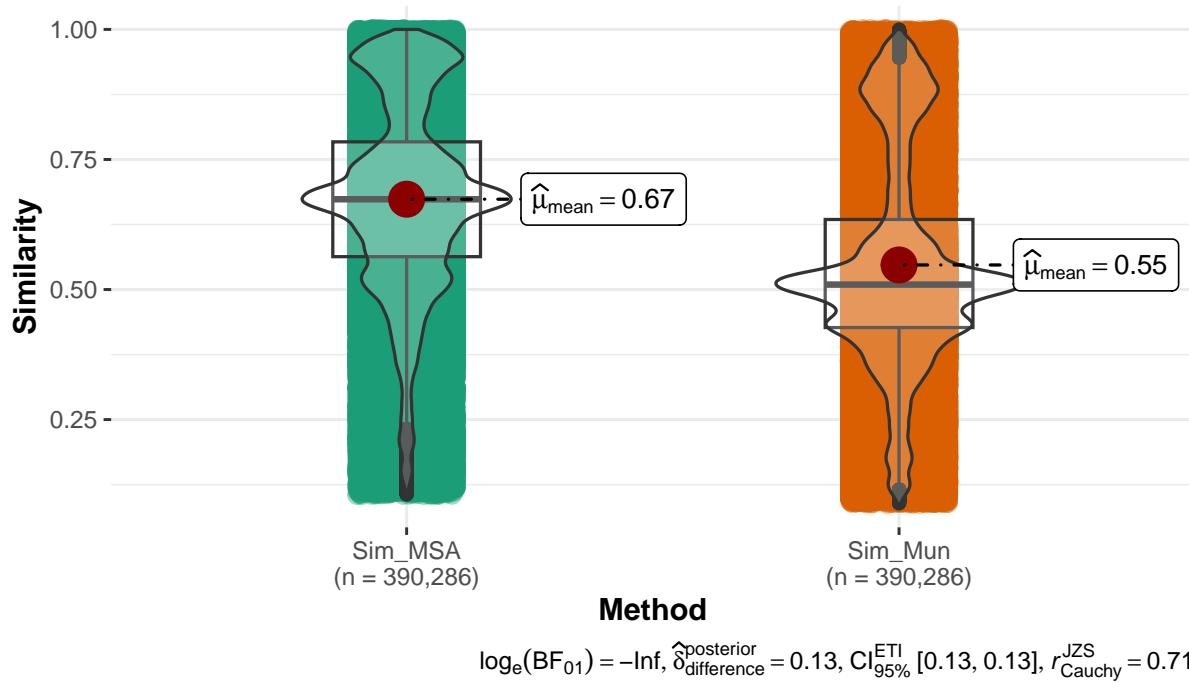
```

aux= Sim_comp%>% pivot_longer(
  cols=c(Sim_MSA,Sim_Mun),
  names_to="Method",
  values_to="Similarity")

ggstatsplot::ggbetweenstats(
  data = aux,
  x = Method,
  y = Similarity)

```

$$t_{\text{Welch}}(773000) = 290.27, p = 0.00, \hat{g}_{\text{Hedges}} = 0.66, \text{CI}_{95\%} [\text{NA}, \text{NA}], n_{\text{obs}} = 780,572$$



5 Graph

Some statistics for graphs

5.1 Read all graphs from a level of the experiment

Read all graphs from a level from experiment; for example individuals. We read only first (alphabetic) two graph

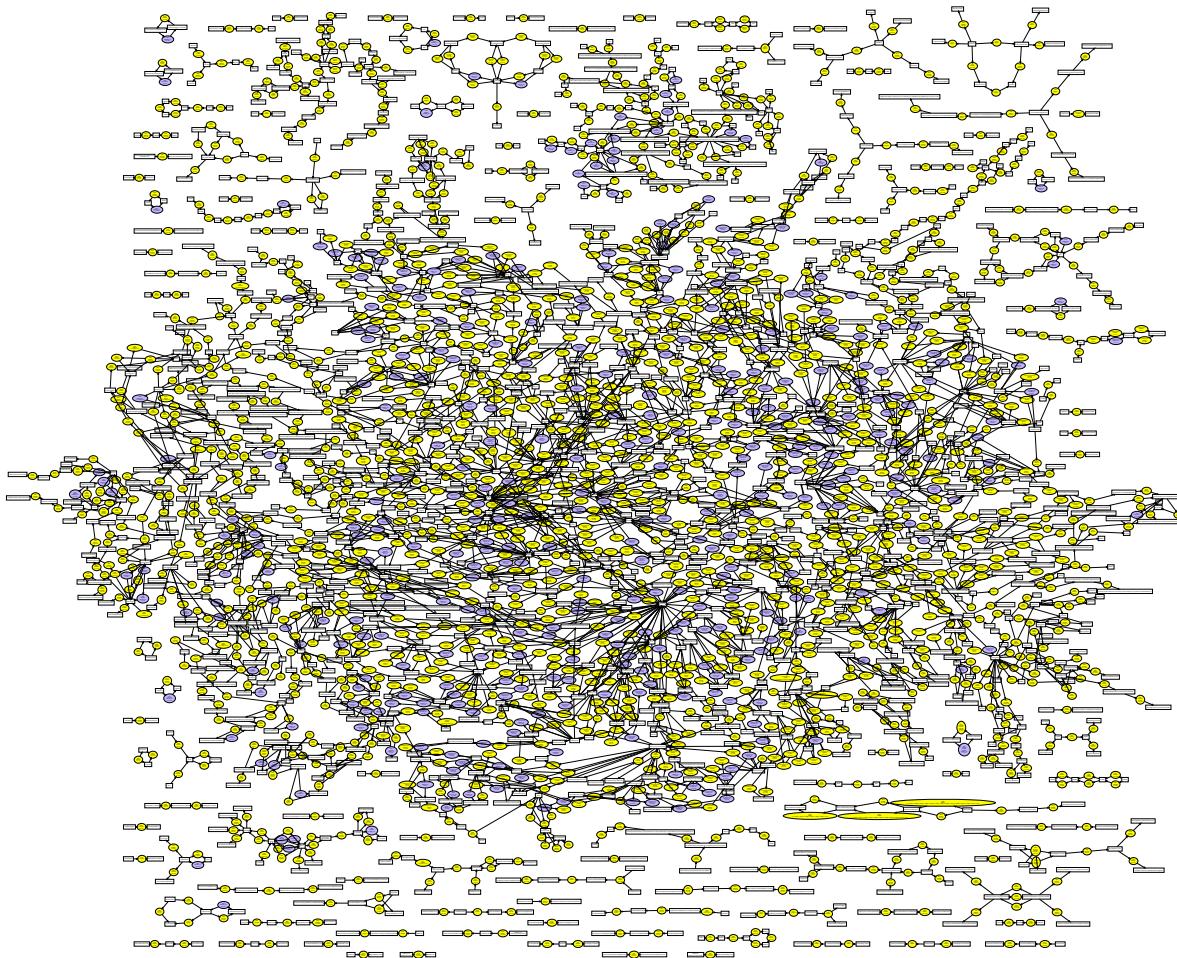
```
path_exp="data/result_bb261b6e-95c6-3e39-b82b-b68eea80e30b/data/"
list_names=dir(paste0(path_exp,"Individuals/"))

list_names= list_names[-1] # filter 0000_RefPw
length(list_names)
```

```
[1] 884
```

```
graphs_list=paste0(path_exp,"Individuals/",
                     list_names,"/",list_names,
                     "_MDAG.graphml")

knitr:::include_graphics(
  paste0(path_exp,"Individuals/cang/cang_RC.pdf"))
```



5.2 Graph statistics

```

read_mDAG=function(x) {DAG=read.graph(file=x,
                                         format="graphml")
  return(DAG)}
mDAG_componets=function(x) {
  sort(components(x,mode = "weak")$csizes,
       decreasing=TRUE)
}

compo_list=lapply(graphs_list,
                  FUN=function(x) {
                    gg=read_mDAG(x)

```

```

    aux=list(
      mDAG_componets=mDAG_componets(gg),
      degree_count=igraph::degree(gg,mode="total"))
    return(aux)
  )

names(compo_list)=list_names
n=max(sapply(compo_list,FUN=function(x) {length(x[[1]])}))
n

[1] 234

size_compo_list=lapply(compo_list,FUN=function(x) {
  return(c(x[[1]],rep(NA,n-length(x[[1]]))))}

aux=do.call(bind_cols,size_compo_list)
aux2=pivot_longer(aux,aaf:zvi,names_to="Organism",
                  values_to="csize") %>%
  arrange(Organism,-csize)
aux2$index=rep(1:n,times=dim(aux)[2])
aux2=aux2 %>%
  left_join(meta_taxo,by="Organism")

Organism=names(compo_list)
big_MBB=function(org){
  x=Results %>% filter(Organism==org)
  x=as.character(x[1,5:dim(Results)[2]])
  x=x[x!="NA"]
  tt=sort(table(x),decreasing=TRUE)
  return(tt)
}
big_MBB_list= lapply(Organism,FUN=function(x) big_MBB(x))
nMBB=max(sapply(big_MBB_list,FUN=function(x) length(x)))
nMBB

[1] 1041

big_MBB_list=lapply(big_MBB_list,
                     FUN=function(x){
  x=c(x,rep(NA,nMBB-length(x)))
  return(x)}
)
names(big_MBB_list)=Organism

```

```

big_MBB_list=do.call(bind_cols,big_MBB_list)

kMBB=nrow(big_MBB_list)
index=rep(1:kMBB,times=length(Organism))

big_MBB_list2=pivot_longer(big_MBB_list,
                           cols=names(big_MBB_list),
                           values_to = "MBBszie",
                           names_to = "Organism") %>%
  arrange(Organism,-MBBszie) %>%
  mutate(index=index) %>%
  left_join(meta_taxo,by="Organism")

```

5.2.1 Sizes of MBB for each mDAG

```

COLOR_KINGDOM=c("red","green","yellow","black")
colors_kingdom=big_MBB_list2 %>%
  select(Organism,Kingdom) %>%
  distinct()

names(COLOR_KINGDOM)=sort(unique(colors_kingdom$Kingdom))

p0<-ggplot(data=big_MBB_list2) +
  geom_line(mapping=aes(x=index,
                        y=MBBszie,
                        group = Organism,
                        color=Kingdom)) +
  scale_y_continuous(trans='log10') +
  scale_x_continuous(trans='identity') +
  scale_color_manual(values =COLOR_KINGDOM[colors_kingdom$Kingdom]) +
  ggtitle("Plot log-identity of MBB \n decreasing index.") +
  ylab("Log10 MBB size") + xlab("Index")

p1<- ggplot(data=big_MBB_list2) +
  geom_line(mapping=aes(x=index,
                        y=MBBszie,
                        group = Organism,color=Kingdom),
            na.rm=TRUE) +
  scale_x_continuous(trans='log10') +
  scale_y_continuous(trans='log10') +
  scale_color_manual(values =COLOR_KINGDOM[colors_kingdom$Kingdom]) +

```

```

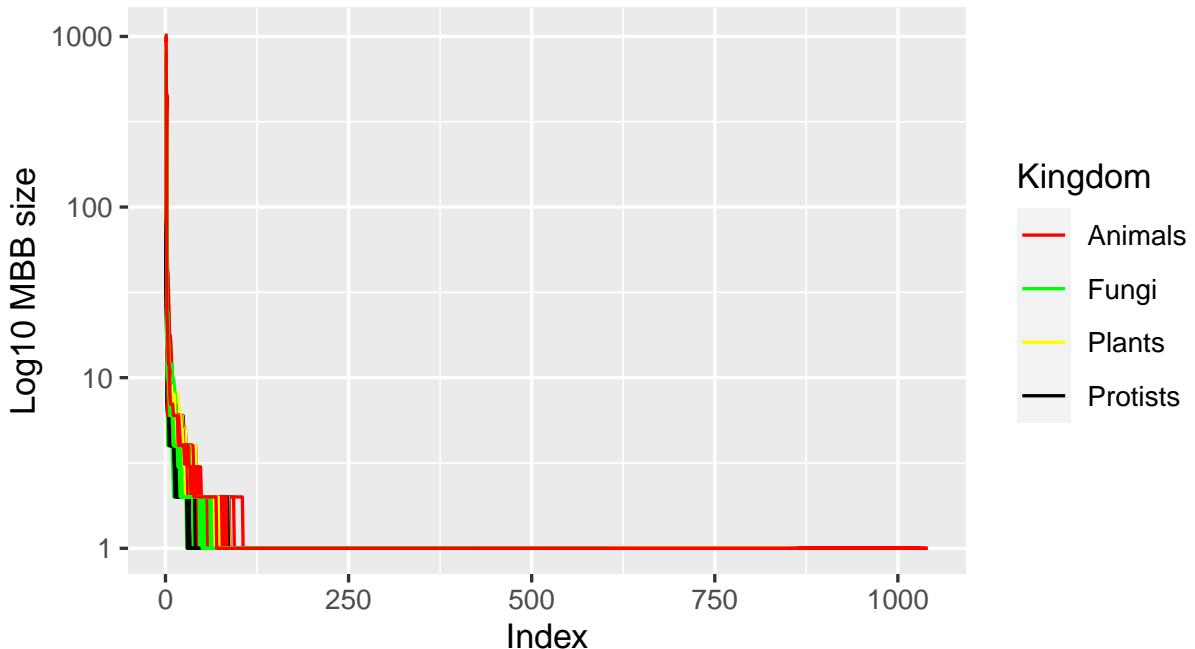
ggtitle("Plot log10-log10identity of MBB \n decreasing index.") +
  ylab("Log10 MBB size") + xlab("Log10 Index")

p2<- ggplot(data=big_MBB_list2) +
  geom_line(mapping=aes(x=index,
                        y=MBBsize,
                        group = Organism,
                        color=Kingdom),
            na.rm=TRUE)+ 
  scale_x_continuous(trans="identity") +
  scale_y_continuous(trans="identity") +
  ylim(0,1039) +
  ggtitle("Plot of MBB sized decreasing index.") +
  ylab("MBB size") + xlab("Index") +
  scale_color_manual(values =COLOR_KINGDOM[colors_kingdom$Kingdom])

```

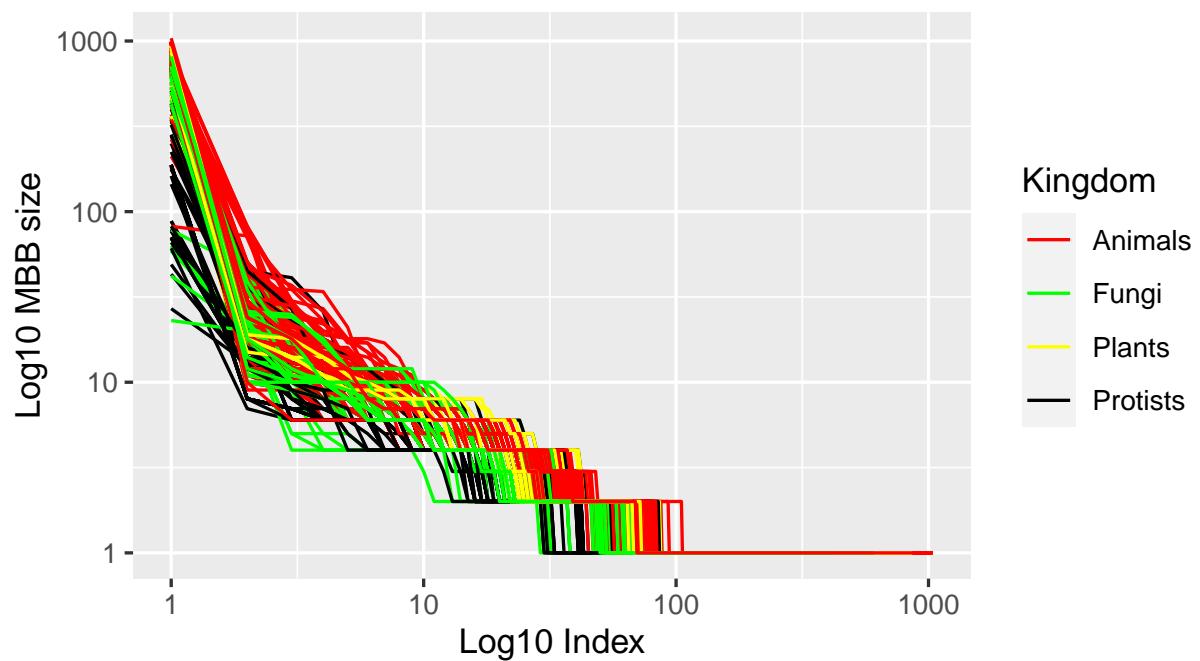
p0

**Plot log–identity of MBB
decreasing index.**



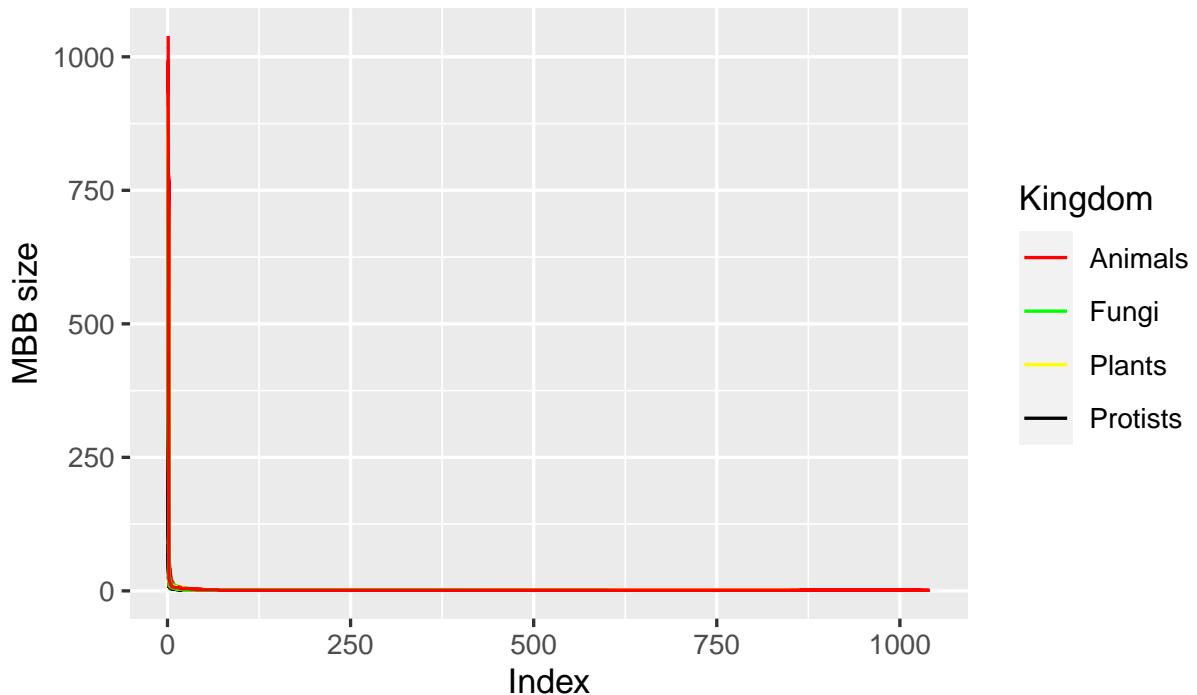
p1

Plot log10–log10identity of MBB
decreasing index.



p2

Plot of MBB sized decreasing index.



5.2.2 Sizes of weak components for each mDAG

```
COLOR_KINGDOM=c("red","yellow","green","yellow","black")
colors_kingdom=aux2%>% select(Organism,Kingdom) %>% distinct()
names(COLOR_KINGDOM)=sort(unique(colors_kingdom$Kingdom))

p0<-ggplot(data=aux2) +
  geom_line(mapping=aes(x=index,
                        y=csizes,
                        group = Organism,
                        color=Kingdom),
            na.rm=TRUE) +
  scale_x_continuous(trans='log10') +
  scale_y_continuous(trans='identity') +
  scale_color_manual(values =COLOR_KINGDOM[colors_kingdom$Kingdom]) +
  ggtitle("Plot log-identity of size weak components decreasing index.") +
  ylab("Log10 Weak componente size") + xlab("Index")

p1<- ggplot(data=aux2) +
```

```

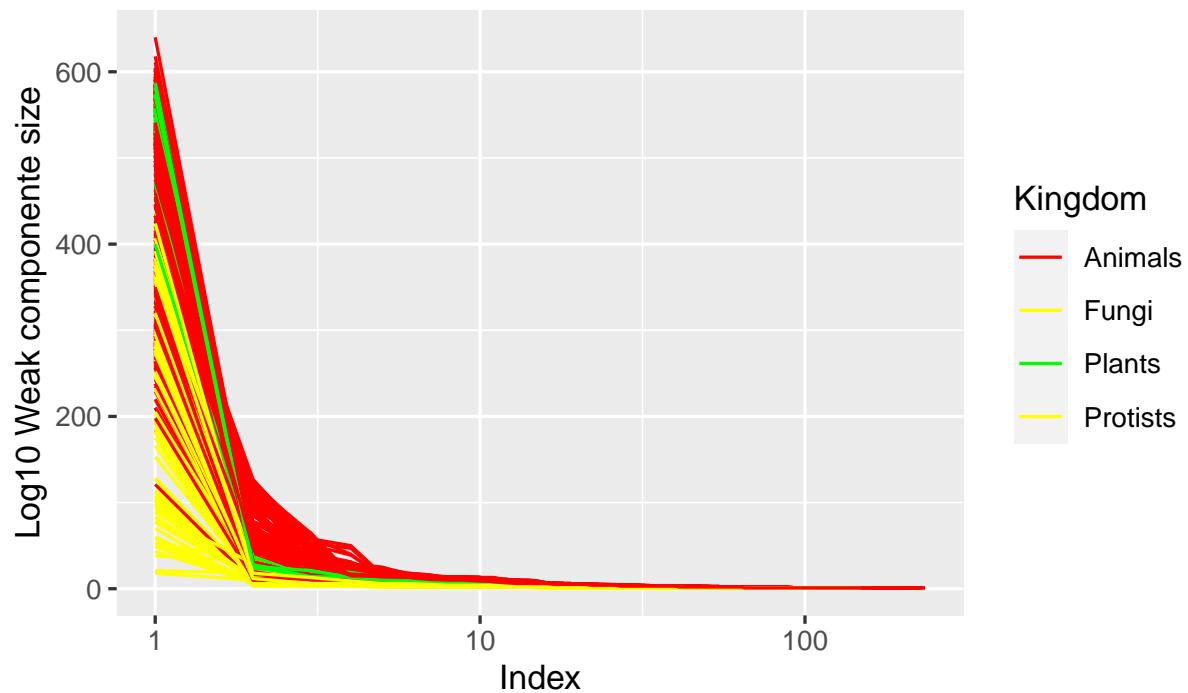
geom_line(mapping=aes(x=index,
                      y=csize, group = Organism,
                      color=Kingdom),
          na.rm=TRUE) +
scale_y_continuous(trans='log10') +
scale_x_continuous(trans='log10') +
scale_color_manual(values =COLOR_KINGDOM[colors_kingdom$Kingdom])+
ggtitle("Plot log-log of size weak components decreasing index.") +
ylab("Log10 weak component size") + xlab("Log10 Index")

p2<- ggplot(data=aux2) +
  geom_line(mapping=aes(x=index,
                        y=csize, group = Organism,
                        color=Kingdom),
            na.rm=TRUE) +
  scale_x_continuous(trans="identity") +
  scale_y_continuous(trans="identity") +
  ylim(0,1039)+
  ggtitle("Plot of size weak components decreasing index.")+
  ylab("Weak components size") + xlab("Index")+
  scale_color_manual(values =COLOR_KINGDOM[colors_kingdom$Kingdom])

```

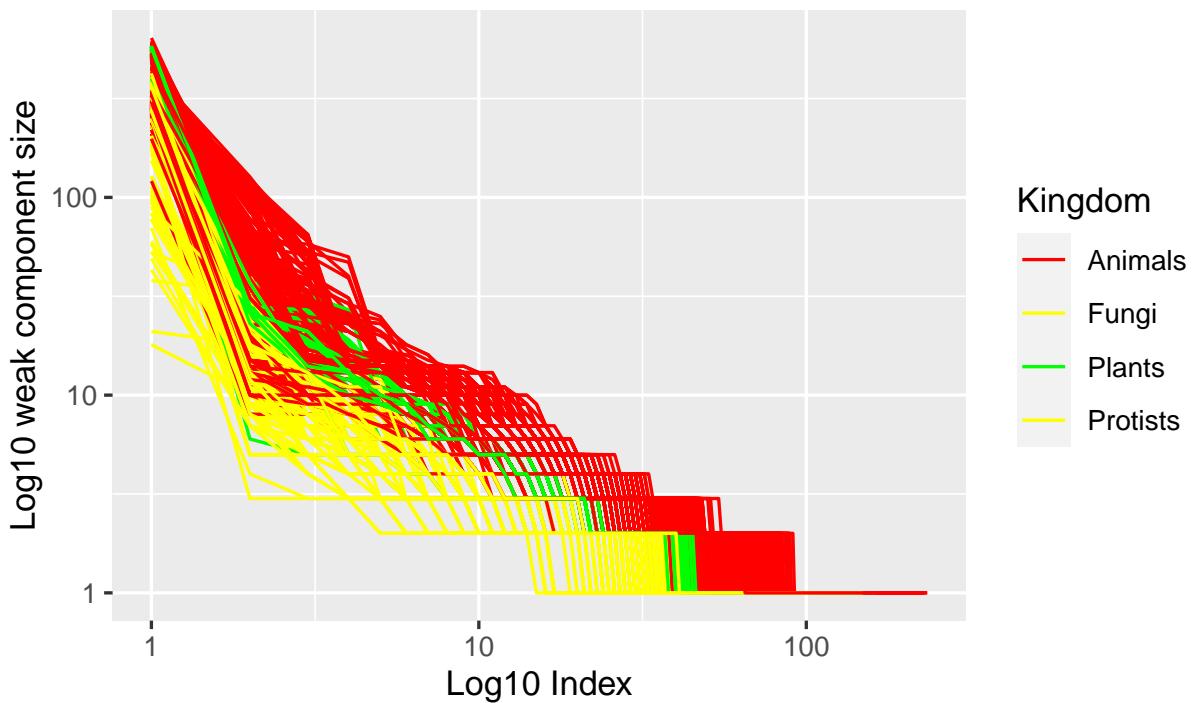
p0

Plot log–identity of size weak components decreasing index.



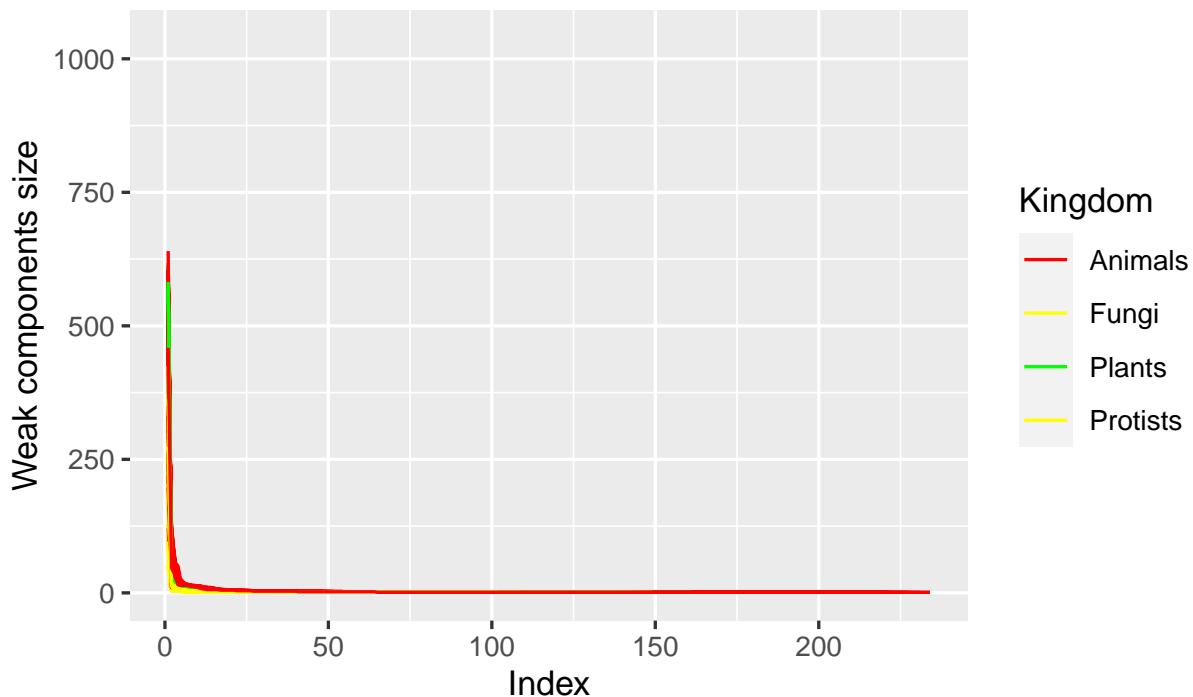
p1

Plot log–log of size weak components decreasing index.



p2

Plot of size weak components decreasing index.



```

data2=big_MBB_list2 %>% filter(index %in% 2:20)
p3<- ggplot(data=data2) +
  geom_line(mapping=aes(x=index,
                        y=MBBsize,
                        group = Organism,
                        color=Kingdom),
            na.rm=TRUE) +
  scale_x_continuous(trans="identity") +
  scale_y_continuous(trans="identity") +
  ylim(0,25) +
  ggtitle("Plot of size weak components decreasing index 2 to 20.") +
  ylab("Weak components size") + xlab("Index") +
  scale_color_manual(values =COLOR_KINGDOM[colors_kingdom$Kingdom])

```

p3