Elo-rating analysis on Microbes Atlas Project dataset

Angel Rain

2025-06-10

Contents

1 Load data

1.1 Load packages

```
rm(list=ls())
library(stringr)
library(ggplot2)
library(stringr)
library(tidyr)
library(dplyr)
library(kableExtra)
library(cowplot)
cbp1 <- c("#999999","#0072B2","#D55E00", "#E69F00", "#56B4E9", "#009E73","#F0E442", "#CC79A7","#293352"
```

1.2 Load Elo-rating results

```
elo.files<-list.files(path="../data/Elo.estimations/",pattern="average_MultiElo_*")

list.tmp<-list()

for (i in 1:length(elo.files)){
  df.tmp<-read.csv(paste0("../data/Elo.estimations/",elo.files[i]))
  df.tmp$Category<-str_split_fixed(elo.files[i],"_",4)[,3]
  df.tmp<-df.tmp[,c(2:6)]
  names(df.tmp)<-c("player_id","biome","mean_rating","sd_rating","Category")
  list.tmp[[i]]<-df.tmp
  rm(df.tmp) # save some memory
}

#Combine datasets
df.all<-do.call(rbind,list.tmp)
rm(list.tmp)

df.agg.elo<-df.all
rm(elo.files,df.all,i)
```

1.3 Filter out non-bacteria OTUs

```
#Load taxonomy
# 1) Read the raw file (no header).  Replace "myfile.tsv" with your actual filename.
df_raw <- read.delim("../data/otus.info",
                     header    = T,
                     sep       = "\t",
                     stringsAsFactors = FALSE,
                     quote     = "")  # turn off any quoting behavior
str(df_raw)
```

```
## 'data.frame':    597954 obs. of  19 variables:
```

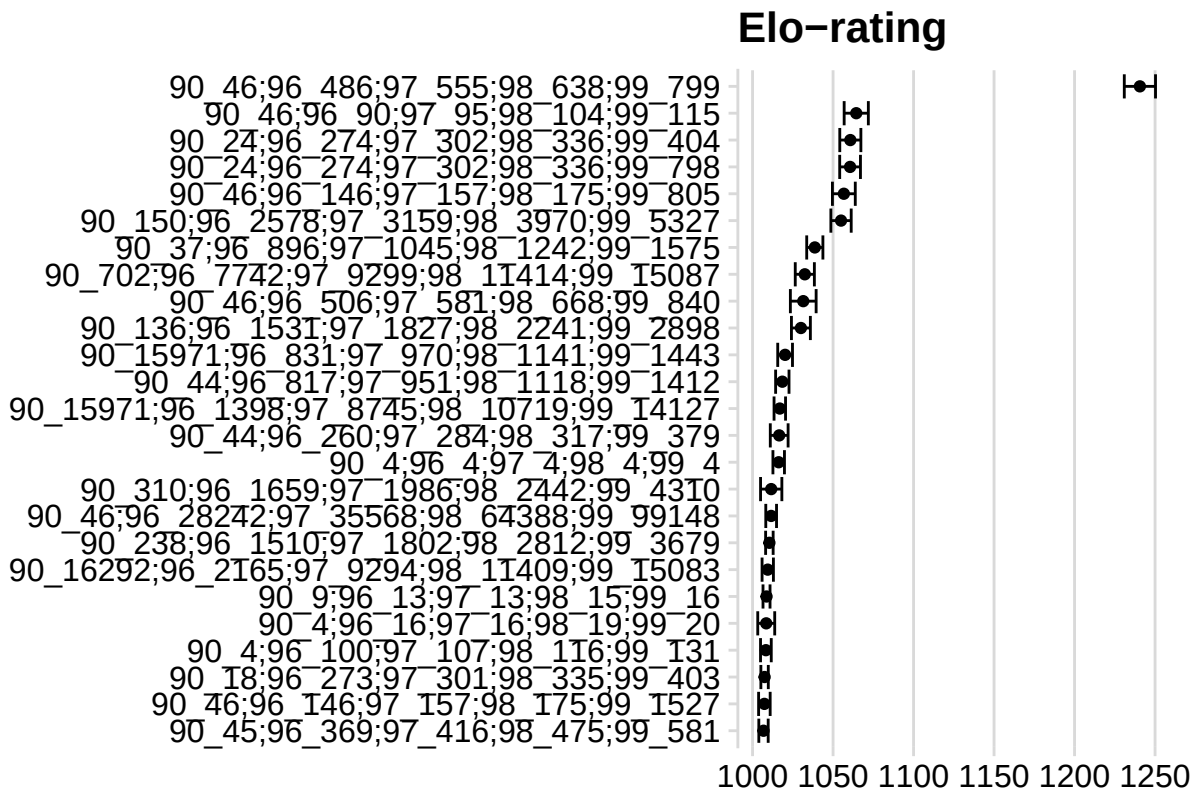
player_id	biome	Category	mean_rating	sd_rating
90_1;96_1;97_1;98_1;99_1	animal.3	Corrected	802.0144	8.0890303
90_1;96_1;97_1;98_1;99_1	plant.2	Classic	999.9884	0.0280091
90_1;96_1;97_1;98_1;99_1	freshwater.3	Corrected	793.9269	6.7698969
90_1;96_1;97_1;98_1;99_1	freshwater.2	Classic	1000.0020	0.0504772
90_1;96_1;97_1;98_1;99_1	freshwater.1	Classic	999.9862	0.0472560
90_1;96_1;97_1;98_1;99_1	animal.2	Classic	999.9971	0.0178922

```
##  $ OTU          : chr  "90_17776;96_71281;97_92606;98_125911;99_193128" "90_17776;96_71281;97_92606;9
##  $ Tax          : chr  "Archaea" "Archaea" "Archaea" "Archaea" ...
##  $ SpeciesRep   : chr  "" "" "" "" ...
##  $ SeqCount     : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ GoldCount    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ GenomeCount  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ TypeStrains  : chr  "" "" "" "" ...
##  $ Strains      : chr  "" "" "" "" ...
##  $ Genomes      : chr  "" "" "" "" ...
##  $ GoldSeqs     : chr  "" "" "" "" ...
##  $ Aliases      : chr  "" "" "" "" ...
##  $ GoldHit      : chr  "NR_074195:1..1470" "NR_074195:1..1470" "NR_074195:1..1470" "NR_074195:1..147
##  $ GoldID       : int  725 725 725 725 725 569 569 569 569 569 ...
##  $ GoldScore    : num  0.754 0.754 0.754 0.754 0.754 ...
##  $ RepSpecies   : chr  "" "" "" "" ...
##  $ Taxaname     : chr  "Archaea" "Archaea" "Archaea" "Archaea" ...
##  $ OrigTax      : chr  "Archaea" "Archaea" "Archaea" "Archaea" ...
##  $ RepSequenceID: chr  "KC471280:1..1464" "KC471280:1..1464" "KC471280:1..1464" "KC471280:1..1464" .
##  $ RepSequence  : chr  "TACCCGTTGATCCTGCGGGAGGTCGCTGCTATCAGAATTCGACTTAACTAAGCTAGTTCTGGGGCTTCTTCGGAAG
```

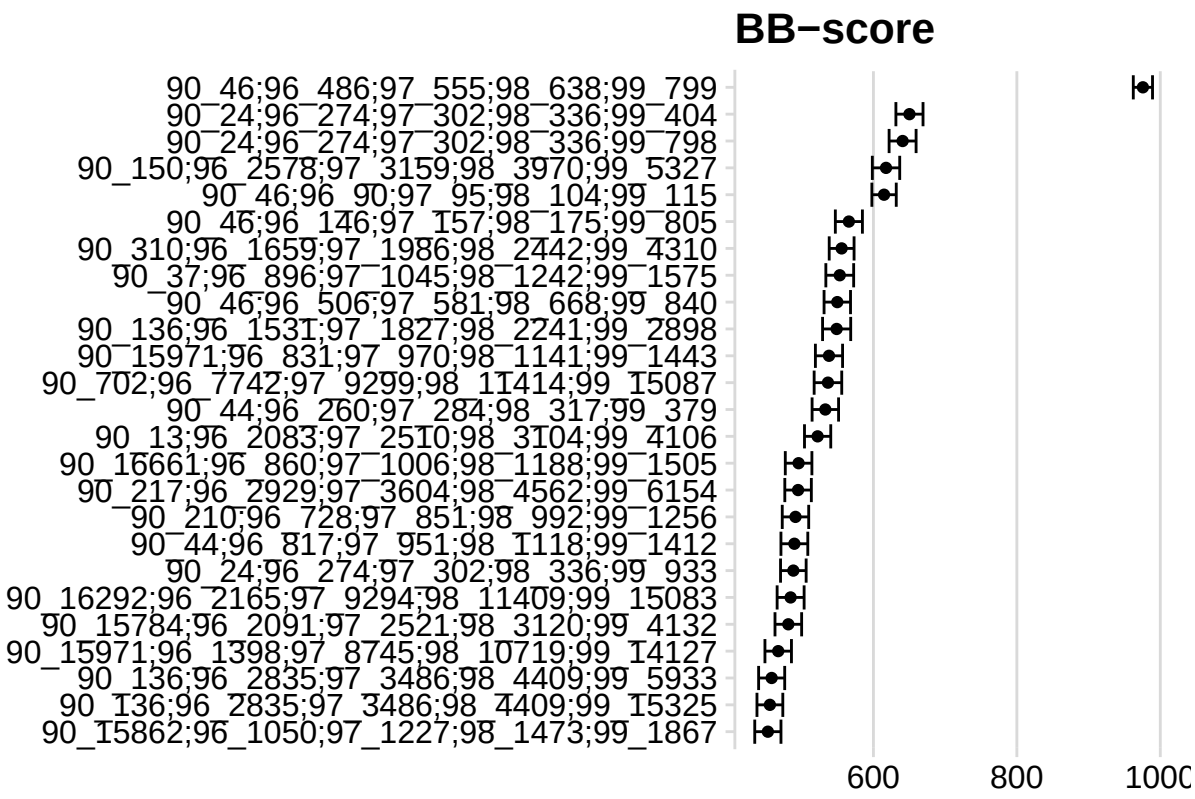
```
#set taxonomy
taxonomy.table<- df_raw[c("OTU","Tax","RepSequence")]

taxonomy.table <- df_raw %>%
  select(OTU, Tax, RepSequence) %>%
  separate(
    col   = Tax,
    into  = c("Domain", "Phylum", "Class", "Order", "Family", "Genus", "Species"),
    sep   = ";",
    fill  = "right",   # missing levels become NA
    extra = "drop"     # drop any beyond Species
  )

### Filter non-bacteria hits

data.all<-merge(df.agg.elo,taxonomy.table,by.x="player_id", by.y="OTU")
to.be.removed<-str_detect(data.all$Domain,"Eukaryota")
data.all.filtered<-data.all[!to.be.removed,]

df.agg.elo<-data.all.filtered[,c("player_id","biome","Category","mean_rating","sd_rating" )]
kbl(head(df.agg.elo))%>%
  kable_styling(full_width = FALSE, position = "center")
```

#Temporal step for displaying Ratings estimations Files loades correspond to the average and standard deviation from the 1000 iterations

```r
#select one biome as example
df.agg.elo%>%
  filter(biome == "animal.1") %>%
  filter(Category == "Classic") %>%
  arrange(desc(mean_rating)) %>%                # Sort by rating in descending order
  head(25)%>%
  ggplot(aes(x=mean_rating,y=reorder(player_id,mean_rating)))+
  geom_point()+
  geom_errorbar(aes(xmin=mean_rating-sd_rating, xmax=mean_rating+sd_rating))+
  theme_minimal_vgrid()+
  labs(x="",y="",title="Elo-rating")
```




```r
df.agg.elo%>%
  filter(biome == "animal.1") %>%
  filter(Category == "Corrected") %>%
  arrange(desc(mean_rating)) %>%                # Sort by rating in descending order
  head(25)%>%
  ggplot(aes(x=mean_rating,y=reorder(player_id,mean_rating)))+
  geom_point()+
  geom_errorbar(aes(xmin=mean_rating-sd_rating, xmax=mean_rating+sd_rating))+
  theme_minimal_vgrid()+
  labs(x="",y="",title="BB-score")
```



1.4 Save pre-processed data

```
df_MAP_Elo<-df.agg.elo
rm(list = setdiff(ls(),c("df_MAP_Elo")))
str(df_MAP_Elo)
```

```
## 'data.frame':    1816294 obs. of  5 variables:
##  $ player_id  : chr  "90_1;96_1;97_1;98_1;99_1" "90_1;96_1;97_1;98_1;99_1" "90_1;96_1;97_1;98_1;99_1
##  $ biome      : chr  "animal.3" "plant.2" "freshwater.3" "freshwater.2" ...
##  $ Category   : chr  "Corrected" "Classic" "Corrected" "Classic" ...
##  $ mean_rating: num  802 1000 794 1000 1000 ...
##  $ sd_rating  : num  8.089 0.028 6.7699 0.0505 0.0473 ...
```

```
save.image(file = "Elo_MAPdata_from_Script02.RData")
```