Results of PangoVis

Devan Becker

2021-04-09

Load Packages and Data

```
# Packages that Art hates
library(dplyr)
library(tidyr)
library(ggplot2)
library(stringr)
library(here)

dirich <- TRUE #params$dirich

# Read in CSV files
csvs <- list.files(here("data/", "pangolineages"),
    pattern = ifelse(dirich, "*_d.csv", "*.csv"),
    full.names = TRUE)

# Remove any copies
csvs <- csvs[!grepl("-1", csvs)]

# Bring them into one data frame
lins <- bind_rows(lapply(csvs, read.csv))

# Taxon is encoded as _ACCSESSIONNUMBER.ID, split into ACCESSIONNUMBER and ID
lins <- lins %>%
    separate(col = "taxon", sep = "\\.",
        into = c("taxon", "sample")) %>%
    mutate(taxon = str_replace(taxon, "\\_", ""))

badlins <- table(lins$taxon)
badlins <- names(badlins[which(badlins < 1000)])
cat(length(badlins), " runs were removed for having too few samples.")
```

```
## 11  runs were removed for having too few samples.
```

```
lins <- filter(lins, !taxon %in% badlins)

#### Visualize the uncertainty in the base calls ----
taxons <- unique(lins$taxon)
length(taxons)
```

```
## [1] 82
```

Abstract Info

```
summs <- lins %>%
    group_by(taxon) %>%
    summarise(
        maxperc = mean(lineage == names(sort(table(lineage),
            decreasing = TRUE)[1]),
        uniques = length(unique(lineage)),
        minpango = min(probability),
        maxpango = max(probability),
        menpango = mean(probability),
        max = names(sort(table(lineage), decreasing = TRUE))[1]))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
print("summary info")
```

```
## [1] "summary info"
```

```
print(summs)
```

```
## # A tibble: 82 x 2
##    taxon      maxperc
##    <chr>        <dbl>
##  1 ERR4363387   0.938
##  2 ERR4364007   0.871
##  3 ERR4664555   0.945
##  4 ERR4667618   0.990
##  5 ERR4692364   0.914
##  6 ERR4693034   0.847
##  7 ERR4693061   0.941
##  8 ERR4693079   0.866
##  9 ERR4693537   0.972
## 10 ERR4693605   0.970
## # ... with 72 more rows
```

```
1 - mean(summs$maxperc); 1 - mean(summs$menpango)
```

```
## [1] 0.09124434
```

```
## Warning: Unknown or uninitialised column: `menpango`.
```

```
## Warning in mean.default(summs$menpango): argument is not numeric or logical:
## returning NA
```

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

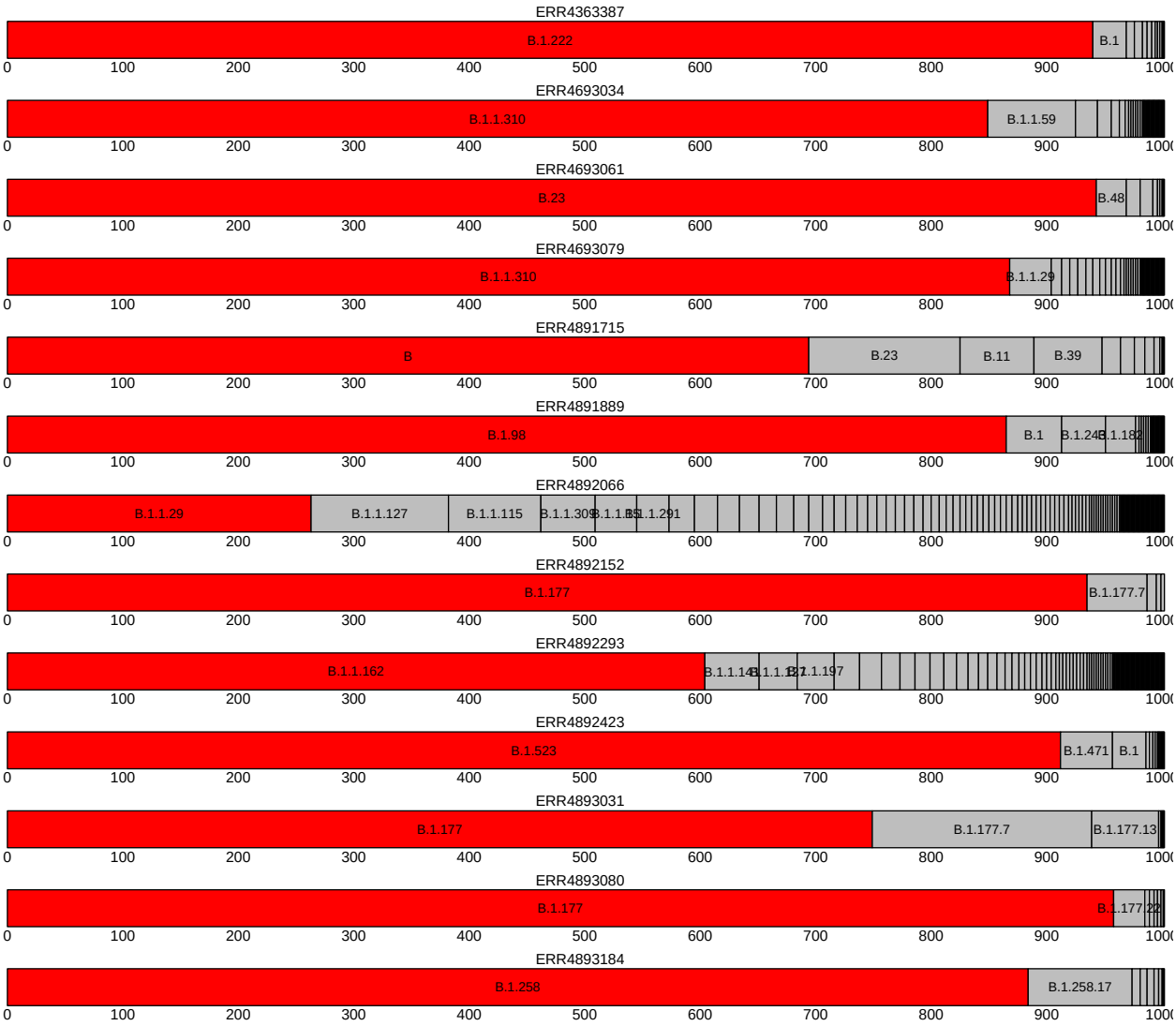
Stacked Bar Plots

```
max_label <- max(table(lins$taxon)) / 40

par(mfrow = c(15, 1), mar = c(1, 0.5, 1.5, 0.5))
for (i in 1:38) {
    pang <- lins[lins$taxon == taxons[i], ]
    called <- pang$lineage[pang$sample == 0][1]
    pangtab <- sort(table(pang$lineage), decreasing = TRUE)
```

```
    colvec <- rep("grey", length(pangtab))
    colvec[which(names(pangtab) == called)] <- "red"

    n <- sum(pangtab > max_label)
    if (n > 1) {
        barlabx <- c(0, cumsum(pangtab[1:(n - 1)])) +
            pangtab[1:n] / 2

        barplot(as.matrix(pangtab),
            col = colvec, hori = TRUE, axes = FALSE)
        text(barlabx, 0.7, names(pangtab)[1:n])
        mtext(side = 3, text = taxons[i], cex = 0.75, las = 1)
        pretty_labels <- seq(0, sum(pangtab),
                by = ifelse(sum(pangtab) < 2000, 100, 1000))
        mtext(side = 1,
            at = pretty_labels,
            text = pretty_labels,
            line = 0,
            cex = 0.75
        )
    } else {
        cat("Taxon", taxons[i], "had", length(pangtab),
            "unique calls, with largest accounting for",
            pangtab[1], "lineage calls, \n\t with second place",
            pangtab[2], ". First place was",
            ifelse(names(pangtab)[1] == called, "", "not"),
            "the conseq call.\n")
    }
}
```

```
## Taxon ERR4364007 had 43 unique calls, with largest accounting for 873 lineage calls,
##   with second place 14 . First place was  the conseq call.
## Taxon ERR4664555 had 18 unique calls, with largest accounting for 947 lineage calls,
##   with second place 15 . First place was  the conseq call.
## Taxon ERR4667618 had 4 unique calls, with largest accounting for 992 lineage calls,
##   with second place 7 . First place was  the conseq call.
## Taxon ERR4692364 had 20 unique calls, with largest accounting for 916 lineage calls,
##   with second place 22 . First place was  the conseq call.
## Taxon ERR4693537 had 6 unique calls, with largest accounting for 974 lineage calls,
##   with second place 23 . First place was  the conseq call.
## Taxon ERR4693605 had 11 unique calls, with largest accounting for 972 lineage calls,
##   with second place 12 . First place was  the conseq call.
## Taxon ERR4758772 had 4 unique calls, with largest accounting for 999 lineage calls,
##   with second place 1 . First place was  the conseq call.
## Taxon ERR4891711 had 15 unique calls, with largest accounting for 966 lineage calls,
##   with second place 7 . First place was  the conseq call.
## Taxon ERR4891805 had 7 unique calls, with largest accounting for 983 lineage calls,
##   with second place 7 . First place was  the conseq call.
## Taxon ERR4891841 had 97 unique calls, with largest accounting for 732 lineage calls,
##   with second place 19 . First place was  the conseq call.
## Taxon ERR4891863 had 58 unique calls, with largest accounting for 838 lineage calls,
##   with second place 18 . First place was  the conseq call.
```

```
## Taxon ERR4891898 had 16 unique calls, with largest accounting for 969 lineage calls,
##   with second place 4 . First place was  the conseq call.
## Taxon ERR4891916 had 1 unique calls, with largest accounting for 1002 lineage calls,
##   with second place NA . First place was  the conseq call.
## Taxon ERR4891988 had 35 unique calls, with largest accounting for 873 lineage calls,
##   with second place 25 . First place was  the conseq call.
## Taxon ERR4892048 had 8 unique calls, with largest accounting for 990 lineage calls,
##   with second place 5 . First place was  the conseq call.

## Taxon ERR4892112 had 13 unique calls, with largest accounting for 975 lineage calls,
##   with second place 14 . First place was  the conseq call.

## Taxon ERR4892200 had 2 unique calls, with largest accounting for 993 lineage calls,
##   with second place 9 . First place was  the conseq call.
## Taxon ERR4892203 had 32 unique calls, with largest accounting for 905 lineage calls,
##   with second place 12 . First place was  the conseq call.

## Taxon ERR4892339 had 8 unique calls, with largest accounting for 962 lineage calls,
##   with second place 17 . First place was  the conseq call.
## Taxon ERR4892386 had 5 unique calls, with largest accounting for 997 lineage calls,
##   with second place 2 . First place was  the conseq call.
## Taxon ERR4892392 had 10 unique calls, with largest accounting for 968 lineage calls,
##   with second place 11 . First place was  the conseq call.

## Taxon ERR4893013 had 18 unique calls, with largest accounting for 901 lineage calls,
##   with second place 21 . First place was  the conseq call.

## Taxon ERR4893033 had 8 unique calls, with largest accounting for 988 lineage calls,
##   with second place 5 . First place was  the conseq call.
## Taxon ERR4893037 had 124 unique calls, with largest accounting for 543 lineage calls,
##   with second place 17 . First place was  the conseq call.

## Taxon ERR4893138 had 3 unique calls, with largest accounting for 993 lineage calls,
##   with second place 8 . First place was  the conseq call.
```


ERR4363387
B.1.222
B.1
ERR4693034
B.1.1.310
B.1.1.59
ERR4693061
B.23
B.48
ERR4693079
B.1.1.310
B.1.1.29
ERR4891715
B
B.23
B.11
B.39
ERR4891889
B.1.98
B.1
B.1.24
B.1.182
ERR4892066
B.1.1.29
B.1.1.127
B.1.1.115
B.1.1.291
ERR4892152
B.1.177
B.1.177.7
ERR4892293
B.1.1.162
B.1.1.197
ERR4892423
B.1.523
B.1.471
B.1
ERR4893031
B.1.177
B.1.177.7
B.1.177.13
ERR4893080
B.1.177
ERR4893184
B.1.258
B.1.258.17
0
100
200
300
400
500
600
700
800
900