Results of PangoVis

Devan Becker

2021-04-15

Load Packages and Data

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

dirich <- 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 < 4500)])
cat(length(badlins), " runs were removed for having too few samples.")
```

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

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

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

```
## [1] 138
```

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: 138 x 7
##    taxon      maxperc uniques minpango maxpango menpango max
##    <chr>        <dbl>   <int>    <dbl>    <dbl>    <dbl> <chr>
##  1 ERR4305816   0.958      15        1        1        1 B.3.1
##  2 ERR4307842   0.950      29        1        1        1 B.1.1.289
##  3 ERR4363387   0.953      27        1        1        1 B.1.222
##  4 ERR4364007   0.855      89        1        1        1 B.1.1.29
##  5 ERR4440194   0.515      18        1        1        1 B.1
##  6 ERR4440219   0.937      70        1        1        1 B.1.1.164
##  7 ERR4440247   0.631     227        1        1        1 B.1.1.29
##  8 ERR4440332   0.984       8        1        1        1 B.40
##  9 ERR4440354   0.665     211        1        1        1 B.1.1.29
## 10 ERR4440373   0.995       9        1        1        1 B.40
## # ... with 128 more rows
```

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

```
## [1] 0.1098993
```

```
## [1] 0.03622898
```

Stacked Bar Plots

```
max_label <- 250
other_label <- 100

par(mfrow = c(17, 1), mar = c(0.05, 7.75, 0.05, 0.05))
if (exists("seq_info")) rm(seq_info)
for (i in seq_along(taxons)) {
    pang <- lins[lins$taxon == taxons[i], ]
    called <- pang$lineage[pang$sample == 0][1]
    pangtab <- sort(table(pang$lineage), decreasing = TRUE)

    # Prep the data for a nicely formatted table
    # Subtract one because of the conseq.
    seq_info_i <- data.frame(
        called = called,
        mode = names(pangtab)[1],
            mode_n = pangtab[1] - 1,
            perc = round(100 * (pangtab[1] - 1) / (sum(pangtab) - 1), 2),
        runner_up = names(pangtab)[2],
```

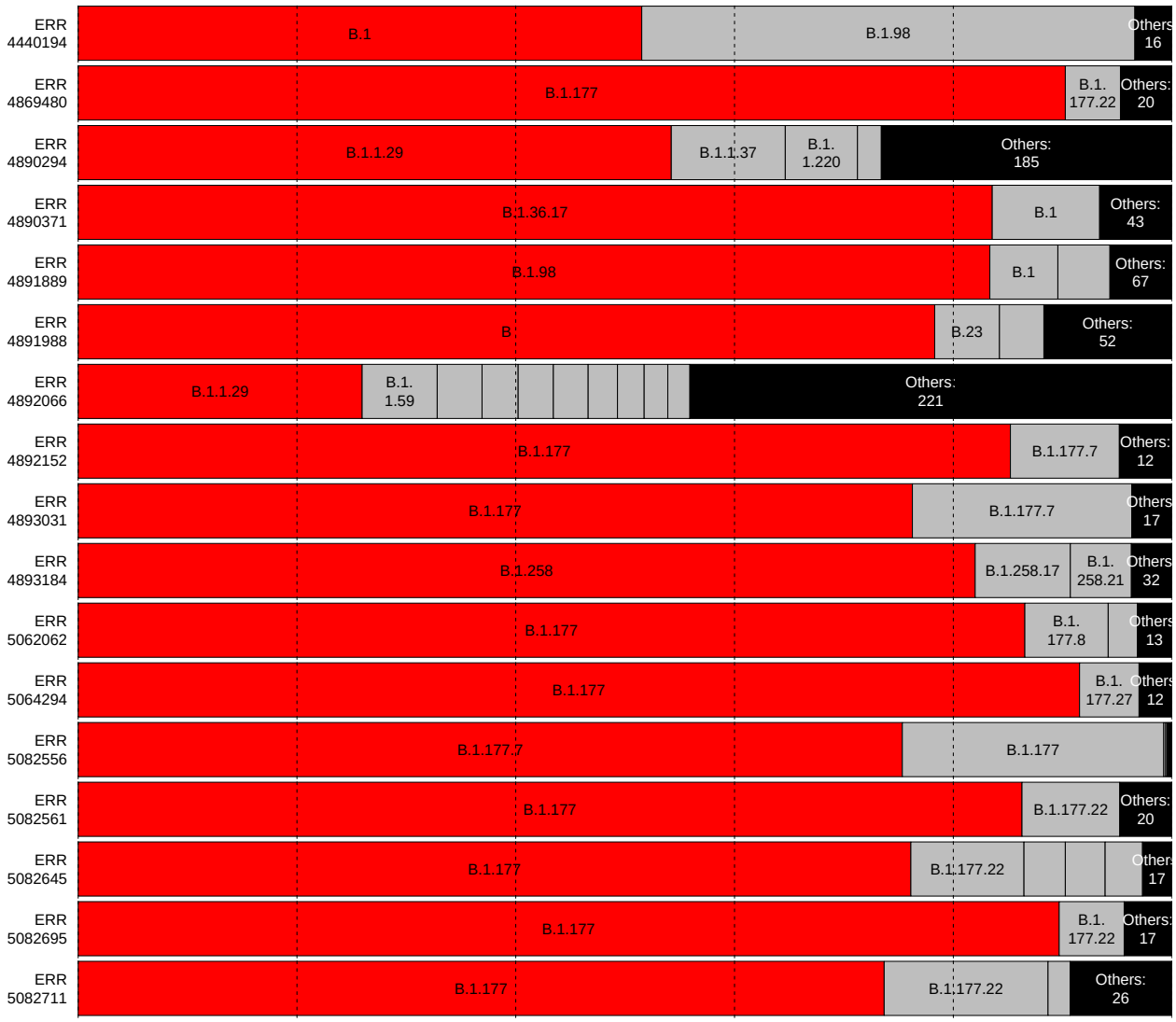
```
            ru_n = pangtab[2],
        unique = length(pangtab), atoms = sum(pangtab == 1))
    seq_info_i$taxon <- taxons[i]

    if (!exists("seq_info")) {
        seq_info <- seq_info_i
    } else {
        seq_info <- bind_rows(seq_info, seq_info_i)
    }

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

    n <- sum(pangtab > max_label)
    if (n > 1) {
        add_other <- FALSE
        if (sum(pangtab < other_label) > 10) {
            add_other <- TRUE
            other_count <- sum(pangtab <= other_label)
            pangtab <- c(pangtab[pangtab > other_label],
                c("other" = sum(pangtab[pangtab <= other_label])))
            colvec[which(names(pangtab) == "other")] <- "black"
        }
        barlabx <- c(0, cumsum(pangtab[1:(n - 1)])) +
            pangtab[1:n] / 2
        barlabels <- names(pangtab)[1:n]
        barlens <- sapply(gregexpr("\\.", barlabels), length)
        for (j in seq_along(barlabels)) {
            if (pangtab[j] < 400 & barlens[j] >= 2) {
                barsplit <- strsplit(barlabels[j], split = "\\.")[[1]]
                barn <- length(barsplit)
                half <- floor(barn / 2)
                barlabels[j] <- paste0(
                    paste(barsplit[1:half], collapse = "."),
                    ".\n",
                    paste(barsplit[(half + 1):barn], collapse = ".")
                )
            }
        }

        barplot(as.matrix(pangtab),
            col = colvec, hori = TRUE, axes = FALSE)
        text(barlabx, 0.7, barlabels, cex = 1.5)
        if (add_other) {
            text(x = sum(pangtab) - pangtab["other"] / 2,
            y = 0.7, col = "white", cex = 1.5,
            label = paste0("Others:\n", other_count))
        }
        mtext(side = 2, cex = 1, las = 1,
            text = paste(substr(taxons[i], 1, 3),
                substr(taxons[i], 4, 20), sep = "\n"))
        abline(v = seq(0, 10000, 1000), lty = 2)
        "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
        )"
    }
}
```




```
seq_info$taxon <- taxons
seq_info <- arrange(seq_info, mode, mode_n) %>%
    select(taxon, everything())
knitr::kable(seq_info, row.names = FALSE)
```

taxon	called	mode	mode_n	perc	runner_up	ru_n	unique	atoms
SRR12762573	A	A	3532	70.68	B	1028	17	4
SRR13092002	A.1	A.1	4922	98.50	B.40	37	17	7
SRR13020990	A.2.2	A.2.2	2892	57.87	A.2	737	107	48
ERR4891988	B	B	3914	78.33	B.23	296	55	24
ERR4999282	B	B	4041	80.87	B.54	246	53	11
ERR4891715	B	B	4521	90.47	B.23	123	28	6
SRR13020989	B.1.1.273	B.1	1338	26.87	B	302	425	132
ERR4440194	B.1	B.1	2575	51.53	B.1.98	2254	18	4
ERR4891841	B.1	B.1	3785	75.75	B.1.243	38	219	52
ERR4890354	B.1	B.1	4326	86.57	B.1.243	25	179	48
ERR4893013	B.1	B.1	4444	88.93	B.1.88	124	49	17
ERR4692364	B.1	B.1	4524	90.53	B.1.400	170	58	25
ERR4440731	B.1	B.1	4666	93.38	B.1.391	50	54	19
ERR5069624	B.1	B.1	4821	96.48	B.1.247	83	42	18
ERR4694556	B.1.1.15	B.1.1.15	4772	95.50	B.1.1.107	19	57	17
ERR4892293	B.1.1.162	B.1.1.162	3145	62.94	B.1.1.197	93	190	48
ERR4440219	B.1.1.164	B.1.1.164	4682	93.70	B.1.1	32	70	23
ERR4891863	B.1.1.216	B.1.1.216	4201	84.07	B.1	119	127	48
ERR4893186	B.1.1.216	B.1.1.216	4432	88.69	B.1	44	86	24
ERR4892203	B.1.1.216	B.1.1.216	4495	89.95	B.1.1.208	64	90	24
ERR5080893	B.1.1.251	B.1.1.251	2943	58.90	B.1	185	149	27
ERR4664555	B.1.1.253	B.1.1.253	4813	96.32	B.1	49	23	4

taxon	called	mode	mode_n	perc	runner_up	ru_n	unique	atoms
ERR4307842	B.1.1.289	B.1.1.289	4745	94.96	B.1	56	29	11
ERR4759453	B.1.1.29	B.1.1.29	977	19.55	B.1.1.208	244	207	47
ERR4440425	B.1.1.29	B.1.1.29	1076	21.53	B.1	129	260	24
ERR4892066	B.1.1.29	B.1.1.29	1297	25.96	B.1.1.59	343	231	56
ERR4890228	B.1.1.29	B.1.1.29	1982	39.66	B.1.1.273	133	232	58
ERR4893037	B.1.1.29	B.1.1.29	2495	49.93	B.1	90	223	28
ERR4890294	B.1.1.29	B.1.1.29	2710	54.23	B.1.1.37	521	189	41
ERR4440247	B.1.1.29	B.1.1.29	3155	63.14	B.1	229	227	38
ERR4694571	B.1.1.29	B.1.1.29	3176	63.56	B.1.1.44	131	213	42
ERR4440354	B.1.1.29	B.1.1.29	3323	66.50	B.1.1.132	65	211	33
ERR4440402	B.1.1.29	B.1.1.29	3490	69.84	B.1	168	215	45
ERR4364007	B.1.1.29	B.1.1.29	4271	85.47	B.1.1	56	89	29
ERR4694617	B.1.1.304	B.1.1.304	4711	94.28	B.1.1.10	66	28	3
ERR4891898	B.1.1.304	B.1.1.304	4831	96.68	B.1.1.4	18	29	7
ERR4893033	B.1.1.307	B.1.1.307	4870	97.46	B.1	63	34	22
ERR5062514	B.1.1.307	B.1.1.307	4943	98.92	B.1.1.311	18	17	10
ERR4892048	B.1.1.307	B.1.1.307	4966	99.38	B.1.1.311	16	8	5
ERR4893353	B.1.1.307	B.1.1.307	4973	99.52	B.1	13	7	3
ERR4890271	B.1.1.307	B.1.1.307	4978	99.62	B.1	7	8	3
ERR4693079	B.1.1.310	B.1.1.310	4376	87.57	B.1.1.29	231	107	50
ERR4693034	B.1.1.310	B.1.1.310	4399	88.03	B.1.1.59	57	89	44
ERR5080913	B.1.1.311	B.1.1.311	4861	97.28	B.1.1.281	47	16	8
ERR5082696	B.1.1.315	B.1.1.315	4575	91.55	B.1	147	43	20
ERR5082664	B.1.1.315	B.1.1.315	4866	97.38	B.1	60	13	4
ERR4667618	B.1.1.315	B.1.1.315	4933	98.72	B.1	30	11	3
ERR4869497	B.1.1.315	B.1.1.315	4944	98.94	B.1	15	11	4
ERR5062388	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5062935	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5063143	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5063807	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5064346	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5064811	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5069584	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5069616	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5069871	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5070294	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5077411	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5077618	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5082610	B.1.1.7	B.1.1.7	4997	100.00	NA	NA	1	0
ERR5074314	B.1.160	B.1.160	4833	96.72	B.1.160.8	95	18	10
ERR5082569	B.1.160	B.1.160	4845	96.96	B.1.160.5	68	23	10
ERR4869446	B.1.160.7	B.1.160.7	4947	99.00	B.1.160	42	5	0
ERR5082711	B.1.177	B.1.177	3683	73.70	B.1.177.22	749	29	11
ERR5082645	B.1.177	B.1.177	3805	76.15	B.1.177.22	517	22	7
ERR4893031	B.1.177	B.1.177	3812	76.29	B.1.177.7	1003	19	7
ERR4892152	B.1.177	B.1.177	4260	85.25	B.1.177.7	498	14	3
ERR5082561	B.1.177	B.1.177	4313	86.31	B.1.177.22	447	22	5
ERR5062062	B.1.177	B.1.177	4326	86.57	B.1.177.8	381	16	2
ERR5082695	B.1.177	B.1.177	4483	89.71	B.1.177.22	297	19	4
ERR4869480	B.1.177	B.1.177	4511	90.27	B.1.177.22	252	22	5
ERR4893242	B.1.177	B.1.177	4561	91.27	B.1.177.6	164	15	3
ERR4892339	B.1.177	B.1.177	4566	91.37	B.1.177.23	131	19	1
ERR5082576	B.1.177	B.1.177	4567	91.39	B.1.177.22	237	20	1
ERR5064294	B.1.177	B.1.177	4576	91.57	B.1.177.27	272	14	2
ERR5082580	B.1.177	B.1.177	4601	92.08	B.1.177.22	242	20	1
ERR5081301	B.1.177	B.1.177	4638	92.82	B.1.177.22	195	21	4
ERR4869458	B.1.177	B.1.177	4648	93.02	B.1.177.22	198	19	3
ERR4869487	B.1.177	B.1.177	4702	94.10	B.1.177.22	147	17	1
ERR5080918	B.1.177	B.1.177	4723	94.52	B.1.177.22	150	19	4
ERR5064166	B.1.177	B.1.177	4731	94.68	B.1.177.22	149	16	1
ERR5062729	B.1.177	B.1.177	4760	95.26	B.1.177.22	116	14	1
ERR4893080	B.1.177	B.1.177	4767	95.40	B.1.177.22	118	14	5
ERR5063922	B.1.177	B.1.177	4769	95.44	B.1.177.22	148	16	1
ERR4892392	B.1.177	B.1.177	4770	95.46	B.1.177.22	95	18	5
ERR5063539	B.1.177	B.1.177	4774	95.54	B.1.177.22	121	20	4
ERR4893197	B.1.177	B.1.177	4779	95.64	B.1.177.23	105	18	5
ERR5077151	B.1.177	B.1.177	4785	95.76	B.1.177.22	114	20	2
ERR5062571	B.1.177	B.1.177	4800	96.06	B.1.177.22	98	21	3
ERR5060778	B.1.177	B.1.177	4827	96.60	B.1.177.22	115	15	5
ERR4890403	B.1.177	B.1.177	4831	96.68	B.1.177.22	82	18	5
ERR5070060	B.1.177	B.1.177	4854	97.14	B.1.177.22	69	14	2
ERR5081316	B.1.177.15	B.1.177.15	4799	96.04	B.1.177	131	16	6
ERR4893138	B.1.177.16	B.1.177.16	4954	99.14	B.1.177	30	7	3
ERR4892200	B.1.177.17	B.1.177.17	4956	99.18	B.1.177	39	4	2
ERR5064787	B.1.177.17	B.1.177.17	4958	99.22	B.1.177	23	5	2
ERR4890386	B.1.177.18	B.1.177.18	4969	99.44	B.1.177.13	15	6	2
ERR5076163	B.1.177.19	B.1.177.19	4856	97.18	B.1.177	99	11	1
ERR5076748	B.1.177.19	B.1.177.19	4892	97.90	B.1.177	67	12	5
ERR5063165	B.1.177.19	B.1.177.19	4905	98.16	B.1.177	67	8	2

taxon	called	mode	mode_n	perc	runner_up	ru_n	unique	atoms
ERR5063813	B.1.177.19	B.1.177.19	4932	98.70	B.1.177	47	9	3
ERR4693605	B.1.177.3	B.1.177.3	4825	96.56	B.1.177	68	17	2
ERR5081304	B.1.177.4	B.1.177.4	4954	99.14	B.1.177.2	21	17	10
ERR5082590	B.1.177.4	B.1.177.4	4971	99.48	B.1.177.2	13	11	7
ERR5062648	B.1.177.4	B.1.177.4	4988	99.82	B.1.177	4	6	3
ERR5082674	B.1.177.6	B.1.177.6	4858	97.22	B.1.177	79	13	4
ERR4891805	B.1.177.6	B.1.177.6	4939	98.84	B.1.177	24	8	0
ERR5082712	B.1.177.7	B.1.177.7	3334	66.72	B.1.177	1612	15	5
ERR5082556	B.1.177.7	B.1.177.7	3766	75.37	B.1.177	1195	11	2
ERR5082630	B.1.177.7	B.1.177.7	4833	96.72	B.1.177	128	13	6
ERR4693537	B.1.177.7	B.1.177.7	4849	97.04	B.1.177	126	13	6
ERR4363387	B.1.222	B.1.222	4761	95.28	B.1	96	27	14
ERR5081322	B.1.235	B.1.235	4743	94.92	B.1	128	21	8
ERR4893184	B.1.258	B.1.258	4099	82.03	B.1.258.17	435	35	15
ERR5062004	B.1.258	B.1.258	4563	91.31	B.1	80	35	7
ERR4891711	B.1.36	B.1.36	4770	95.46	B.1.36.9	81	27	8
ERR4890371	B.1.36.17	B.1.36.17	4177	83.59	B.1	491	45	17
ERR5080897	B.1.36.17	B.1.36.17	4763	95.32	B.1	132	24	13
ERR4890352	B.1.36.17	B.1.36.17	4938	98.82	B.1	20	16	10
ERR4890337	B.1.36.17	B.1.36.17	4951	99.08	B.1	20	9	3
ERR4892423	B.1.523	B.1.523	4493	89.91	B.1	128	31	9
ERR4893393	B.1.523	B.1.523	4658	93.22	B.1	153	36	15
ERR4890285	B.1.523	B.1.523	4892	97.90	B.1	50	23	9
ERR4891889	B.1.98	B.1.98	4166	83.37	B.1	311	70	42
ERR4693061	B.23	B.23	4734	94.74	B.48	113	19	5
ERR4694400	B.3	B.3	4903	98.12	B.3.1	42	13	3
ERR4305816	B.3.1	B.3.1	4788	95.82	B.3	147	15	3
ERR4892112	B.39	B.39	4733	94.72	B.3	85	31	13
ERR4890427	B.39	B.39	4757	95.20	A.16	96	20	4
ERR4440332	B.40	B.40	4917	98.40	B.1	41	8	2
ERR4892386	B.40	B.40	4967	99.40	B.1	10	8	2
ERR4440373	B.40	B.40	4972	99.50	B	10	9	5
ERR4891916	None	None	4993	99.96	B.1	2	2	0
ERR4999251	None	None	4997	100.00	NA	NA	1	0
ERR4999255	None	None	4997	100.00	NA	NA	1	0
ERR4999275	None	None	4997	100.00	NA	NA	1	0
SRR12639958	None	None	4997	100.00	NA	NA	1	0