Code for ‘Daily longitudinal sampling of early SARS-CoV-2 infection reveals substantial heterogeneity in infectiousness and effects of the B.1.1.7 (Alpha) variant’

Contents

R packages

```
require(magick)
require(ggpubr)
require(cowplot)
require(MASS)
require(PupillometryR)
require(fitdistrplus)
require(tidyverse)
require(dplyr)
require(deSolve)
```

Data individuals

```
#### Neg CN values are recorded as 48, Ct as 47, and Virus_pos_days as 5.9 ####
dat1 <- read_csv("../Data/data_samples.csv") %>%
  mutate(Ind=as_factor(Ind))

ID_fig1 <- dat1 %>% select(Ind) %>%
  distinct() %>% slice_head(n=30)

ID_figS1 <- dat1 %>% select(Ind) %>%
  distinct() %>% slice_tail(n=30)

IDs <- dat1 %>% mutate(id=parse_number(as.character(Ind))) %>%
  select(Ind, id) %>% distinct()

minpos=function(x) min(x[x > 0])
maxneg=function(x) max(x[x < 0])
```

Figure 1.

1A. Tests over time

```
figure1A <- dat1 %>% filter(Index == 1) %>% right_join(ID_fig1) %>%
  ggplot() + geom_rect(data=. %>% filter(!is.na(Antigen)), aes(xmin=Time-0.5,
             xmax=Time+0.5, ymin=-Inf, ymax=Inf, alpha=Antigen),
             fill="#DDCC77", stat="unique") +
  geom_point(aes(x=Time, y=Nasal_CN, color="#004488", shape="N"),
             alpha=0.7, size=0.5) +
  geom_point(aes(x=Time, y=Saliva_Ct, col="#009988", shape="S"),
             alpha=0.7, size=0.5) + theme_minimal() +
  geom_point(aes(Time, (Virus_pos_days - 1)*10, color="#B2182B", shape="V"),
             size=0.5, stat="unique", stroke=0.8, alpha=0.7) +
  geom_hline(yintercept=45, lty="dashed", size=0.3) +
  geom_vline(xintercept=0, linetype=3, size=0.3, alpha=0.7) +
  scale_x_continuous("Days since lowest nasal CN value", breaks=seq(-8, 12, 4))+
  scale_alpha_manual("", breaks=c("Neg", "Pos"), values=c(0.2, 0.6),
              labels=c("Negative Antigen test", "Positive Antigen test"),
              guide=guide_legend(override.aes=list(alpha=c(0.4, 1)))) +
  scale_shape_manual("", guide="legend", breaks=c("N","S","V"), values=c(19,19,4),
              labels=c("Nasal PCR","Saliva PCR", "Viral culture")) +
  scale_color_identity("", labels=c("Nasal PCR", "Saliva PCR", "Viral culture"),
              guide=guide_legend(override.aes=list(size=1))) +
  scale_y_continuous("Saliva Ct values\nNasal CN values", trans="reverse",
              breaks=seq(10, 40, 10), limits=c(50, 10),
              sec.axis =sec_axis(~.*0.1+1, name="Day of culture positivity",
                                 breaks=5:2)) +
  facet_wrap(~Ind, scales="free", nrow=5) +
  theme(axis.line=element_line(size=0.3),
        axis.ticks=element_line(size=0.3),
        axis.text.y=element_text(size=8, colour="black"),
        axis.text.x=element_text(size=7, colour="black"),
        axis.title=element_text(size=9, colour="black"),
        axis.line.y.right=element_line(color="#B2182B"),
        axis.text.y.right=element_text(color="#B2182B"),
        axis.ticks.y.right=element_line(color="#B2182B"),
        axis.title.y.right=element_text(color="#B2182B"),
        strip.text=element_text(size=8),
        legend.key.height=unit(0.3, "cm"),
        legend.key.width=unit(0.3, "cm"),
        legend.title=element_blank(),
        legend.text=element_text(size=9),
        legend.background=element_blank(),
        legend.position="bottom",
        panel.grid=element_blank())
```

1B. Ct/CN and viral culture assay result

```
figure1B <- dat1 %>% filter(!is.na(Virus_pos_days)) %>%
  mutate(Virus_pos_days=ifelse(Virus_pos_days > 5, "Negative", Virus_pos_days),
         Virus_pos_days=as_factor(Virus_pos_days),
         Virus_pos_days=fct_relevel(Virus_pos_days, sort)) %>%
  mutate(Saliva_Ct=ifelse(Saliva_Ct >= 40, NA, Saliva_Ct),
         Nasal_CN=ifelse(Nasal_CN >= 42, NA, Nasal_CN)) %>%
  select(Virus_pos_days, Saliva_Ct, Nasal_CN) %>%
  pivot_longer(cols=Saliva_Ct:Nasal_CN, names_to="Type", values_to="Ct") %>%
  ggplot(aes(x=Virus_pos_days, y=Ct, fill=Type))+ theme_minimal() +
  geom_flat_violin(aes(fill=Type, colour=Type), adjust=1.5, trim=T,
                   alpha=0.5, size=0.3, position=position_nudge(x=0.2, y=0)) +
  geom_point(aes(x=Virus_pos_days, y=Ct, colour=Type),
             alpha=0.2, show.legend=F, size=1.5, shape=20,
             position=position_jitterdodge(
               jitter.width=0.1, dodge.width=0.3)) +
  geom_boxplot(aes(x=Virus_pos_days, y=Ct, fill=Type),
               show.legend=F, lwd=0.35,
               outlier.shape=NA, alpha=.5, width=.25, colour="black",
               position=position_dodge(0.3)) +
  scale_color_manual("PCR test", values=c("#004488", "#009988"),
                     labels=c("Nasal", "Saliva"))+
  scale_fill_manual("PCR test", values=c("#004488", "#009988"),
                    labels=c("Nasal", "Saliva")) +
  scale_y_continuous("Saliva Ct values\nNasal CN values", trans="reverse",
                     breaks=seq(10, 40, 5), limits=c(42, 10)) +
  labs(x="Day of viral culture positivity") +
  scale_x_discrete(expand=c(0, 0.6)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        legend.key.height=unit(0.4, "cm"),
        legend.key.width=unit(0.2, "cm"),
        legend.title=element_text(size=9),
        legend.text=element_text(size=8),
        legend.background=element_blank(),
        legend.position="right",
        panel.grid=element_blank(),
        panel.grid.major.y=element_line(size=.05, color="gray"))
```

1C. Antigen tests and viral culture assay

```
data1C <- dat1 %>% select(Virus_pos_days, Antigen) %>%
  filter(!is.na(Virus_pos_days), !is.na(Antigen)) %>%
  mutate(Virus_pos_days=ifelse(Virus_pos_days %in% 2:5, "Positive", "Negative"),
         Antigen=recode_factor(Antigen, "Neg"="Negative", "Pos"="Positive"))

figure1C <- data1C %>% ggplot(aes(x= Virus_pos_days, alpha=Antigen)) +
  theme_minimal() + labs(x="Viral culture", y="Counts") +
  geom_bar(fill="#DDCC77", width=0.75) +
  annotate("text", x=c(1,2), y=c(100,100),
           label=c("28%", "93%"), color="tan4", size=2.5) +
  scale_alpha_manual("Antigen test", values=c(0.4, 1)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        legend.title=element_text(size=9),
        legend.text=element_text(size=8),
        legend.background=element_blank(),
        legend.key.height=unit(0.4, "cm"),
        legend.key.width=unit(0.4, "cm"),
        legend.position="right",
        panel.grid=element_blank())
```

Model fitting RTqPCR data figure 2

Nasal

```
ODEs_twoPopsN <- function (t, y, parms) {
  beta <- parms[2]*1e-8
  delta <- parms[3]
  pi <- parms[4]
  phi <- parms[5]*1e-6
  rho <- parms[6]
  TC <- y[1]
  PC <- y[2]
  EC <- y[3]
  IC <- y[4]
  VL <- y[5]

  dtdTC <- - beta*TC*VL - phi*TC*IC + rho*PC
  dtdPC <- phi*TC*IC -rho*PC
  dtdEC <- beta*TC*VL - kk*EC
  dtdIC <- kk*EC - delta*IC
  dtdVL <- pi*IC - cc*VL

  out <- list(c(dtdTC, dtdPC, dtdEC, dtdIC, dtdVL))
}

runODEN <- function(parms){
  yini <- c(TC=TON, PC=0, EC=1, IC=0, VL=0)
  times <- c(seq(from=-parms[1], to=tEND, by=0.1))
  out <- as.data.frame(ode(y=yini, times=times,
                           func=ODEs_twoPopsN, parms=parms))
  return(out)
}

pars_MonolixN <- read_csv("../Data/params_nasal_VL.csv")
```

```
kk <- 4
cc <- 10
tEND <- 20
TON <- 4e+6*20

parsN <- pars_MonolixN %>% select(id, tt0_mode:rho_mode) %>%
  mutate(R0s=logbeta_mode*1e-8*TON*pi_mode/delta0_mode/10,
         rr=(-(kk+delta0_mode) + sqrt((kk+delta0_mode)^2 +
                                        4*kk*delta0_mode*(R0s-1)))/2)

dataAllN <- tibble(NULL)
for (i in parsN$id){
  prs <- parsN %>% filter(id == i) %>%
    select(tt0_mode:rho_mode) %>% as.matrix()
  out <- runODEN(prs) %>% as_tibble() %>%
    rowid_to_column("count") %>%
    mutate(logVL=pmax(log10(VL), 1e-1)) %>%
    mutate(Sim=(11.35 - logVL)/0.25)
  dataOut <- tibble(id=i, out %>% select(time, Sim))
  dataAllN <- dataAllN %>% bind_rows(dataOut)
}

dataAllN <- dataAllN %>% left_join(IDs) %>%
  mutate(Type="Nasal_Sim")
```

Saliva

```
ODEs_twoPopsS <- function (t, y, parms) {
  beta <- parms[2]*1e-8
  delta <- parms[3]
  pi <- parms[4]
  t1 <- parms[5]
  gamma0 <- parms[6]
  if (t<t1) {gamma <- 0} else {gamma <- gamma0}

  TC <- y[1]
  EC <- y[2]
  IC <- y[3]
  VL <- y[4]

  dtdTC <- - beta*TC*VL
  dtdEC <- beta*TC*VL - kk*EC
  dtdIC <- kk*EC - delta*IC -gamma*IC
  dtdVL <- pi*IC - cc*VL

  out <- list(c(dtdTC, dtdEC, dtdIC, dtdVL))
  return(out)
}

runODES <- function (parms) {
  yini <- c(TC=TOS, EC=1, IC=0, VL=0)
  times <- c(seq(from=-parms[1], to=tEND, by=0.1))
  out <- ode(y=yini, times=times, func=ODEs_twoPopsS, parms=parms)
  return(as.data.frame(out))
}

pars_MonolixS <- read_csv("../Data/params_saliva_VL.csv")
kk <- 4
cc <- 10
tEND <- 20
TOS <- 1.08e+8

parsS <- pars_MonolixS %>% select(id, tt0_mode:gamma0_mode) %>%
  mutate(R0s=logbeta_mode*1e-8*TOS*pi_mode/delta0_mode/10,
         rr=(-(kk+delta0_mode) + sqrt((kk+delta0_mode)^2 +
                                        4*kk*delta0_mode*(R0s-1)))/2)

dataAllS <- tibble(NULL)
for (i in parsS$id){
  prs <- parsS %>% filter(id == i) %>%
    select(tt0_mode:gamma0_mode) %>% as.matrix()
  out <- runODES(prs) %>% as_tibble() %>%
    rowid_to_column("count") %>%
    mutate(logVL=pmax(log10(VL), 1e-1)) %>%
    mutate(Sim=(14.24 - logVL)/0.28, AUC = sum(logVL)*0.1)
  dataOut <- tibble(id=i, out %>% select(time, Sim, AUC))
  dataAllS <- dataAllS %>% bind_rows(dataOut)
}

dataAllS <- dataAllS %>% left_join(IDs) %>%
  mutate(Type="Saliva_Sim")
```

Figure 2.

2B. Model Fit

```
dat2S <- bind_rows(dataAllN, dataAllS) %>%
  pivot_wider(names_from=Type, values_from=Sim) %>%
  select(Ind, Time=time, Nasal_Sim, Saliva_Sim) %>%
  filter(is.finite(Nasal_Sim) | is.finite(Saliva_Sim)) %>%
  mutate(Ind=as_factor(Ind))

datT <- dat1 %>% filter(Index == 1) %>% group_by(Ind) %>%
  summarise(TimeMin = min(Time)) %>% ungroup()

dat2 <- dat1 %>% left_join(datT) %>%
  mutate(Time = Time - TimeMin) %>%
  full_join(dat2S) %>% group_by(Ind) %>%
  arrange(Nasal_Sim, Time) %>% mutate(Timez=Time - Time[1]) %>%
  ungroup() %>% arrange(Ind, Timez) %>% distinct() %>%
  pivot_longer(cols=c(Nasal_CN, Saliva_Ct, Nasal_Sim, Saliva_Sim),
               names_to="Compartment", values_to="Value") %>%
  separate(Compartment, c("Compartment", "Type"), "_") %>%
  mutate(Type=recode(Type, CN="Obs", Ct="Obs")) %>%
  filter(!is.na(Value), !is.na(Ind)) %>% mutate(row=row_number()) %>%
  pivot_wider(names_from=Type, values_from=Value) %>%
  select(Ind, Timez, Compartment, Sim, Obs) %>% distinct() %>%
  mutate(Obs=case_when(Compartment == "Saliva" & Obs == 47 ~ 40,
                       Compartment == "Nasal" & Obs == 48 ~ 42, TRUE ~ Obs))

figure2B <- dat2 %>% right_join(ID_fig1) %>% ggplot() +
  geom_point(aes(x=Timez, y=Obs, alpha="O", col=Compartment),
             size=1, stat="unique") +
  geom_line(data=. %>% filter(!is.na(Sim)), size=0.5,
            aes(x=Timez, y=Sim, alpha="S", col=Compartment)) +
  geom_hline(yintercept=42, linetype=3, size=0.5, col="#004488") +
  geom_hline(yintercept=40, linetype=3, size=0.5, col="#009988") +
  geom_vline(xintercept=0, linetype=3, size=0.2, alpha=0.7) +
  facet_wrap(~Ind, scales="free", nrow=5) + theme_minimal() +
  scale_alpha_manual("" , breaks=c("O", "S"), values=c(0.5, 1),
            labels=c("Observed", "Model fit"),
            guide=guide_legend(override.aes=
            list(linetype=c("blank", "solid"), shape=c(19, NA)))) +
  scale_color_manual("", breaks=c("Nasal","Saliva"),
                     values=c("#004488", "#009988")) +
  scale_x_continuous("\nDays since predicted lowest nasal CN value\n",
                     breaks=seq(-8, 16, 4), limits=c(-8.5, 16)) +
  scale_y_continuous("Saliva Ct values\nNasal CN values", trans="reverse",
                     breaks=seq(10, 40, 10), limits=c(44, 10)) +
  theme(axis.line=element_line(size=0.3),
        axis.ticks=element_line(size=0.3),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=10, colour="black"),
        strip.text=element_text(size=8),
        legend.key.height=unit(0.3, "cm"),
        legend.key.width=unit(0.3, "cm"),
        legend.title=element_blank(),
        legend.text=element_text(size=10),
        legend.background=element_blank(),
        legend.position="top",
        panel.grid=element_blank())
```

2C. Nasal distribution values

```
figure2C1 <- parsN %>% select(id, rr) %>% distinct() %>%
  ggplot(aes(rr)) + theme_minimal() +
  geom_histogram(fill="#004488", alpha =0.7,
                 col="#002f5e", size =0.3, binwidth=0.5) +
  scale_x_continuous(bquote(paste("Growth rate r (", day^-1, ")"))) +
  scale_y_continuous("Number of individuals ", breaks=seq(0, 30, 7)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8.5, colour="black"),
        axis.title.y=element_text(size=9, colour="black"),
        axis.title.x=element_text(size=9, colour="black"),
        strip.text=element_blank(),
        strip.background=element_blank(), strip.placement="outside")

dat2C23 <- dat2 %>% filter(!is.na(Sim), Compartment == "Nasal") %>%
  group_by(Ind) %>% arrange(Sim, Timez) %>%
  mutate(TimeC=ifelse(Sim > 42, Timez, max(Timez)),
         TimeToPeak=-1*maxneg(TimeC),
         TimeToUnd=pmin(16, minpos(TimeC))) %>%
  ungroup() %>% select(Ind, TimeToPeak, TimeToUnd) %>% distinct()

figure2C2 <- dat2C23 %>% ggplot(aes(TimeToPeak)) +
  geom_histogram(fill="#004488", alpha =0.7, col="#002f5e",
                 size =0.3, binwidth=0.5) + theme_minimal() +
  scale_y_continuous("", breaks=seq(0, 34, 7)) +
  scale_x_continuous("Days to peak", breaks=seq(0, 50, 1)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8.5, colour="black"),
        axis.title.y=element_blank(),
        axis.title.x=element_text(size=9, colour="black"),
        strip.text=element_blank(),
        strip.background=element_blank(), strip.placement="outside")

figure2C3 <- dat2C23 %>% ggplot(aes(TimeToUnd)) +
  geom_histogram(fill="#004488", alpha =0.7, col="#002f5e",
                 size=0.3, binwidth=1) + theme_minimal() +
  scale_y_continuous("", breaks=seq(0, 36, 8)) +
  scale_x_continuous("Days from peak to undetectable",
                     breaks=c(10, 12, 14, 16),
                     labels=c("10", "12", "14", "16+")) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8.5, colour="black"),
        axis.title.y=element_blank(),
        axis.title.x=element_text(size=9, colour="black"),
        strip.text=element_blank(),
        strip.background=element_blank(), strip.placement="outside")

figure2C <- plot_grid(figure2C1, figure2C2, figure2C3, nrow=1)
```

2D. Saliva distribution values

```
figure2D1 <- parsS %>% select(id, rr) %>% distinct() %>%
  ggplot(aes(rr)) + geom_histogram(fill="#009988",
                                   alpha =0.7, col="#006b5f", size=0.3, binwidth =2) +
  scale_x_continuous(bquote(paste("Growth rate r (", day^-1, ")")),
                     breaks=seq(4, 16, 2)) + theme_minimal() +
  scale_y_continuous("Number of individuals ", breaks=seq(0, 25, 5)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8.5, colour="black"),
        axis.title.y=element_text(size=9, colour="black"),
        axis.title.x=element_text(size=9, colour="black"),
        strip.text=element_blank(),
        strip.background=element_blank(), strip.placement="outside")

data2D23 <- dat2 %>% filter(!is.na(Sim), Compartment == "Saliva") %>%
  group_by(Ind) %>% arrange(Sim, Timez) %>%
  mutate(Time=Timez - Timez[1], TimeC=ifelse(Sim > 40, Time, max(Time)),
         TimeToPeak=-1*maxneg(TimeC), TimeToUnd=pmin(16, minpos(TimeC))) %>%
  ungroup() %>% select(Ind, TimeToPeak, TimeToUnd) %>% distinct()

figure2D2 <- data2D23 %>% ggplot(aes(TimeToPeak)) +
  scale_y_continuous("", breaks=seq(0, 50, 5)) +
  scale_x_continuous("Days to peak", breaks=seq(1, 10, 1)) +
  geom_histogram(fill="#009988", alpha =0.7, col="#006b5f",
                 size =0.3, binwidth=0.5) + theme_minimal() +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8.5, colour="black"),
        axis.title.y=element_blank(),
        axis.title.x=element_text(size=9, colour="black"),
        strip.text=element_blank(),
        strip.background=element_blank(), strip.placement="outside")

figure2D3 <- data2D23 %>% ggplot(aes(TimeToUnd)) +
  geom_histogram(fill="#009988", alpha =0.7, col="#006b5f",
                 size =0.3, binwidth=1) + theme_minimal() +
  scale_y_continuous("", breaks=seq(0, 30, 2)) +
  scale_x_continuous("Days from peak to undetectable",
                     breaks=c(6, 8, 10, 12, 14, 16),
                     labels=c("6", "8", "10", "12", "14", "16+")) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8.5, colour="black"),
        axis.title.y=element_blank(),
        axis.title.x=element_text(size=9, colour="black"),
        strip.text=element_blank(),
        strip.background=element_blank(), strip.placement="outside")

figure2D <- plot_grid(figure2D1, figure2D2, figure2D3, nrow=1)
```

2E. Immnure response and age

```
figure2E <- parsN %>% select(id, phi0_mode) %>%
  left_join(IDs) %>% left_join(dat1 %>% select(Ind, Age)) %>%
  distinct() %>% mutate(log_phi=log10(phi0_mode)) %>%
  ggplot(aes(Age, log_phi)) + theme_minimal() +
  geom_line(stat="smooth", method=lm, alpha=0.75,
            lty="longdash", size=0.3) +
  ylab(bquote(atop(NA, atop("Induction of refractory", "cells"~Phi~(log10)))))+
  geom_point(col="#004488", size=2, alpha=0.5) +
  stat_cor(aes(label=paste(..rr.label.., ..p.label..,
                           sep="~`, `~")), p.accuracy=0.01, label.x=40,
           label.y.npc="top", r.accuracy=0.01, size=2.75) +
  scale_x_continuous("Age (years)", breaks=seq(20, 90, 15)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8, colour="black"),
        plot.margin=margin(t=5.5, r=5.5, b=5.5, l=-3, unit="pt"),
        axis.title.x=element_text(size=9, colour="black"),
        axis.title.y=element_text(size=12, colour="black"),
        strip.text=element_text(size=8, colour="black"))
```

2F. Timing of the peak

```
#### Since the min for nasal is center in zero, we just look at saliva ####
data2F <- dat2 %>% filter(!is.na(Sim)) %>% group_by(Ind, Compartment) %>%
  filter(Sim == min(Sim, na.rm=T)) %>% ungroup() %>%
  select(Ind, Timez, Compartment) %>%
  pivot_wider(names_from=Compartment, values_from=Timez) %>%
  filter(!is.na(Saliva), !is.na(Nasal)) %>%
  mutate(deltaSN=round(Saliva, digits=0),
         cols=case_when(deltaSN == 0 ~ "gray30", deltaSN > 0 ~ "#002f5e",
                        deltaSN < 0 ~ "#006b5f", TRUE ~ NA_character_),
         fills=case_when(deltaSN == 0 ~ "gray", deltaSN > 0 ~ "#004488",
                         deltaSN < 0 ~ "#009988", TRUE ~ NA_character_))

figure2F <- data2F %>% ggplot(aes(deltaSN, fill=fills, color=cols)) +
  theme_minimal() + geom_histogram(binwidth=1, size=0.25, alpha =0.7) +
  scale_fill_identity("", guide="legend", labels=c("Nasal peaks earlier",
                               "Saliva peaks earlier", "Similar timing")) +
  scale_color_identity("", guide="legend", labels=c("Nasal peaks earlier",
                                "Saliva peaks earlier", "Similar timing")) +
  scale_y_continuous("Number of individuals ", breaks=seq(0,20,4)) +
  scale_x_continuous("Days between nasal and saliva peak", breaks=seq(-7,1,2))+
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        legend.title=element_blank(),
        legend.position=c(-0.02, 1.05),
        legend.justification=c(-0.02, 1.05),
        legend.text=element_text(size=6.5, colour="black"),
        legend.background=element_blank(),
        legend.key.height=unit(0.25, "cm"),
        legend.key.width=unit(0.25, "cm"),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"))
```

Models figure 3

Saturation model (3C)

```
generate_popParas_corr <- function(paras_pop, num){
  inds_mean <- c(1, 2)
  inds_sd <- c(4:5)
  mean_2 <- paras_pop$value[1];mean_4=paras_pop$value[2]
  sd_2 <- paras_pop$value[4];sd_4=paras_pop$value[5]
  corr <- paras_pop$value[6];
  cov24 <- corr*sd_2*sd_4
  sigma <- rbind(c(sd_2^2, cov24), c(cov24, sd_4^2))
  mu <- c(log(mean_2), log(mean_4))
  res <- mvrnorm(n=num, mu=mu, Sigma=sigma)

  Paras <- matrix(0, num, 3)
  Paras[, 1:2] <- exp(res)
  Paras[, 3] <- paras_pop$value[3]
  return(Paras)
}

#### Generating upper and lower bounds
paras_pop <- read_csv("../Data/params_pop_cellCulture.csv")
aa <- -0.25
bb <- 11.5
CNs_sim <- seq(10, 50, by=0.5)
VLs_sim <- 10^(CNs_sim*aa+bb)
#VLs <- 10^(CNs*aa+bb)

numRuns <- 1000
Paras <- generate_popParas_corr(paras_pop, numRuns)

Vm <- Paras[, 1]
hh <- Paras[, 2]
Km <- Paras[, 3]
Sims <- matrix(-1, length(Paras[, 1]), length(VLs_sim))

for(i in 1:length(VLs_sim)){
  Sims[, i]=1-exp(-Vm/(1+(Km/VLs_sim[i])^hh))
}
```

```r
CNs_sim_lower <- rep(0, length(VLs_sim))
CNs_sim_upper <- rep(0, length(VLs_sim))
CNs_sim_median <- rep(0, length(VLs_sim))
CNs_sim_mean <- rep(0, length(VLs_sim))

for (i in 1:length(VLs_sim)){
  CNs_sim_lower[i] <- pmax(quantile(Sims[, i], 0.05, na.rm =TRUE), 0)
  CNs_sim_upper[i] <- pmax(quantile(Sims[, i], 0.95, na.rm =TRUE), 0)
  CNs_sim_median[i] <- median(Sims[, i])
  CNs_sim_mean[i] <- mean(Sims[, i])
}

data3CS <- tibble(CN=CNs_sim, Mean=CNs_sim_mean,
                  Lower=CNs_sim_lower, Upper=CNs_sim_upper)
```

Trajectories (3D)

```r
#### Saturation model prediction ####
paras_saturation <- read_csv("../Data/params_cellCulture.csv")
iter <- nrow(paras_saturation)
Model_infect <- matrix(0, iter, length(VLs_sim))
Model_sat <- matrix(0, iter, length(VLs_sim))

for (i in 1:iter){
  Vm <- unlist(paras_saturation[i, 5])
  hh <- unlist(paras_saturation[i, 6])
  Km <- unlist(paras_saturation[i, 7])
  Model_infect[i, ] <- Vm/(1+(Km/VLs_sim)^hh)
  Model_sat[i, ] <- 1-exp(-Vm/(1+(Km/VLs_sim)^hh))
}

id <- as.vector(t(matrix(seq(1:56), 56, length(VLs_sim))))
VLs <- rep(VLs_sim, iter)
Infect <- as.vector(t(Model_infect))

data3D <- tibble(id=id, SimVL=VLs, Infectiousness=Infect) %>%
  mutate(logVL=pmax(log10(SimVL), 1e-1)) %>%
  mutate(Sim=(11.35 - logVL)/0.25) %>% filter(Sim < 42)
```

Total infectiousness nasal (3E)

```r
tEND <- 30
paras_culture <- read_csv('../Data/params_cellCulture.csv')
pars_MonolixN <- read_csv("../Data/params_nasal_VL.csv")

paras <- as.matrix(pars_MonolixN[, 8:13])
numPts <- nrow(paras)
AUC1 <- NULL
```

```r
for (i in 1:numPts){
  paras_ODE <- paras[i, ]
  paras_ODE[1] <- 0
  out <- runODEN(as.numeric(paras_ODE))
  xx <- pmax(out[, 'VL'], 0)
  ind_hh <- which(paras_culture$id==pars_MonolixN$id[i])
  Vm <- as.numeric(paras_culture[ind_hh, 5])
  hh <- as.numeric(paras_culture[ind_hh, 6])
  Km <- as.numeric(paras_culture[ind_hh, 7])

  Infectiousness <- Vm/(1+(Km/xx)^hh)
  AUC1[i] <- sum(Infectiousness)*0.1
}

dat3E <- tibble(id=pars_MonolixN$id, Infectiousness=AUC1)
```

Figure 3.

3A. Days of viral positivity

```r
#### Virus_pos_days == 5.9 are negative cultures
figure3A <- dat1 %>%
  mutate(Virus_pos_days=ifelse(Virus_pos_days == 5.9, NA, Virus_pos_days)) %>%
  group_by(Ind) %>% summarise(n=sum(!is.na(Virus_pos_days))) %>%
  ungroup() %>% count(n, name="Count") %>%
  mutate(n=as_factor(n)) %>% ggplot(aes(x=n, y=Count)) +
  geom_col(fill="#B2182B", alpha=0.6, width=1,
           color="#700f1b", size=0.3) + theme_minimal() +
  scale_x_discrete("Days with a positive viral culture", expand=c(0.1, 0)) +
  scale_y_continuous("Number of individuals", breaks=seq(0, 12, 3)) +

  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"))
```

3B. Viral results by age

```r
figure3B <- dat1 %>% filter(Virus_pos_days %in% 1:5) %>%
  count(Ind, Age) %>% ggplot(aes(x=Age, y=n)) +
  geom_line(stat="smooth", method=lm, alpha=0.75,
            lty="longdash", size=0.3) +
  geom_point(color="#B2182B", alpha=0.5, size=2) +
  scale_x_continuous("Age (years)", breaks=seq(20, 95, 12)) +
  theme_minimal() + stat_cor(aes(label=
                                   paste(..rr.label.., ..p.label.., sep="~`, `~")),
                             label.x=20, label.y=9, p.accuracy=0.01,
                             r.accuracy=0.01, size=2.75) +
  scale_y_continuous("Positive viral culture (days) ",
                     breaks=seq(1, 10, 2)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"))
```

3C. Viral culture positivity and nasal CN

```
dat3CO <- dat1 %>% select(Nasal_CN, Virus_pos_days) %>%
  mutate(Nasal_CN=round(Nasal_CN, digits=0),
         Virus_cult=case_when(Virus_pos_days %in% 2:5 ~ "Pos",
           Virus_pos_days == 5.9 ~ "Neg", TRUE ~ NA_character_)) %>%
  count(Nasal_CN, Virus_cult) %>%
  pivot_wider(names_from=Virus_cult, values_from=n, values_fill=0) %>%
  mutate(Total=Pos + Neg, Prop=Pos/Total)

figure3C <- ggplot(data3CS, aes(x=CN)) + theme_minimal() +
  geom_ribbon(aes(ymin=Lower, ymax=Upper), fill="lightskyblue3", alpha=0.2) +
  geom_line(aes(y=Mean, col="M"), size=1) +
  geom_point(data=dat3CO, aes(x=Nasal_CN, y=Prop, col="O"), size=2, alpha=0.75) +
  scale_y_continuous("Positive viral culture (%)", breaks=seq(0, 1, 0.2),
                     labels=scales::percent_format(accuracy=1)) +
  scale_color_manual("", breaks=c("O","M"), values=c("#004488","lightskyblue3"),
                     labels=c("Observed", "Saturation model"),
                     guide=guide_legend(override.aes=list(shape=c(19, NA),
                                        linetype=c("blank", "solid")))) +
  scale_x_continuous("Nasal CN value", breaks=seq(10, 40, 5), limits=c(9,41)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        legend.position=c(1, 1), legend.justification=c(1, 1),
        legend.title=element_blank(),
        legend.text=element_text(size=9, colour="black"),
        legend.key.height=unit(0.4, "cm"),
        legend.key.width=unit(0.4, "cm"),
        legend.background=element_blank(),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"))
```

3D. Viral load and virus shed

```
figure3D <- data3D %>% ggplot(aes(x=Sim)) + theme_minimal() +
  geom_line(aes(y=Infectiousness, group=id, col="I"), alpha=0.5, size=0.5) +
  scale_color_manual("", values="#004488", labels="Individual\ntrajectories") +
  scale_x_continuous("Nasal CN value", breaks=seq(10,40,6), limits=c(9.5,41))+
  scale_y_continuous("Infectious virus shed (a.u.) ", breaks=seq(0,30,7)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"),
        legend.position=c(1, 1), legend.justification=c(1, 1),
        legend.title=element_blank(),
        legend.text=element_text(size=9, colour="black"),
        legend.key.height=unit(0.4, "cm"),
        legend.key.width=unit(0.4, "cm"),
        legend.background=element_blank(),
        panel.grid=element_blank())
```

3E. Infectiousness histogram AUC

```
fit.gamma <- fitdist(dat3E$Infectiousness, distr="gamma", method="mle")
dfGamma <- tibble(x=seq(min(dat3E$Infectiousness)-2,
                        max(dat3E$Infectiousness), by=0.1)) %>%
  mutate(density=pgamma(x, shape=fit.gamma$estimate[1],
                        rate=fit.gamma$estimate[2])) %>%
  mutate(density2=lead(density) - density, x=lead(x),
         dens=density2*nrow(dat3E)*10*7.5)

figure3E <- dat3E %>% ggplot(aes(Infectiousness)) +
  geom_histogram(fill="#004488", alpha =0.7, binwidth=7.5,
                 col="#002f5e", size =0.3) + theme_minimal() +
  scale_x_continuous('Total infectious virus shed (a.u.)',
                     breaks=seq(0,120,15)) +
  scale_y_continuous("Number of individuals", breaks=seq(0,15,3)) +
  geom_line(data=dfGamma, aes(x, dens, col="M"), size=1) +
  scale_colour_manual("", values="darkgray", labels="Gamma\ndistribution") +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        legend.position=c(1, 1), legend.justification=c(1, 1),
        legend.title=element_blank(),
        legend.text=element_text(size=9, colour="black"),
        legend.background=element_blank(),
        axis.text=element_text(size=8, colour="black"),
        axis.title=element_text(size=9, colour="black"))
```

3F. Infectiousness and Age

```
dat3F <- dat3E %>% left_join(IDs) %>%
  left_join(dat1 %>% select(Ind, Age) %>% distinct())

figure3F <- dat3F %>% ggplot(aes(x=Age, y=Infectiousness)) +
  geom_line(stat="smooth", method=lm, alpha=0.75, lty="longdash", size=0.3) +
  geom_point(color="#004488", alpha=0.5, size=2) + theme_minimal() +
  scale_x_continuous("Age (years)", breaks=seq(20, 95, 15)) +
  stat_cor(aes(label= paste(..rr.label.., ..p.label.., sep="~`, `~")),
           label.x=20, p.accuracy=0.01, r.accuracy=0.01, size=2.75) +
  scale_y_continuous('Total infectious virus shed (a.u.) ',
                     breaks=seq(0,105,20)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        axis.text=element_text(size=8, colour="black"),
        axis.title.x=element_text(size=9, colour="black"),
        axis.title.y=element_text(size=8, colour="black"))
```

Simulations B.1.1.7 versus non-B.1.1.7 figure 4

Nasal

```
#### Function run_popParasN ####
run_popParasN <- function(paras_pop, num, type){
  inds_mean <- c(1,2,4,5,6,8)
  inds_sd <- c(10:12,NA,13,14)
  Paras <- matrix(0,num,6)

  for (i in 1:length(inds_mean)){
    mean_para <- paras_pop$value[inds_mean[i]]
    sd_para <- paras_pop$value[inds_sd[i]]

    if (i==1) {
      Paras[,i] <- rnorm(num, mean = mean_para, sd = sd_para)
    }else if (i==4) {
      Paras[,4] <- paras_pop$value[inds_mean[4]]
    }else if (i==5) {
      Ages <- dat4N %>% filter(Type == type) %>%
        select(Age) %>% unlist() %>% setNames(NULL)
      age <- sample(Ages, num, replace = TRUE)
      Paras[,i] <- exp(rnorm(num, mean = log(mean_para)+age*paras_pop$value[7],
                             sd = sd_para))
    }else if (i==2) {
      Paras[,i] <- exp(rnorm(num, mean = log(mean_para)+type*paras_pop$value[3],
                             sd = sd_para))
    }else if (i==6) {
      Paras[,i] <- exp(rnorm(num, mean = log(mean_para)+type*paras_pop$value[9],
                             sd = sd_para))
    }else{
      Paras[,i] <- exp(rnorm(num, mean = log(mean_para), sd = sd_para))
    }
  }
  return(Paras)
}
```

```
#### Function run_simsN ####
run_simsN <- function(Paras, times){
  output <- matrix(0, length(Paras[, 1]), length(times))

  for (i in 1:length(Paras[, 1])){
    out <- runODEN(as.numeric(Paras[i, ])) %>%
      mutate(logVL=pmax(log10(VL), 1e-1), Sim=(11.35 - logVL)/0.25)
    ind <- which.min(out[, 'Sim'])
    inds0 <- which(times<= - as.numeric(Paras[i, 1]) - out[ind, 'time'])
    out[, 'time'] <- out[, 'time'] - out[ind, 'time']

    if (length(inds0)>1){
      res <- approx(out[, 'time'], out$Sim, xout=times[-inds0])
      output[i, ] <- c(rep(50, length(inds0)), res$y)
    }else {
      res <- approx(out[, 'time'], out$Sim, xout=times)
      output[i, ] <- res$y
    }
  }
  return(output)
}

#### Run simulations ####
num <- 1000
tEND <- 30
paras_popN <- read_csv("../Data/params_pop_nasal.csv")

dat4N <- dat1 %>% select(Ind, Lineage, Age) %>% distinct() %>%
  mutate(Type = ifelse(Lineage == "B.1.1.7", 1, 0)) %>%
  right_join(dat2S %>% filter(!is.na(Nasal_Sim)) %>%
               select(Ind) %>% distinct())

for (type in c(0, 1)){
  Paras <- run_popParasN(paras_popN, num, type)
  times <- seq(-13, 20, 0.1)
  out <- run_simsN(Paras, times)
  Mean <- colMeans(out, na.rm =TRUE)
  levels <- c(0.05, 0.95)
  Quantiles <- matrix(-1, length(out[1, ]), length(levels))
  for (i in 1:length(out[1, ])){
    Quantiles[i, ]=pmin(quantile(out[, i], levels, na.rm =TRUE), 50)
  }
  colnames(Quantiles)=c('Q5', 'Q95')
  if (type==0) {
    df1 <- data.frame(Time=times, Variant='non-B.1.1.7', Mean, Quantiles)
  } else{
    df2 <- data.frame(Time=times, Variant='B.1.1.7', Mean, Quantiles)
  }
}
dat4AS <- bind_rows(df1, df2)
```

Saliva

```r
run_popParasS <- function(paras_pop, num, type){
  inds_mean <- c(1, 2, 3, 5, 6, 8)
  inds_sd <- c(9:14)
  Paras <- matrix(0, num, length(inds_mean))

  for (i in 1:length(inds_mean)){
    mean_para <- paras_pop$value[inds_mean[i]]
    sd_para <- paras_pop$value[inds_sd[i]]
    if (i==3 || i==5) {
      mean_type <- paras_pop$value[inds_mean[i]+1]
    } else{ mean_type <- 0 }
    if (i==1 || i==5) {
      Paras[, i] <- rnorm(num, mean=mean_para+type*mean_type, sd=sd_para)
    } else {
      Paras[, i] <- exp(rnorm(num, mean=log(mean_para)+type*mean_type,
                              sd=sd_para))
    }
  }

  mean_2 <- paras_pop$value[inds_mean[2]]
  mean_4 <- paras_pop$value[inds_mean[4]]
  sd_2 <- paras_pop$value[inds_sd[2]]
  sd_4 <- paras_pop$value[inds_sd[4]]
  corr <- paras_pop$value[15]
  cov24 <- corr*sd_2*sd_4
  sigma <- rbind(c(sd_2^2, cov24), c(cov24, sd_4^2))
  mu <- c(log(mean_2), log(mean_4))
  res <- mvrnorm(n=num, mu=mu, Sigma=sigma)
  Paras[, c(2, 4)] <- exp(res)
  return(Paras)
}

run_simsS <- function(Paras, times){
  output <- matrix(0, length(Paras[, 1]), length(times))
  for (i in 1:length(Paras[, 1])){
    out <- runODES(as.numeric(Paras[i, ])) %>%
      mutate(logVL=pmax(log10(VL), 1e-1)) %>%
      mutate(Sim=(14.24 - logVL)/0.28)
    ind <- which.min(out[, 'Sim'])
    inds0 <- which(times <= -as.numeric(Paras[i, 1]) - out[ind, 'time'])
    out[, 'time'] <- out[, 'time'] - out[ind, 'time']

    if (length(inds0)>1){
      res <- approx(out[, 'time'], out$Sim, xout=times[-inds0])
      output[i, ] <- c(rep(50, length(inds0)), res$y)
    }else {
      res <- approx(out[, 'time'], out$Sim, xout=times)
      output[i, ] <- res$y
    }
  }
  return(output)
}

#### Params ####
num <- 1000
tEND <- 30
paras_popS <- read_csv("../Data/params_pop_saliva.csv")

for (type in c(0, 1)){
  Paras <- run_popParasS(paras_popS, num, type)
  times <- seq(-13, 20, 0.1)
  out <- run_simsS(Paras, times)
  Mean <- colMeans(out, na.rm =TRUE)
  levels <- c(0.05, 0.95)
  Quantiles <- matrix(-1, length(out[1, ]), length(levels))
  for (i in 1:length(out[1, ])){
    Quantiles[i, ]=pmin(quantile(out[, i], levels, na.rm =TRUE), 50)
  }
  colnames(Quantiles)=c('Q5', 'Q95')
  if (type==0) {
    df1 <- data.frame(Time=times, Variant='non-B.1.1.7', Mean, Quantiles)
  } else{
    df2 <- data.frame(Time=times, Variant='B.1.1.7', Mean, Quantiles)
  }
}
dat4CS <- bind_rows(df1, df2)
```

Figure 4.

4A. Nasal values

```
data4AO <- dat1 %>% select(Ind, Lineage) %>% distinct() %>% right_join(dat2) %>%
  mutate(Variant=ifelse(Lineage == "B.1.1.7", Lineage, "non-B.1.1.7")) %>%
  filter(!is.na(Obs), Compartment == "Nasal")

figure4A <- dat4AS %>%
  ggplot(aes(x=Time, colour=Variant, fill=Variant)) +
  geom_ribbon(aes(ymin=Q5, ymax=Q95), alpha=0.2, colour=NA)+
  geom_line(aes(y=Mean), size=0.4) + theme_minimal() +
  geom_hline(yintercept=42, linetype=3, size=0.3) +
  geom_point(data=data4AO, aes(x=Timez, y=Obs), size=0.4, alpha=0.5) +
  scale_y_continuous("Nasal CN values", trans="reverse", breaks=seq(10,40,10)) +
  scale_x_continuous("Days since predicted lowest nasal CN value",
                     breaks=seq(-12, 18, 3), limits=c(-6, 16)) +
  scale_color_manual(values=c("#332288", "#CC6677")) +
  scale_fill_manual(values=c("#332288", "#CC6677")) +
  coord_cartesian(ylim=c(44, 10)) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        legend.position=c(1.05, 1.05),
        legend.justification=c(1.05, 1.05),
        legend.text=element_text(size=6.5, colour="black"),
        legend.title=element_text(size=7, colour="black"),
        legend.background=element_blank(),
        legend.key.height=unit(0.35, "cm"),
        legend.key.width=unit(0.35, "cm"),
        axis.text=element_text(size=7.5, colour="black"),
        axis.title=element_text(size=8, colour="black"))
```

4B. Nasal Parameter estimates

```
dat4BD <- IDs %>% left_join(dat1) %>%
  select(Ind, id, Lineage, Age) %>% distinct() %>%
  left_join(dat2S) %>% left_join(dat3E) %>%
  mutate(Variant=ifelse(Lineage == "B.1.1.7", Lineage, "non-B.1.1.7"),
         Cols=ifelse(Variant =="non-B.1.1.7", "#CC6677", "#332288"))

dat4B <- dat4BD %>% filter(!is.na(Nasal_Sim)) %>%
  left_join(parsN %>% select(id, rr)) %>% group_by(Ind) %>%
  arrange(Nasal_Sim, Time) %>% mutate(Timez=Time - Time[1],
         TimeC=ifelse(Nasal_Sim > 42, Timez, max(Timez))) %>%
  mutate(TimeMin=-1*maxneg(TimeC), TimeMax=pmin(16, minpos(TimeC)),
         minSimCN=min(Nasal_Sim)) %>% ungroup() %>%
  select(Ind, Variant, Cols, `Growth~rate~r`=rr,
         `Days~to~peak`=TimeMin,
         `Predicted~minimun~nasal~CN`=minSimCN,
         `Total~infectious~virus~shed~(a.u.)`=Infectiousness) %>%
  distinct() %>%
  pivot_longer(`Growth~rate~r`:`Total~infectious~virus~shed~(a.u.)`,
               names_to="Parameter", values_to="value") %>%
  mutate(Parameter=as_factor(Parameter),
         Parameter=fct_relevel(Parameter, "Growth~rate~r", "Days~to~peak",
                               "Predicted~minimun~nasal~CN",
                               "Total~infectious~virus~shed~(a.u.)"))

figure4B <- dat4B %>%
  ggplot(aes(x=Variant, y= value, fill=Cols)) +
  theme_minimal() + geom_boxplot(alpha =0.7) +
  ylab("Value") + xlab("Variant") + labs(subtitle= "Nasal") +
  stat_compare_means(method="wilcox.test", aes(label=..p.signif..),
                     label.x=1.5, size=2) + scale_fill_identity() +
  facet_wrap(nrow=1, ~Parameter, scales="free", labeller=label_parsed) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        strip.text=element_text(size=7.25, colour="black"),
        axis.text=element_text(size=7, colour="black"),
        axis.title=element_text(size=8, colour="black"),
        plot.subtitle=element_text(size=9, colour="black",
                                   margin=margin(0, 0, 0, 0)))
```

4C. Saliva values

```r
data4CO <- dat1 %>% select(Ind, Lineage) %>% distinct() %>% right_join(dat2) %>%
  mutate(Variant=ifelse(Lineage == "B.1.1.7", Lineage, "non-B.1.1.7")) %>%
  rename(Time=Timez) %>% filter(Compartment == "Saliva") %>%
  group_by(Ind) %>% arrange(Sim, Time) %>% mutate(Timez=Time - Time[1]) %>%
  ungroup() %>% arrange(Ind, Timez) %>% filter(!is.na(Obs))

figure4C <- dat4CS %>%
  ggplot(aes(x=Time, colour=Variant, fill=Variant)) +
  geom_ribbon(aes(ymin=Q5, ymax=Q95), alpha=0.2, colour=NA)+
  geom_line(aes(y=Mean), size=0.4) + theme_minimal() +
  geom_hline(yintercept=40, linetype=3, size=0.3) +
  geom_point(data=data4CO, aes(x=Timez, y=Obs), size=0.4, alpha=0.5) +
  scale_y_continuous("Saliva Ct values", trans="reverse",
                     breaks=seq(0, 40, 8)) +
  coord_cartesian(ylim=c(42, 13)) +
  scale_x_continuous("Days since predicted lowest saliva Ct value",
                     breaks=seq(-12, 18, 3), limits=c(-6, 16)) +
  scale_color_manual(values=c("#332288", "#CC6677")) +
  scale_fill_manual(values=c("#332288", "#CC6677")) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        legend.position=c(1.05, 1.05),
        legend.justification=c(1.05, 1.05),
        legend.text=element_text(size=6.5, colour="black"),
        legend.title=element_text(size=7, colour="black"),
        legend.background=element_blank(),
        legend.key.height=unit(0.35, "cm"),
        legend.key.width=unit(0.35, "cm"),
        axis.text=element_text(size=7.5, colour="black"),
        axis.title=element_text(size=8, colour="black"))
```

4D. Saliva Parameter estimates

```r
dat4D <- dat4BD %>% filter(!is.na(Saliva_Sim)) %>%
  left_join(parsS %>% select(id, rr)) %>% group_by(Ind) %>%
  arrange(Saliva_Sim, Time) %>% mutate(Timez=Time - Time[1],
         TimeC=ifelse(Saliva_Sim > 40, Timez, max(Timez))) %>%
  mutate(TimeMin=-1*maxneg(TimeC), TimeMax=pmin(16, minpos(TimeC)),
         minSimCt=min(Saliva_Sim)) %>% ungroup() %>%
  select(Ind, Variant, Cols, `Growth~rate~r`=rr,
         `Days~to~peak`=TimeMin,
         `Predicted~minimun~saliva~Ct`=minSimCt) %>% distinct() %>%
  left_join(dataAllS %>% select(Ind, AUC) %>% distinct()) %>%
  rename(`Area~under~the~curve~(log10)` = AUC) %>%
  pivot_longer(`Growth~rate~r`:`Area~under~the~curve~(log10)`,
               names_to="Parameter", values_to="value") %>%
  mutate(Parameter=as_factor(Parameter),
         Parameter=fct_relevel(Parameter, "Growth~rate~r", "Days~to~peak",
                               "Predicted~minimun~saliva~Ct",
                               "Area~under~the~curve~(log10)"))

figure4D <- dat4D %>% ggplot(aes(x=Variant, y= value, fill=Cols)) +
  theme_minimal() + geom_boxplot(alpha =0.7) + scale_fill_identity() +
  ylab("Value") + xlab("Variant") + labs(subtitle= "Saliva") +
  stat_compare_means(method="wilcox.test", aes(label=..p.signif..),
                     label.x=1.5, size=2) +
  facet_wrap(nrow=1, ~Parameter, scales="free", labeller=label_parsed) +
  theme(axis.line=element_line(size=0.3, colour ="black"),
        axis.ticks=element_line(size=0.3, colour ="black"),
        panel.grid=element_blank(),
        strip.text=element_text(size=7.25, colour="black"),
        axis.text=element_text(size=7, colour="black"),
        axis.title=element_text(size=8, colour="black"),
        plot.subtitle=element_text(size=9, colour="black",
                                   margin=margin(0,0,0,0)))
```