COVID Phylogenies

using phangorn

Kim

12/11/2020

Set Up

Read in aligned data and grab some sequence info

```
states<-read.aa("data/aligned_states_china_muscle.txt", format="fasta")
coding<-read.dna("data/aln-fasta.txt",format="fasta")
```

Add in additional data

```
state_data<-read.delim("data/states_data.txt", header=FALSE)
china_data<-read.delim("data/china_data.txt", header=FALSE)
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## incomplete final line found by readTableHeader on 'data/china_data.txt'
```

```
label_data=rbind(state_data,china_data)
names(label_data)<-c("accession","state","date")
```

Update labels to just be the accession number for easier graphing later on

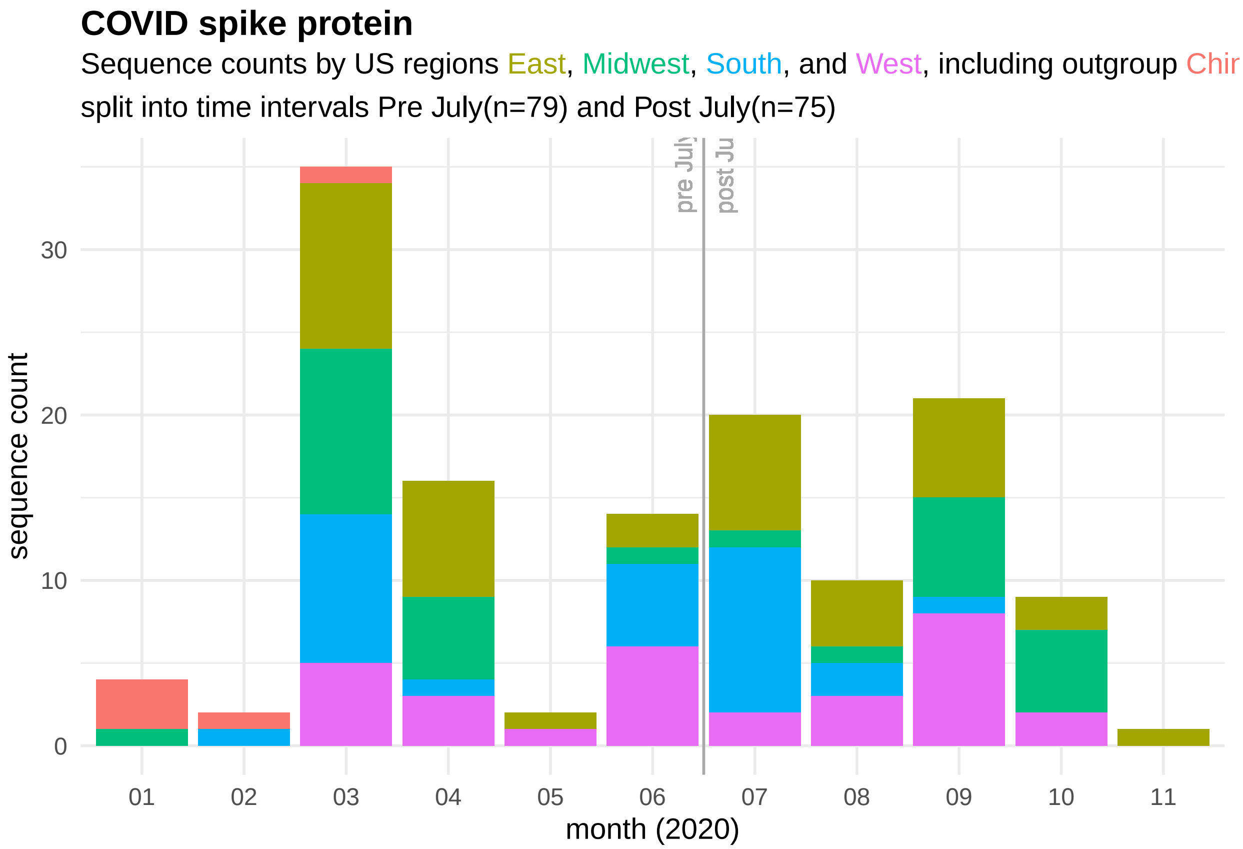
```
old_labels=as.list(names(states))
states=updateLabel(states, old_labels, as.character(label_data$accession))
```

EDA

Cleaning up data for use in plotting

```
timeline<-label_data%>%
  na.omit()%>%
  filter(state!="GA"&state!="IN")%>% #take out incorrectly queried data
  mutate(dated= zoo::as.yearmon(date, "%Y-%m"))%>%
  mutate(dated=format(dated, "%m"))%>%
  mutate(period=ifelse(as.numeric(dated)<7,"pre July","post July"))%>%
  mutate(region=case_when(
    state %in% c("CA","NM","WA") ~ "West",
    state %in% c("FL","GA","TX","SC") ~ "South",
    state %in% c("NY","MD","VA","NC") ~ "East",
    state %in% c("MN","WI","IL","IN") ~ "Midwest",
    TRUE~ "China"
  ))%>%
  mutate(state=as.character(state))%>%
  mutate(state_date=paste(as.character(state),as.numeric(dated)))%>%
  na.omit()
```

distribution of sequence counts over time



Methods

Change alignment to phyDat object for use in the phanghon package.

```
states_phyDat <- phyDat(states, type = "AA", levels = NULL)
coding_phyDat<-phyDat(coding, type="DNA",levels=NULL)
```

Run some model testing to see which distance matrix is best for our data. We can use mt list to pick lowest AIC. Note: the modelTest function takes a while.

```
#mt takes 4everrrr!! change cores to 2 for faster implem in parallel
mt <- modelTest(states_phyDat, model="all")
```

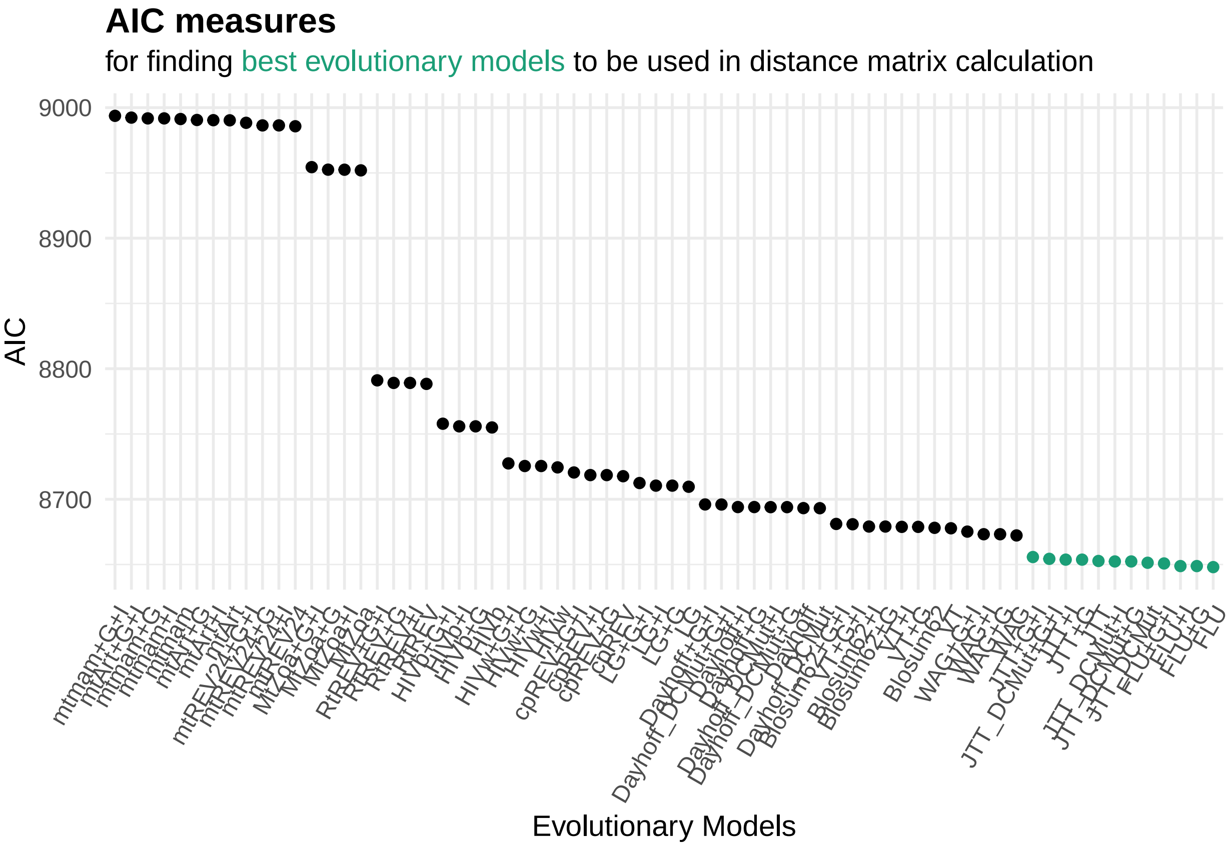
```
## negative edges length changed to 0!
```

```
## [1] "WAG+I"
## [1] "WAG+G"
## [1] "WAG+G+I"
## [1] "JTT+I"
## [1] "JTT+G"
## [1] "JTT+G+I"
## [1] "LG+I"
## [1] "LG+G"
## [1] "LG+G+I"
## [1] "Dayhoff+I"
## [1] "Dayhoff+G"
## [1] "Dayhoff+G+I"
## [1] "cpREV+I"
## [1] "cpREV+G"
## [1] "cpREV+G+I"
## [1] "mtmam+I"
## [1] "mtmam+G"
## [1] "mtmam+G+I"
## [1] "mtArt+I"
## [1] "mtArt+G"
## [1] "mtArt+G+I"
## [1] "MtZoa+I"
## [1] "MtZoa+G"
## [1] "MtZoa+G+I"
## [1] "mtREV24+I"
## [1] "mtREV24+G"
## [1] "mtREV24+G+I"
## [1] "VT+I"
## [1] "VT+G"
## [1] "VT+G+I"
## [1] "RtREV+I"
## [1] "RtREV+G"
## [1] "RtREV+G+I"
## [1] "HIVw+I"
## [1] "HIVw+G"
## [1] "HIVw+G+I"
## [1] "HIVb+I"
## [1] "HIVb+G"
## [1] "HIVb+G+I"
## [1] "FLU+I"
## [1] "FLU+G"
## [1] "FLU+G+I"
## [1] "Blosum62+I"
## [1] "Blosum62+G"
## [1] "Blosum62+G+I"
## [1] "Dayhoff_DCMut+I"
## [1] "Dayhoff_DCMut+G"
## [1] "Dayhoff_DCMut+G+I"
## [1] "JTT_DCMut+I"
## [1] "JTT_DCMut+G"
## [1] "JTT_DCMut+G+I"
```

```
states_dist <- dist.ml(states, model="FLU")

coding_dist<-dist.dna(coding,model = "JC")
```

plotting AIC



Tree construction for NJ, UPGMA, fastME

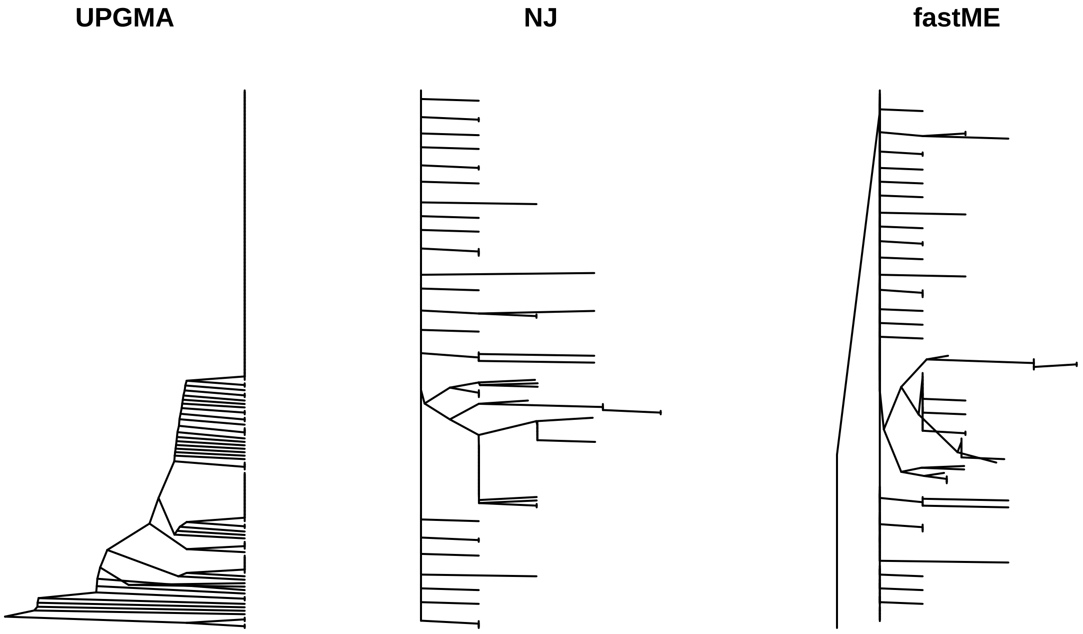
```
states_UPGMA <- upgma(states_dist)
states_NJ   <- NJ(states_dist)
states_fastme<-fastme.bal(states_dist)

coding_UPGMA <- upgma(coding_dist)
coding_NJ   <- NJ(coding_dist)
coding_fastme<-fastme.bal(coding_dist)
```

#Plots

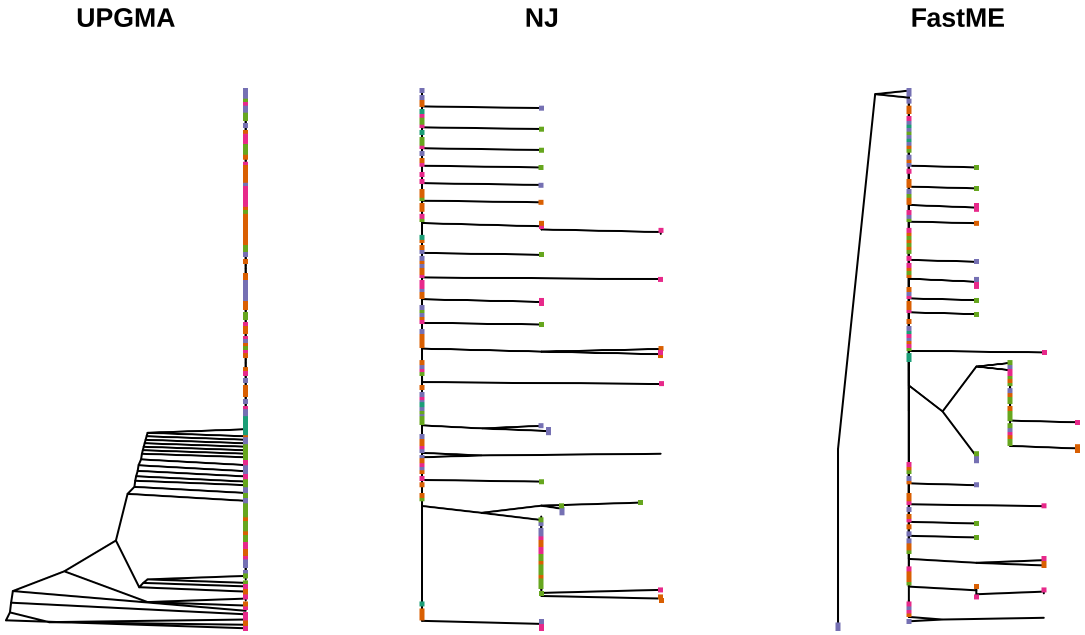
Basic Tree structure

```
par(mfrow=c(1,3),oma = c(0, 0, 2, 0))
plot(coding_UPGMA,
     main = "UPGMA"
     ,type="cladogram",
     show.tip=FALSE)
tiplabels(pch=15,cex=.5,
          col = as.factor(timeline$region[match(coding_UPGMA$tip.label, timeline$accession)])
          )
plot(coding_NJ,
     main = "NJ"
     ,type="cladogram",
     show.tip=FALSE)
tiplabels(pch=15,cex=.5,
          col = as.factor(timeline$region[match(coding_UPGMA$tip.label, timeline$accession)])
          )
plot(coding_fastme,
     main = "fastME"
     ,type="cladogram",
     show.tip=FALSE)
tiplabels(pch=15,cex=.5,
          col = as.factor(timeline$region[match(coding_fastme$tip.label, timeline$accession)])
          )
```

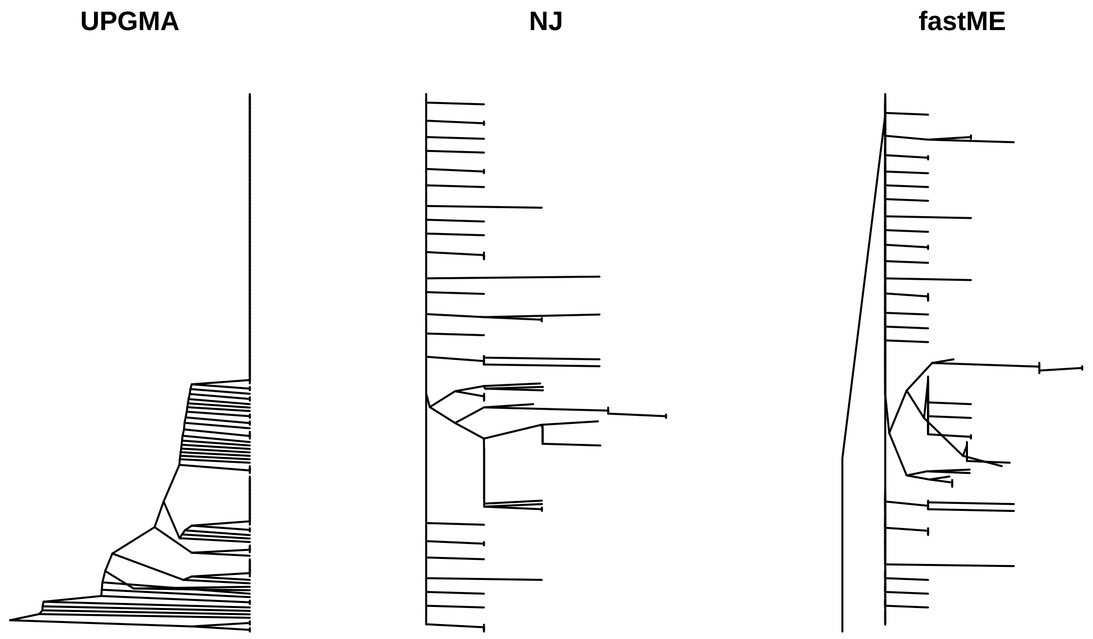


Region plots amino acids

Model selection based off of US region East Midwest South and West with outgroup China

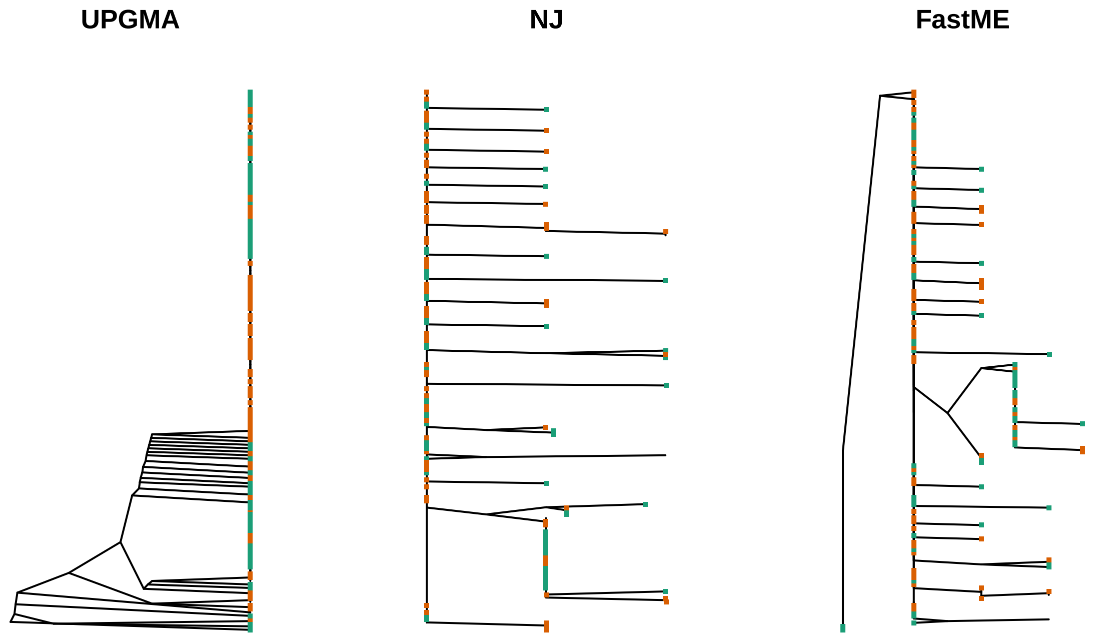


```
par(mfrow=c(1,3),oma = c(0, 0, 2, 0))
plot(coding_UPGMA,
     main = "UPGMA"
     ,type="cladogram",
     show.tip=FALSE)
tiplabels(pch=15,cex=.5,
          col = as.factor(timeline$region[match(coding_UPGMA$tip.label, timeline$accession)])
          )
plot(coding_NJ,
     main = "NJ"
     ,type="cladogram",
     show.tip=FALSE)
tiplabels(pch=15,cex=.5,
          col = as.factor(timeline$region[match(coding_UPGMA$tip.label, timeline$accession)])
          )
plot(coding_fastme,
     main = "fastME"
     ,type="cladogram",
     show.tip=FALSE)
tiplabels(pch=15,cex=.5,
          col = as.factor(timeline$region[match(coding_fastme$tip.label, timeline$accession)])
          )
```



Time period plots

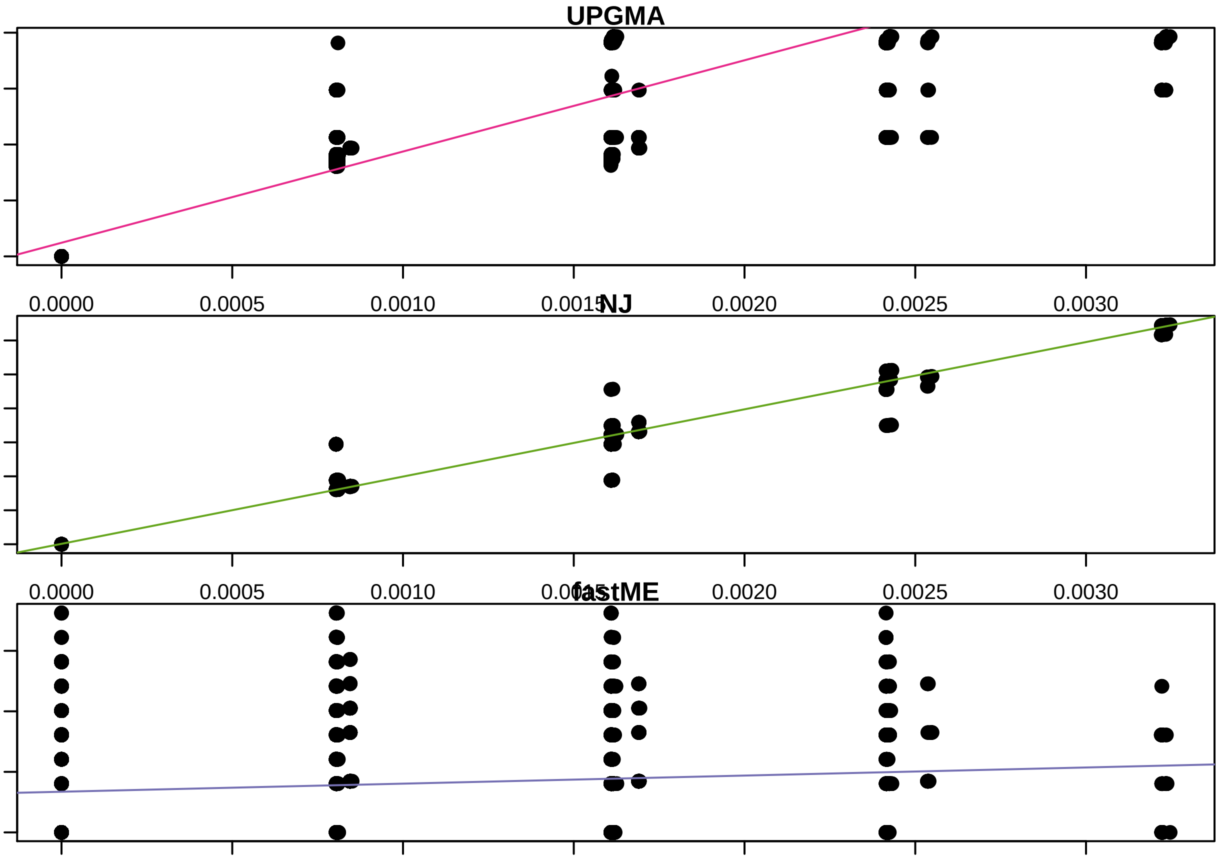
Model selection based on time intervals Pre July and Post July



correlation plots

```
## [1] 0.8157761
```

```
## [1] 0.9905525
```



```
## [1] 0.9905525
```

Junk Drawer

Plotting states data

```
library(usmap)
monthly<-timeline%>%group_by(state,date)%>%count()
periodly<-timeline%>%group_by(state,period)%>%count()
```

pick whether monthly or periodly

```
p<-plot_usmap(data=periodly,values="n",
           include = c("CA","NM","WA","FL","GA","TX",
                       "SC","NY","MD","VA","NC","MN",
                       "WI","IL","IN"),
           color="purple") +
  scale_fill_continuous(
    low = "white", high = "purple", name = "seq count",
    label = scales::comma
  ) + theme(legend.position = "right")+
  labs(title="Sequence counts per month in 2020")+
  facet_wrap(~period)
```