Selected MAGs

```
library(tidyverse)
library(distillR)
library(ape)
library(vegan)
library(ggrepel)
library(ggh4x)
library(broom)
```

```
gene_annotations <- read_tsv("data/gene_annotations.tsv.xz") %>%
  filter(gene!="gene") %>%
  mutate(ec=str_c("[EC:",ec,"]")) %>%
  mutate(genome=substr(gene, 1, 7))

genome_metadata <- read_csv("data/genome_metadata.csv")
```

```
gene_distillation <- distill(gene_annotations,GIFT_db, genomecol = 14, annotcol = c(5,6), verbosity=F)
```

```
##
## Identifiers in the annotation table: 2294
## Identifiers in the database: 1547
## Identifiers in both: 204
## Percentage of annotation table identifiers used for distillation: 8.89%
## Percentage of database identifiers used for distillation: 13.19%
```

```
gift_elements <- to.elements(gene_distillation,GIFT_db)
```

```
gift_elements %>%
    as.data.frame() %>%
    rownames_to_column(var="genome") %>%
    pivot_longer(!genome,names_to="trait",values_to="gift") %>%
    inner_join(genome_metadata,by="genome") %>%
    mutate(functionid = substr(trait, 1, 3)) %>%
    mutate(trait = case_when(
      trait %in% GIFT_db$Code_element ~ GIFT_db$Element[match(trait, GIFT_db$Code_element)],
      TRUE ~ trait
    )) %>%
    mutate(functionid = case_when(
      functionid %in% GIFT_db$Code_function ~ GIFT_db$Function[match(functionid, GIFT_db$Code_function)]
      TRUE ~ functionid
    )) %>%
    mutate(trait=factor(trait,levels=unique(GIFT_db$Element))) %>%
    mutate(functionid=factor(functionid,levels=unique(GIFT_db$Function))) %>%
    ggplot(aes(x=genome,y=trait,fill=gift)) +
        geom_tile(colour="white", linewidth=0.2)+
        scale_fill_gradientn(colours=rev(c("#d53e4f", "#f46d43", "#fdae61", "#fee08b", "#e6f598", "#abd
        facet_nested(functionid ~ group + farm, scales="free",space="free") +
        theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1),
              strip.text.y = element_text(angle = 0)) +
        labs(y="Traits",x="Samples",fill="GIFT")
```


resilient
susceptible
A
B
A
B
Nucleic acid biosynthesis
Amino acid biosynthesis
Amino acid derivative biosynthesis
SCFA biosynthesis
Organic anion biosynthesis
Vitamin biosynthesis
Aromatic compound biosynthesis
Metallophore biosynthesis
Antibiotic biosynthesis
Lipid degradation
Polysaccharide degradation
Sugar degradation
Protein degradation
Amino acid degradation
Nitrogen compound degradation
Alcohol degradation
Xenobiotic degradation
Antibiotic degradation
Cellular structure
Appendages
Spore
Traits
GIFT
1.00
0.75
0.50
0.25
0.00
MAG0001
MAG0052
MAG0119
MAG0622
MAG0743
MAG0882
MAG1019
MAG1020
MAG1028
MAG0059
MAG0301
MAG0439
MAG0461
MAG0611
MAG0677
MAG0851
MAG0877
MAG0880
MAG0905
MAG0959
MAG0983
MAG1090
MAG1114
MAG0373
MAG0626
Samples

Functional differences

Ideally, a mixed effects modelling should be used, but susceptible MAGs are too few for any reasonable modelling.

```
functional_differences <- gift_elements %>%
  as.data.frame() %>%
  rownames_to_column("genome")  %>%
  inner_join(genome_metadata,by="genome") %>%
  pivot_longer(-c(genome, group, farm), names_to = "trait", values_to = "value") %>%
  nest(data = -trait) %>%
  mutate(
    fit  = map(data, ~lm(value ~ group, data = .x)),
    tid  = map(fit, tidy)
  ) %>%
  unnest(tid) %>%
  filter(term == "groupsusceptible") %>%
  mutate(p_value_adj = p.adjust(p.value, method = "bonferroni")) %>%
  select(id   = trait,
         estimate,
         p.value,
         p_value_adj) %>%
  arrange(p.value)
```

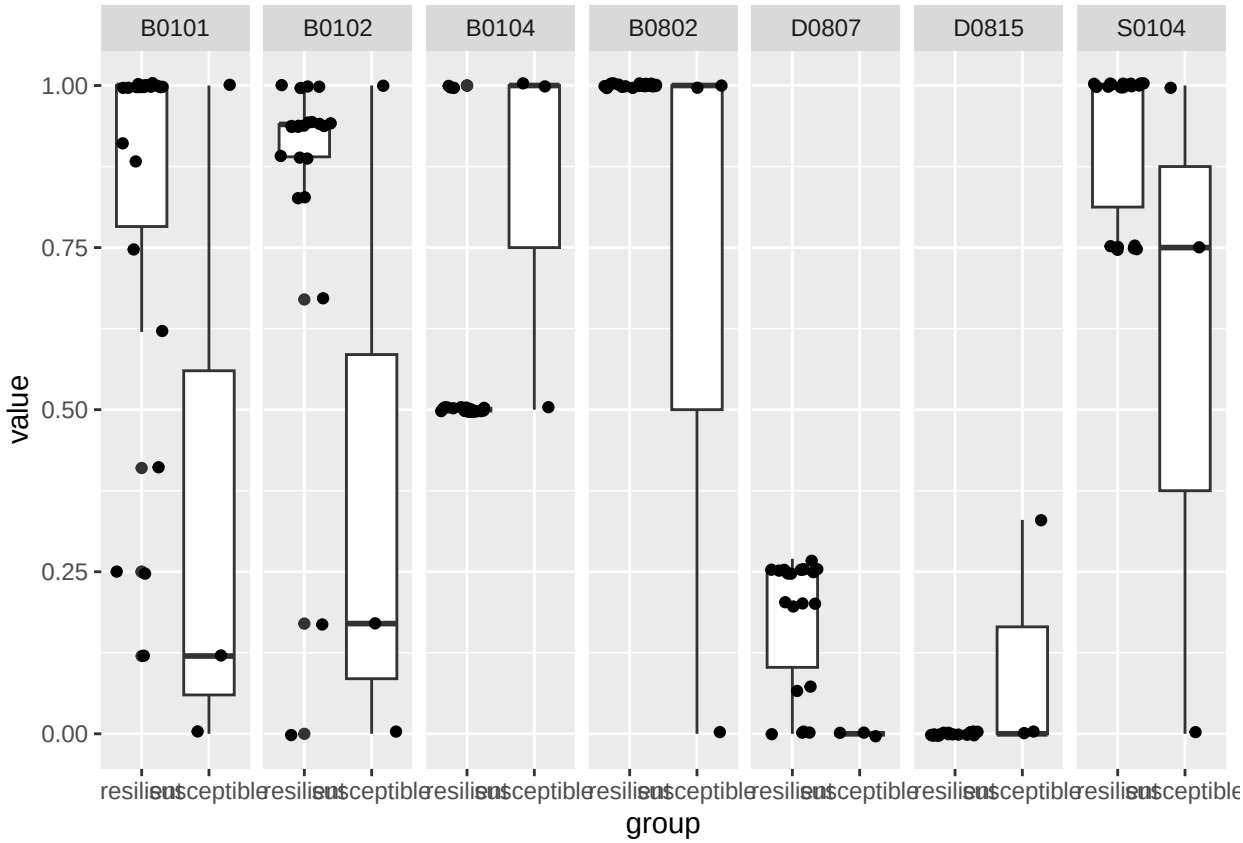
```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `tid = map(fit, tidy)`.
## Caused by warning in `summary.lm()`:
## ! essentially perfect fit: summary may be unreliable
```

```
functional_differences %>%
  filter(p.value<0.05)
```

```
## # A tibble: 7 x 4
##   id    estimate p.value p_value_adj
##   <chr>    <dbl>   <dbl>       <dbl>
## 1 B0802   -0.333 0.00416       0.587
## 2 D0815    0.110 0.00416       0.587
## 3 S0104   -0.348 0.00626       0.882
## 4 D0807   -0.180 0.00627       0.884
## 5 B0102   -0.454 0.0190        1
## 6 B0104    0.265 0.0316        1
## 7 B0101   -0.453 0.0326        1
```

None of the GIFTs yield significant differences after Bonferroni adjustment.

```
gift_elements %>%
  as.data.frame() %>%
  select(functional_differences %>% filter(p.value<0.05) %>% pull(id)) %>%
  rownames_to_column(var="genome") %>%
  pivot_longer(!genome,names_to = "trait", values_to = "value") %>%
  inner_join(genome_metadata,by="genome") %>%
  ggplot(aes(x=group, y=value, group=group))+
    geom_boxplot() +
    geom_jitter() +
    facet_grid(. ~ trait, scales="free",space="free")
```

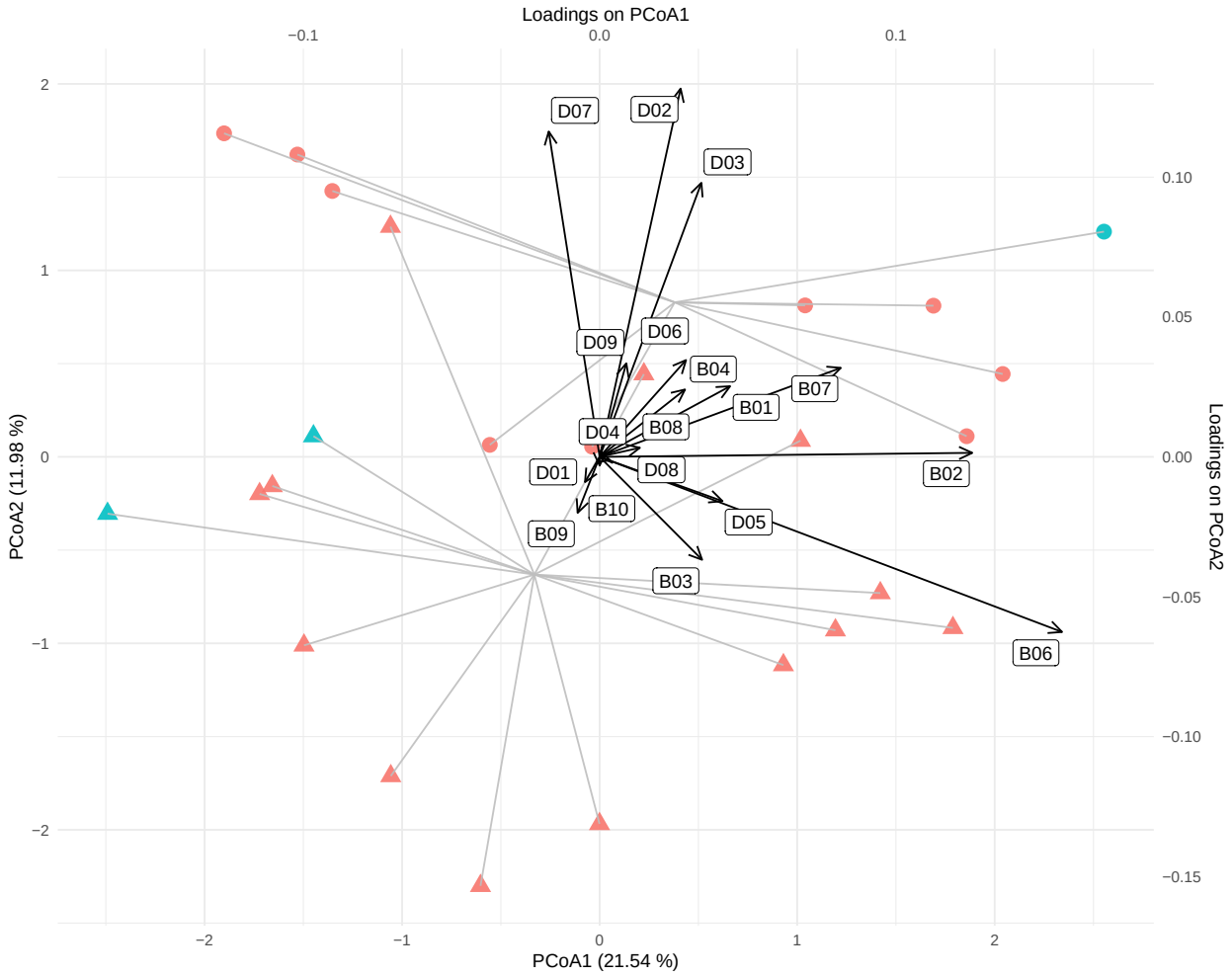


Functional ordination

```
gift_pcoa <- gift_elements %>%
    as.data.frame() %>%
    vegdist(method="euclidean") %>%
    pcoa()

gift_pcoa_rel_eigen <- gift_pcoa$values$Relative_eig[1:10]


# Get genome positions
gift_pcoa_vectors <- gift_pcoa$vectors %>% #extract vectors
  as.data.frame() %>%
  select(Axis.1,Axis.2) # keep the first 2 axes

gift_pcoa_eigenvalues <- gift_pcoa$values$Eigenvalues[c(1,2)]

gift_pcoa_gifts <- cov(gift_elements, scale(gift_pcoa_vectors)) %*% diag((gift_pcoa_eigenvalues/(nrow(g
  as.data.frame() %>%
  rename(Axis.1=1,Axis.2=2) %>%
  rownames_to_column(var="label") %>%
  #get function summary vectors
  mutate(func=substr(label,1,3)) %>%
  group_by(func) %>%
  summarise(Axis.1=mean(Axis.1),
```

```
            Axis.2=mean(Axis.2)) %>%
  rename(label=func) %>%
  filter(!label %in% c("S01","S02","S03"))
```

```
scale <- 15 # scale for vector loadings
gift_pcoa_vectors %>%
  rownames_to_column(var="genome") %>%
  inner_join(genome_metadata,by="genome") %>%
  group_by(farm) %>%
  mutate(x_cen = mean(Axis.1, na.rm = TRUE)) %>%
  mutate(y_cen = mean(Axis.2, na.rm = TRUE)) %>%
  ungroup() %>%
  ggplot() +
      #genome positions
      #scale_color_manual(values=order_colors)+
      geom_point(aes(x=Axis.1,y=Axis.2, color=group, shape=farm), alpha=0.9, size=4) +
      geom_segment(aes(x = x_cen, y = y_cen, xend = Axis.1, yend = Axis.2, group=farm), alpha = 0.9, co
      #scale_color_manual(values=phylum_colors) +
      scale_size_continuous(range = c(0.1,5)) +
      #loading positions
      geom_segment(data=gift_pcoa_gifts,
                   aes(x=0, y=0, xend=Axis.1 * scale, yend=Axis.2 * scale),
                    arrow = arrow(length = unit(0.3, "cm"),
                    type = "open",
                    angle = 25),
                    linewidth = 0.5,
                    color = "black") +
     #Primary and secondary scale adjustments
     scale_x_continuous(name = paste0("PCoA1 (",round(gift_pcoa_rel_eigen[1]*100, digits = 2), " %)"),
                        sec.axis = sec_axis(~ . / scale, name = "Loadings on PCoA1")
            ) +
     scale_y_continuous(name = paste0("PCoA2 (",round(gift_pcoa_rel_eigen[2]*100, digits = 2), " %)"),
                        sec.axis = sec_axis(~ . / scale, name = "Loadings on PCoA2")
            ) +
    geom_label_repel(data = gift_pcoa_gifts,
                     aes(label = label, x = Axis.1 * scale, y = Axis.2 * scale),
                     segment.color = 'transparent') +
    theme_minimal() +
    theme(legend.position = "none")
```



There is a huge effect of the farm in the functional profile of the MAGs. The lines connect all the MAGs from each farm (dot and triangle).