BC COVID model

Jens von Bergmann

24/03/2021

B.1.1.7 growth rate advantage

```
ontario_data <- get_ontario_voc_data()

ontario_estimates <- ontario_data %>%
  group_map(~(lm(log_ratio ~ Date, data=.) %>% tidy())) %>%
  bind_rows()

on_ga <- ontario_estimates %>%
  filter(term=="Date") %>%
  mutate(s1=paste0(round(estimate,3)," ±",round(std.error,3))) %>%
  pull(s1)

lb <- c(0.05,0.1,0.2,0.4,0.8)
mb <- seq(2,length(lb)) %>%
  lapply(function(i){
    s=lb[i-1]
    e=lb[i]
    seq(s,e,s/4)
  }) %>%
  unlist() %>%
  unique()


on_model <- lm(log_ratio ~ Date,data=ontario_data)

on_share <- tibble(Date=voc_share$Date) %>%
              mutate(ratio=exp(predict(on_model,newdata = .))) %>%
              mutate(share=ratio_to_share(ratio))
```

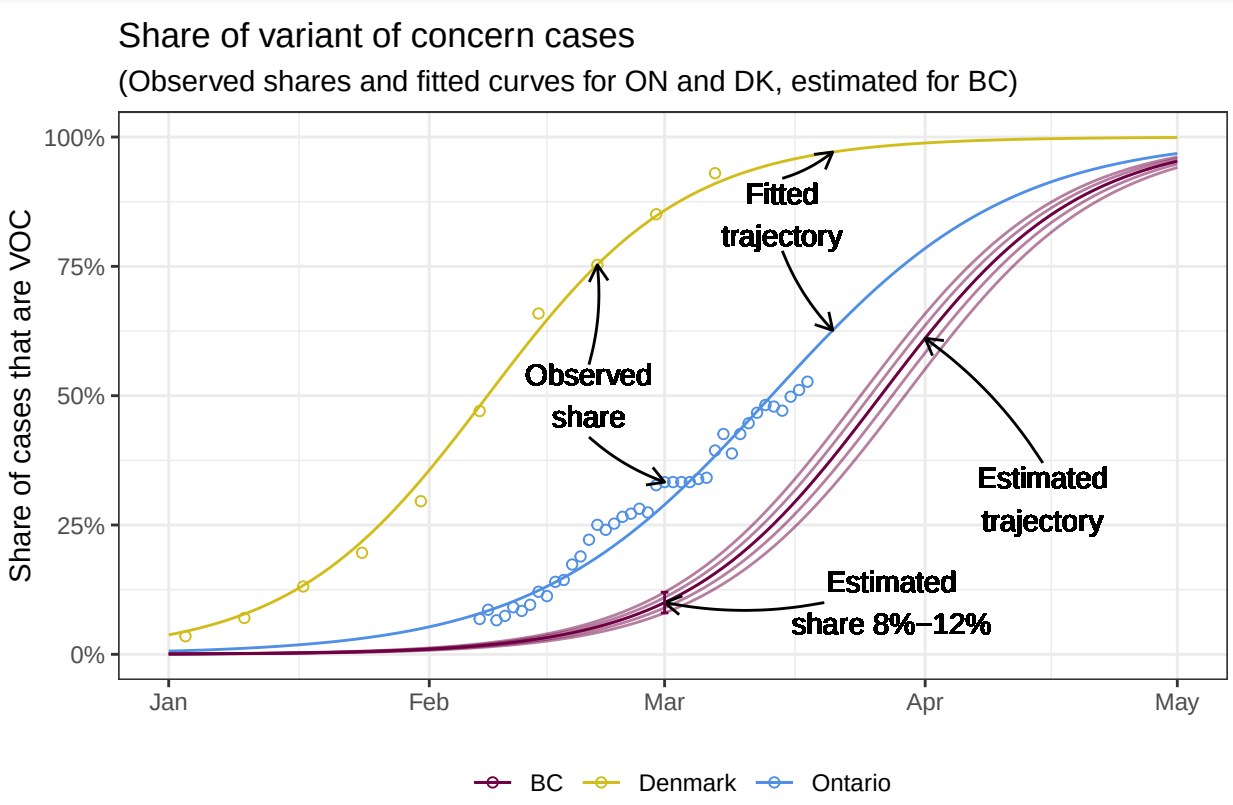
```
dk_data <- get_dk_data()
dk_model <- lm(log_ratio ~ Date,data=dk_data)

dk_share <- tibble(Date=voc_share$Date) %>%
              mutate(ratio=exp(predict(dk_model,newdata = .))) %>%
              mutate(share=ratio_to_share(ratio))
```

```
ch_data <- get_ch_data() %>%
  mutate(ratio=b117/(sequenced-b117)) %>%
  mutate(log_ratio=log(ratio+0.00001)) %>%
  mutate(share=b117/sequenced)
```

```
ch_model <- lm(log_ratio ~ Date,data=ch_data %>% filter(share>0.01))

ch_share <- tibble(Date=voc_share$Date) %>%
              mutate(ratio=exp(predict(ch_model,newdata = .))) %>%
              mutate(share=ratio_to_share(ratio))
```

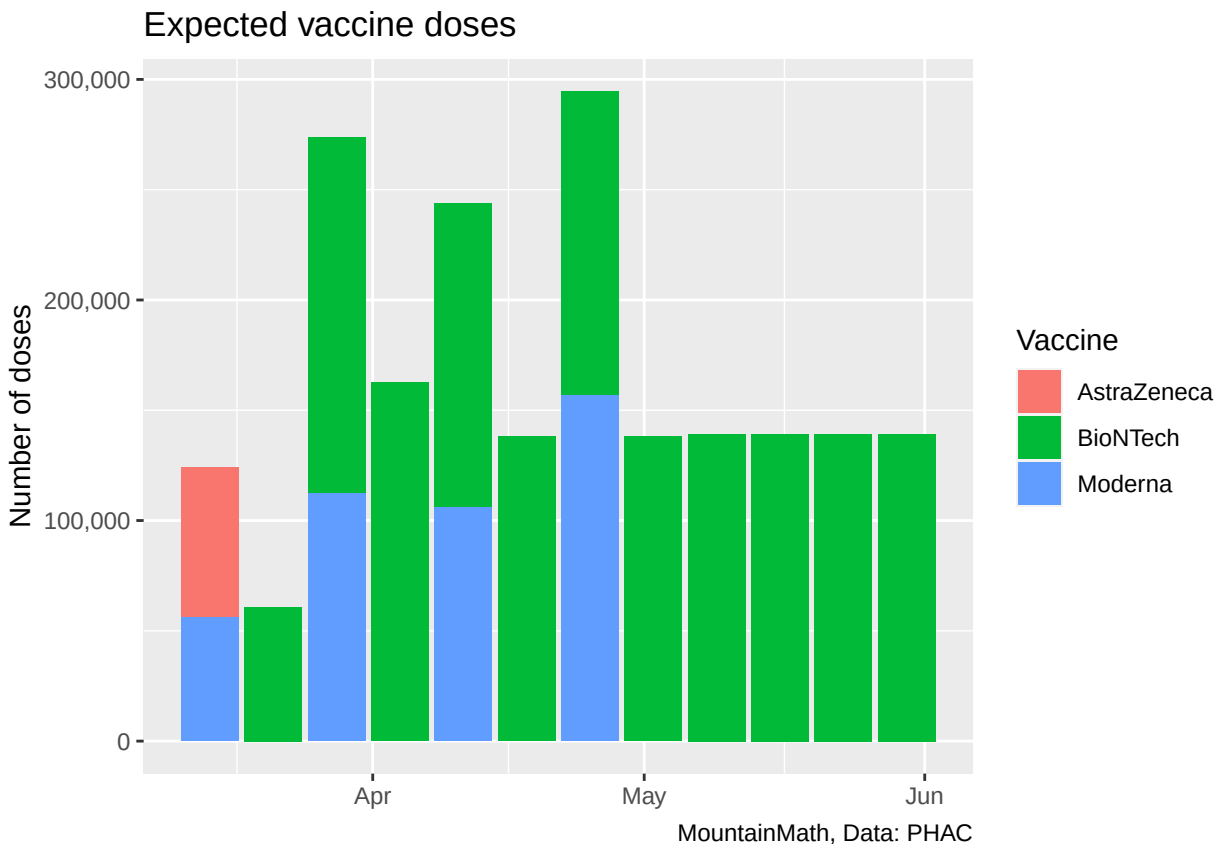
```
#bc_voc_advantage <- mean(c(on_model$coefficients[['Date']],dk_model$coefficients[['Date']]))
bc_voc_advantage <- dk_model$coefficients[['Date']] # use DK data, ON data is dodgy
bc_share <- seq(0.08,0.12,0.01) %>%
              lapply(function(s)
                get_voc_share(initial_voc_share = s,initial_date=as.Date("2021-03-01"),
                              voc_advantage = bc_voc_advantage) %>%
                  mutate(Region="BC",type=s)) %>%
              bind_rows()

school_start <- as.Date("2021-03-29")
today <- Sys.Date()
bc_share %>%
  filter(type==0.1) %>%
  mutate(Region="BC") %>%
  bind_rows(on_share %>% mutate(Region="Ontario")) %>%
  bind_rows(dk_share %>% mutate(Region="Denmark")) %>%
  #bind_rows(ch_share %>% mutate(Region="Switzerland")) %>%
  filter(Date<=as.Date("2021-05-01")) %>%
ggplot(aes(x=Date,y=share)) +
  #geom_point(shape=21,aes(color=Region)) +
  geom_line(aes(color=Region)) +
  geom_line(aes(color=Region,group=type),alpha=0.5,
            data=bc_share %>%
              filter(Date<=as.Date("2021-05-01"))) +
  geom_point(data=bind_rows(ontario_data %>% mutate(Region="Ontario"),dk_data %>% mutate(Region="Denmar
               filter(Date>=as.Date("2021-01-01")),
             aes(color=Region),shape=21) +
  scale_y_continuous(labels=scales::percent) +
  theme_bw() +
  geom_text(data=NULL,x=as.Date("2021-02-20"),y=0.5,label="Observed\nshare") +
  geom_curve(data=dk_data %>% filter(Date==as.Date("2021-02-21")),
             arrow = arrow(length=unit(0.25,"cm")),curvature = 0.1,
             x=as.Date("2021-02-20"),y=0.56,
             aes(xend=Date,yend=share)) +
  geom_curve(data=ontario_data %>% filter(Date==as.Date("2021-03-01")),
             arrow = arrow(length=unit(0.25,"cm")),curvature = 0.1,
             x=as.Date("2021-02-20"),y=0.42,
             aes(xend=Date,yend=share)) +
  geom_text(data=NULL,x=as.Date("2021-03-15"),y=0.85,label="Fitted\ntrajectory") +
  geom_curve(data=dk_share %>% filter(Date==as.Date("2021-03-21")),
             arrow = arrow(length=unit(0.25,"cm")),curvature = 0.1,
             x=as.Date("2021-03-15"),y=0.92,
             aes(xend=Date,yend=share)) +
  geom_curve(data=on_share %>% filter(Date==as.Date("2021-03-21")),
             arrow = arrow(length=unit(0.25,"cm")),curvature = 0.1,
             x=as.Date("2021-03-15"),y=0.78,
             aes(xend=Date,yend=share)) +
  geom_text(data=NULL,x=as.Date("2021-04-15"),y=0.3,label="Estimated\ntrajectory") +
```

```
geom_curve(data=bc_share %>% filter(Date==as.Date("2021-04-01"),type==0.1),
           arrow = arrow(length=unit(0.25,"cm")),curvature = 0.1,
           x=as.Date("2021-04-15"),y=0.37,
           aes(xend=Date,yend=share)) +
geom_text(data=NULL,x=as.Date("2021-03-28"),y=0.1,label="Estimated\nshare 8%-12%") +
geom_curve(data=bc_share %>% filter(Date==as.Date("2021-03-01"),type==0.1),
           arrow = arrow(length=unit(0.25,"cm")),curvature = -0.1,
           x=as.Date("2021-03-20"),y=0.1,
           aes(xend=Date,yend=share)) +
theme(legend.position = "bottom") +
geom_errorbar(data=bc_share %>% filter(Date==as.Date("2021-03-01"),type==0.1),
                                    y=0.1,ymin=0.08,ymax=0.12,
              aes(color=Region)) +
scale_color_manual(values=sanzo::trios$c157) +
#scale_color_manual(values=c(sanzo::trios$c157,"black")) +
#geom_point(data=ch_data %>% filter(Date>=as.Date("2021-01-01")),shape=21) +
labs(title="Share of variant of concern cases",
     subtitle = "(Observed shares and fitted curves for ON and DK, estimated for BC)",
     x=NULL,color=NULL,
     y="Share of cases that are VOC",
     caption="MountainMath, Data: Ontario Daily Epidemiologic Summary, Danish Covid-19 Genome Consorti
```



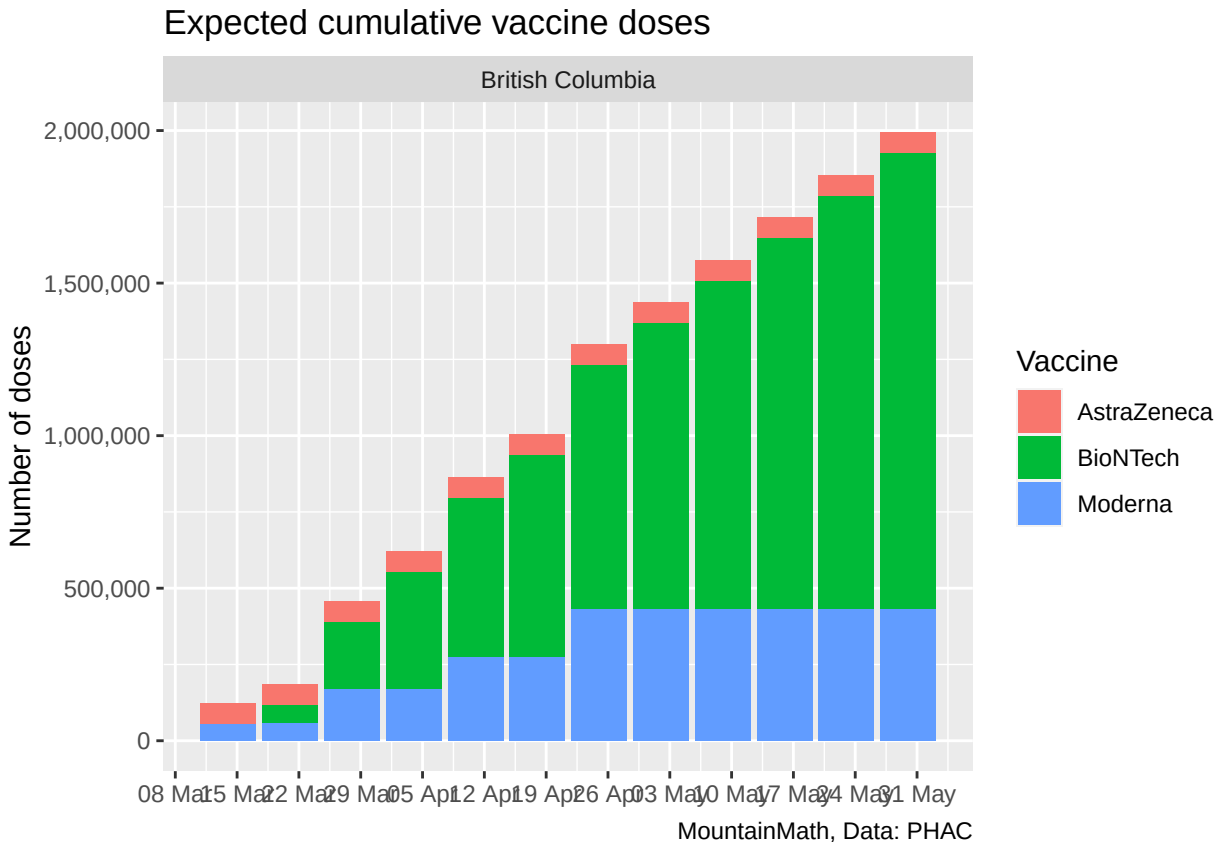
Vaccination schedule

```
schedule <- get_canada_vaccine_projections()

schedule %>%
  filter(`Distribution location`=="British Columbia") %>%
  filter(week>=as.Date("2021-03-01")) %>%
  ggplot(aes(x=week,y=value,fill=type)) +
  geom_bar(stat="identity") +
  scale_y_continuous(labels=scales::comma) +
  labs(title="Expected vaccine doses",y="Number of doses",
       x=NULL, fill="Vaccine",
       caption="MountainMath, Data: PHAC")
```



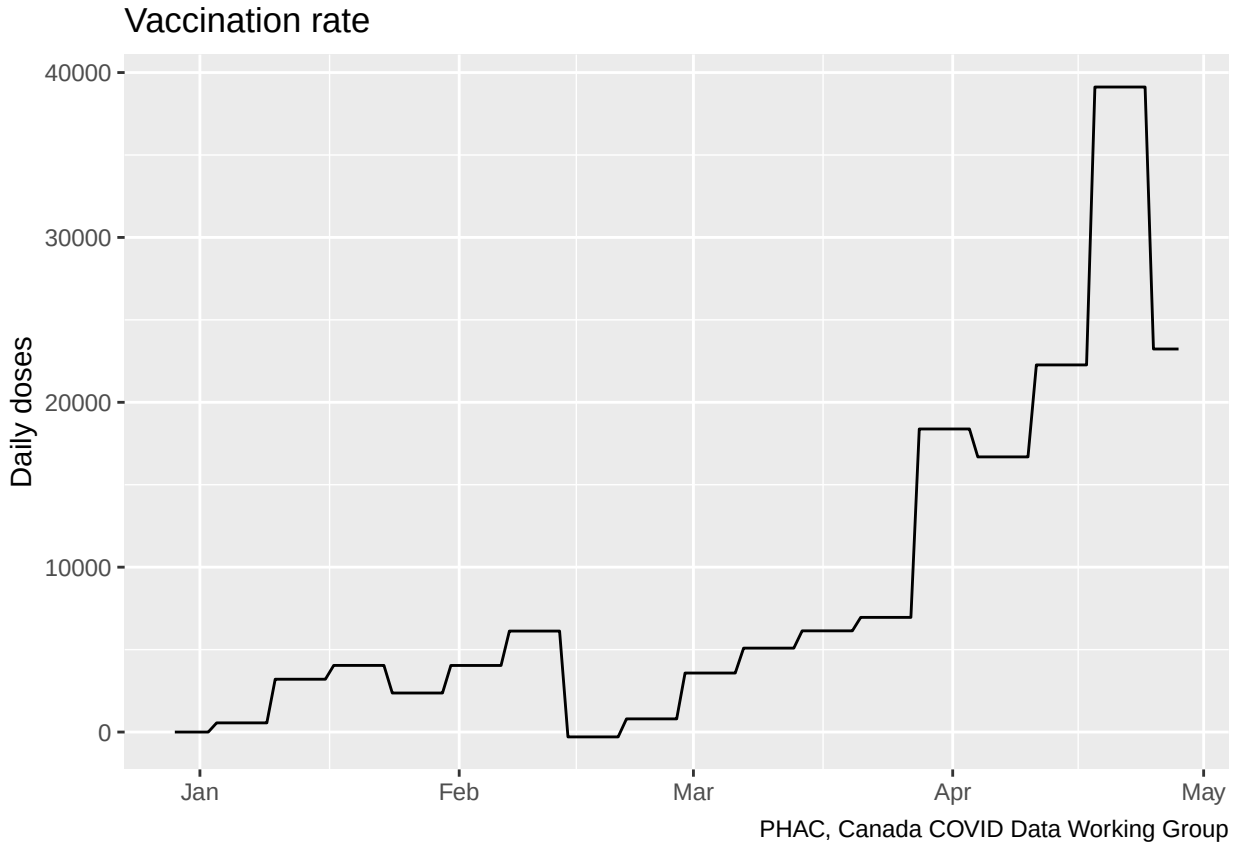

```
schedule %>%
  filter(`Distribution location`=="British Columbia") %>%
  filter(week>=as.Date("2021-03-01")) %>%
  mutate(value=coalesce(value,0)) %>%
  group_by(type) %>%
  arrange(week) %>%
  mutate(cum=cumsum(value)) %>%
  ggplot(aes(x=week,y=cum,fill=type)) +
  geom_bar(stat="identity") +
  scale_y_continuous(labels=scales::comma) +
  facet_wrap("`Distribution location`") +
  scale_x_date(breaks="week",date_labels = "%d %b") +
  labs(title="Expected cumulative vaccine doses",y="Number of doses",
```

```
       x=NULL, fill="Vaccine",
       caption="MountainMath, Data: PHAC")
```




```
Population <- read_csv(here::here("data/ha_pop.csv")) %>%
  filter(`Health Authority`=="British Columbia") %>%
  pull(Total)


past_deliveries <- get_canada_covid_working_group_timeseries(type="dvaccine") %>%
  filter(shortProvince=="BC") %>%
  filter(dvaccine>0) %>%
  select(Date=date_vaccine_distributed,value=dvaccine)
past_second <- get_canada_covid_working_group_timeseries(type="cvaccine") %>%
  filter(shortProvince=="BC") %>%
  filter(cvaccine>0) %>%
  select(Date=date_vaccine_completed,value=cvaccine)


vaccination_schedule <- schedule %>%
  filter(`Distribution location`=="British Columbia") %>%
  group_by(Date=week) %>%
  summarize(value=sum(value,na.rm=TRUE)) %>%
  filter(Date>max(past_deliveries$Date)) %>%
  bind_rows(past_deliveries) %>%
  bind_rows(past_second %>% mutate(value=-value)) %>%
  mutate(D=Date,
         Week=MMWRweek::MMWRweek(Date)$MMWRweek,Year=MMWRweek::MMWRweek(Date)$MMWRyear) %>%
```

```
  mutate(Date=MMWRweek::MMWRweek2Date(Year,Week,1)) %>%
  group_by(Date) %>%
  summarize(value=sum(value,na.rm=TRUE)) %>%
  arrange(Date) %>%
  mutate(time=difftime(Date,model_start_date,units = "day") %>% as.integer + 21) %>%
  bind_rows(tibble(time=0,value=0)) %>%
  arrange(time) %>%
  mutate(duration=coalesce(lead(time)-time,7)) %>%
  mutate(rate=value/Population/duration)

vaccination_rate <- approxfun(vaccination_schedule %>% select(time,rate),
                              method="constant",rule=2)

tibble(time=seq(80,200)) %>%
  mutate(Date=model_start_date+time) %>%
  mutate(vr=vaccination_rate(time)) %>%
  mutate(p=Population*vr) %>%
  ggplot(aes(x=Date,y=p)) +
  geom_line() +
  labs(title="Vaccination rate",caption="PHAC, Canada COVID Data Working Group",
       x=NULL,y="Daily doses")
```



BC Case data

```
bc_cases_raw <- get_british_columbia_case_data() %>%
  count(Date=`Reported Date`,name="Cases") %>%
  filter(Date>=as.Date("2020-03-01")) %>%
  mutate(day=difftime(Date,model_start_date,units = "day") %>% as.integer) %>%
  mutate(stl=add_stl_trend_m(.data$Cases)) %>%
  mutate(Trend=stl$trend,
         Random=stl$remainder,
         Seasonal=stl$seasonal) %>%
  select(-stl) %>%
  mutate(Adjusted=Cases/Seasonal)

get_bc_estimate <- function(initial_voc_share = 0.1,initial_date = as.Date("2021-02-27")){
  bc_cases_raw %>%
  inner_join(get_voc_share(initial_voc_share=initial_voc_share,initial_date=initial_date) %>%
               select(Date,voc_share=share),by="Date") %>%
  mutate(VOC=Trend*voc_share,`Non-VOC`=Trend*(1-voc_share)) %>%
  pivot_longer(c("VOC","Non-VOC"),names_to = "type",values_to = "value") %>%
  mutate(type=factor(type,levels=c("VOC","Non-VOC")))
}
bc_estimate <- get_bc_estimate()

bc_combined <- bc_cases_raw %>%
  filter(Date<=min(voc_share$Date)) %>%
  mutate(type="Combined",value=Trend)
```

```
voc_advantage <- 0.084
#voc_advantage <- bc_voc_advantage
SIR_model <- function(time,yinit,parameters){
    with(as.list(c(yinit,parameters)), {
      vr <- vaccination_rate(time)
      beta <- beta_function(time)
      jerks0 <- jerk0_function(time)

      dSusceptible <- -(beta*Infected0+(beta+voc_advantage)*Infected1)*Susceptible - vr * vaccination_ef
      dCumulativeInfected0  <- beta*Infected0*Susceptible + jerks0
      dCumulativeInfected1  <- (beta+voc_advantage)*Infected1*Susceptible
      dInfected0  <- beta*Infected0*Susceptible - gamma*Infected0 + jerks0
      dInfected1  <- (beta+voc_advantage)*Infected1*Susceptible - gamma*Infected1
      dRecovered <- gamma*(Infected0+Infected1)
      dVaccinated <- vr * vaccination_efficay

      return(list(c(dSusceptible,
                    dInfected0,
                    dInfected1,
                    dCumulativeInfected0,
                    dCumulativeInfected1,
                    dRecovered,
                    dVaccinated)))})
  }
# parameters
# vaccination_rate - daily share of population gaining immunity via vaccination, function of time
# beta0, beta1 - infectivity
```

```r
# gamma - clearance rate
model.vaxx <- function(times,yinit,parameters){
  ode(func = SIR_model,times = times,y = yinit,parms = paramters) %>%
    as_tibble() %>%
    mutate_all(as.numeric)
}


truth <- bc_cases_raw %>%
  filter(Date>=model_start_date) %>%
  select(time=day,value=Adjusted) %>%
  mutate(var_share=0.1)

initial_susceptible <- 0.98
initial_vaccinated <- 0
initial_recovered <- 1-initial_susceptible-initial_vaccinated
vaccination_efficay <- 0.9
gamma <- 1/6

initial_conditions_for_parameters <- function(parameters){
  initial_cases <- parameters[['initial_cases']]
  initial_voc_share <- parameters[['initial_voc_share']]
  base_beta <- parameters[['beta_v0']]
  i0 <- (1-initial_voc_share)*initial_cases/ ((base_beta)*initial_susceptible)
  i1 <- initial_voc_share*initial_cases/  ((voc_advantage+base_beta)*initial_susceptible)
  c(Susceptible=initial_susceptible,
    Infected0=i0,
    Infected1=i1,
    CumulativeInfected0=initial_cases*(1-initial_voc_share),
    CumulativeInfected1=initial_cases*initial_voc_share,
    Recovered=initial_recovered-i0-i1,
    Vaccinated=initial_vaccinated)
}

tibble_for_prefix <- function(parameters,prefix){
  p <-names(parameters)
  betas <- p[grepl(paste0(prefix,"_v"),p)] %>%
    str_extract("\\d+$") %>%
    lapply(function(d){
      tibble(day=parameters[[paste0(prefix,"_d",d)]],
             value=parameters[[paste0(prefix,"_v",d)]])
    }) %>%
    bind_rows()
}


betas_from_params <- function(parameters){
  tibble_for_prefix(parameters,"beta") %>%
    bind_rows(mutate(.,day=lead(day)-1.5/gamma)) %>%
    filter(!is.na(value)) %>%
    arrange(day) %>%
    approxfun(method="linear",rule=2)
}
```

```r
jerks_from_params <- function(parameters){
  jerks <- tibble_for_prefix(parameters,"jerk0")
  if (nrow(jerks)==0) return(function(d)0)
  bind_rows(jerks,
            jerks %>% mutate(.,day=day+3/gamma,value=0),
            jerks %>% mutate(.,day=day-1,value=0)) %>%
    arrange(day) %>%
    approxfun(method="linear",rule=2)
}


ode_for_parameters <- function(parameters,times= truth$time){
  ode(func = SIR_model,
      times = times, y = initial_conditions_for_parameters(parameters),
      parms = c(parameters,
                beta_function = betas_from_params(parameters),
                jerk0_function=jerks_from_params(parameters),
                vaccination_rate=vaccination_rate,
                voc_advantage=voc_advantage,gamma=gamma))
}



initial_voc_day <- bc_cases_raw %>% filter(Date==initial_voc_date) %>% pull(day)
rescale_ode_results <- function(out){
  out %>%
    as_tibble() %>%
    mutate_at(vars(-one_of("time")),function(d)d*Population) %>%
    mutate(Cases0=coalesce(CumulativeInfected0-lag(CumulativeInfected0),CumulativeInfected0)) %>%
    mutate(Cases1=coalesce(CumulativeInfected1-lag(CumulativeInfected1),CumulativeInfected1)) %>%
    mutate(value=(Cases0+Cases1)) %>%
    mutate(var_share =filter(.,time==initial_voc_day) %>% mutate(share=Cases1/value) %>% pull(share)) %
    mutate(day=as.integer(time)) %>%
    mutate(Date=model_start_date+day) %>%
    as_tibble()

}

##Cost function with sum of squared residuals:
cost <- function(parameters){
  out <- ode_for_parameters(parameters) %>%
    rescale_ode_results() %>%
    select(time,value,var_share,Cases0,Cases1)

  ct <-  modCost(out %>% as.data.frame() %>% mutate(var_share=var_share*1000),
                 truth %>% as.data.frame() %>% mutate(var_share=var_share*1000) ,
                 x="time")

  return(ct)
}
```

```
parameters.start <- c(beta_v0=0.227,beta_d0=0,beta_v1=0.171,beta_d1=41,
                      jerk0_v0=150/Population,
                      jerk0_d0=difftime(as.Date("2021-02-17"),model_start_date,units="day") %>% as.inte
                      initial_cases=filter(bc_cases_raw,Date==model_start_date)$Trend/Population,
                      initial_voc_share=0.00001)
parameters.start<-c(beta_v0=0.22,beta_d0=0,beta_v1=0.165,beta_d1=45,#beta_v2=0.176,beta_d2=140,
                                  jerk0_v0=0.00003,jerk0_d0=130,
                    initial_cases=2.650634e-05,initial_voc_share=1.3e-06)
parameters.start<-c(beta_v0=0.22,beta_d0=0,beta_v1=0.165,beta_d1=45,#beta_v2=0.176,beta_d2=140,
                                  jerk0_v0=0.00003,jerk0_d0=130,
                    initial_cases=2.650634e-05,initial_voc_share=1.3e-06)
fit <- modFit(f = cost, p = parameters.start, control = list(nprint = 1),
              #method="SANN",
              lower=c(beta_v0=0,beta_d0=0,beta_v1=0,beta_d1=30,#beta_v2=0,beta_d2=120,
                      jerk0_v0=0,jerk0_d0=129,
                      initial_cases=0,initial_voc_share=0),
              upper=c(beta_v0=0.5,beta_d0=1,beta_v1=0.5,beta_d1=50,#beta_v2=0.5,beta_d2=150,
                      jerk0_v0=300/Population,jerk0_d0=131,
                      initial_cases=1000,initial_voc_share=1))
```

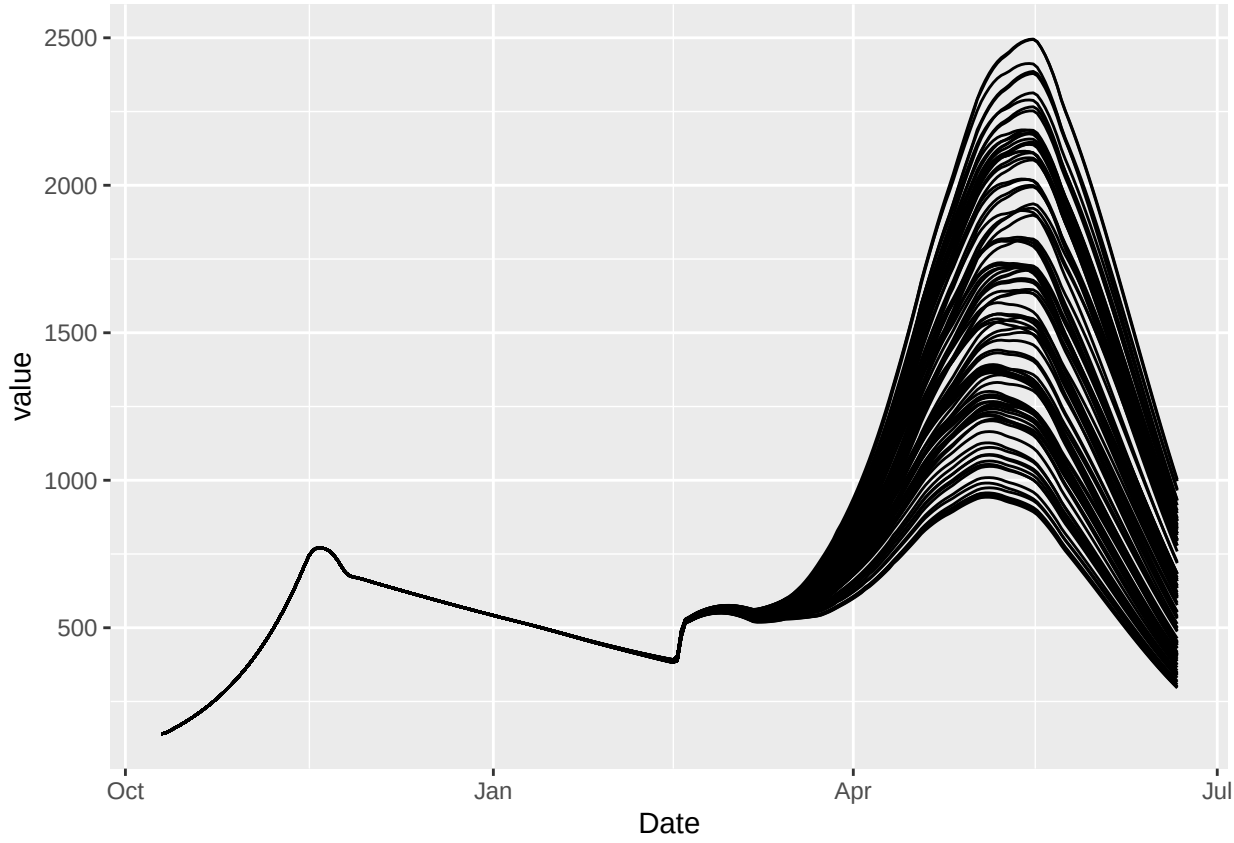
```
## It.    0, RSS =     883261, Par. =   -0.19076 -1.63312e+16  -0.591398          1  0.0224146
## It.    1, RSS =     723411, Par. =  -0.208562 -1.63312e+16  -0.592546    1.17238  -0.324541  -0.6146
```

```
#c(fit$par['initial_cases']*Population,fit$par['beta'],fit$par['initial_voc_share'])

#fit$par
```

Simple sensitivity to WT Covid growth rate and initial VOC share

```
spar <- fit$par
spar[['beta_v2']]<-spar[['beta_v1']]
spar[['beta_d2']]<-max(truth$time)

pr <- data.frame(row.names = c("beta_v2","initial_voc_share"),
                 min=c(spar[['beta_v2']]*0.95,spar[['initial_voc_share']]*0.8),
                 max=c(spar[['beta_v2']]*1.05,spar[['initial_voc_share']]*1.2)
                 )


out.s <- sensRange(function(parameters)ode_for_parameters(parameters,times=seq(0,max(truth$time)+90)) %
                     rescale_ode_results(),
                   parms=spar,sensvar = c("value"),
                   parRange=pr)
o <- out.s %>%
  pivot_longer(-one_of(rownames(pr)),
               #names_sep="\\.+",
               names_pattern = "(Cases1|Cases0|value)\\.*(\\d+)",
               names_to=c(".value","time")) %>%
  mutate(Date=model_start_date+as.integer(time))

s<-summary(out.s) %>%
  as_tibble() %>%
```

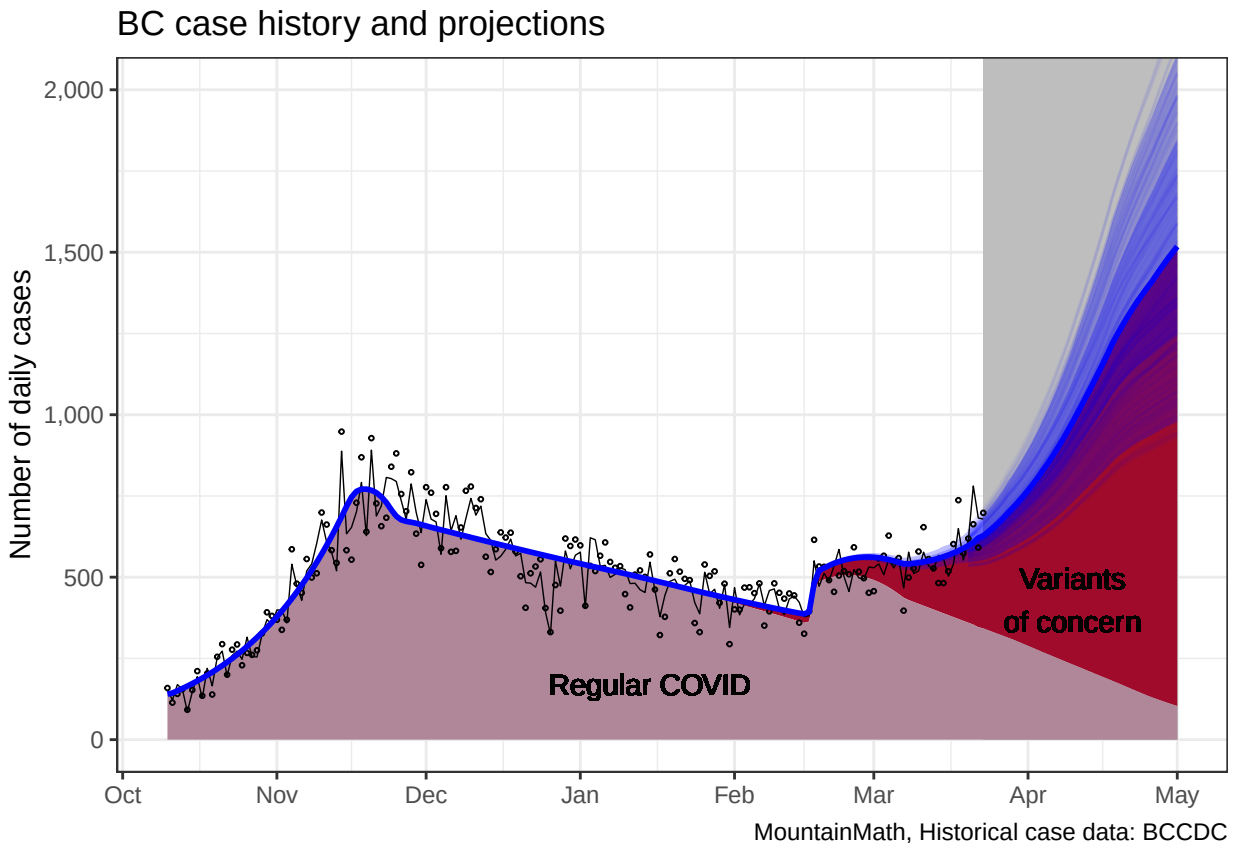
```
  mutate(Date=model_start_date+as.integer(x))


ggplot(o,aes(x=Date,y=value,group=interaction(beta_v2,initial_voc_share))) +
  geom_line()
```



Visualize model output

```
max_date <- as.Date("2021-05-01")
variant_colours <- c("Combined"="steelblue",setNames(sanzo::duos$c035,c("Non-VOC","VOC")))
out <- ode_for_parameters(fit$par,times=seq(0,max(truth$time)+130)) %>%
  rescale_ode_results()

g<-bc_cases_raw %>%
  filter(Date>=model_start_date) %>%
  ggplot(aes(x=Date)) +
  geom_rect(data=NULL,xmin=max(bc_cases_raw$Date),xmax=max_date,ymin=0,ymax=Inf,fill="grey",alpha=0.5) +
  geom_area(data=out %>%
              select(Date,VOC=Cases1,`Non-VOC`=Cases0) %>%
              pivot_longer(-Date) %>%
              filter(value>=1) %>%
              mutate(name=factor(name,levels=c("VOC","Non-VOC"))) %>%
              filter(Date<=max_date),
            aes(fill=name,y=value)) +
  geom_ribbon(data=s %>% filter(Date<=max_date),aes(ymin=q05,ymax=q95),alpha=0.25,fill="blue") +
```

```
  geom_ribbon(data=s %>% filter(Date<=max_date),aes(ymin=q25,ymax=q75),alpha=0.25,fill="blue") +
  geom_line(data=o %>% filter(Date>=Sys.Date()-4,Date<=max_date),
            aes(y=value,group=interaction(beta_v2,initial_voc_share)),alpha=0.05,
            color="blue") +
  geom_point(aes(y=Cases),shape=21,size=0.5) +
  geom_line(size=0.25,aes(y=Adjusted),color="black") +
  geom_line(data=out %>% filter(Date<=max_date),
              size=1,aes(y=value),colour="blue") +
  scale_fill_manual(values=variant_colours,guide=FALSE) +
  scale_y_continuous(labels=scales::comma)+#,trans="log") +
  #geom_line(data=out, aes(y=value),color="brown") +
  #coord_cartesian(ylim=c(0,2000)) +
  theme_bw() +
  geom_text(data=NULL,x=as.Date("2021-01-15"),y=170,label="Regular COVID") +
  geom_text(data=NULL,x=as.Date("2021-04-10"),y=430,label="Variants\nof concern") +
  theme(legend.position="bottom") +
  #geom_line(aes(y=Trend),size=0.25) +
  scale_x_date(breaks="month",date_labels = "%b") +
  labs(title="BC case history and projections",
       x=NULL,fill=NULL,y="Number of daily cases",
       caption="MountainMath, Historical case data: BCCDC") +
  coord_cartesian(ylim = c(0,2000))
g
```




```
#ggsave("~/Desktop/dbh.png",g+snark,width=5,height=5)
ggsave(paste0("~/Desktop/bc_outlook_",Sys.Date(),".png"),g,width=5,height=5)
```

Sanity check on model convergance

```
out <- ode_for_parameters(fit$par,times=seq(0,max(truth$time)+130)) %>%
  rescale_ode_results()
out.start <- ode_for_parameters(parameters.start) %>%
  rescale_ode_results()
out.manual <- ode_for_parameters(c(beta_v0=0.22,beta_d0=0,beta_v1=0.165,beta_d1=45,beta_v2=0.176,beta_d1
                                   jerk0_v0=0.00003,jerk0_d0=130,initial_cases=2.650634e-05,initial_voc
                                 times=seq(0,max(truth$time)+130)) %>%
  rescale_ode_results()

bc_cases_raw %>%
  filter(Date>=model_start_date) %>%
  ggplot(aes(x=Date,y=Adjusted)) +
  geom_point(aes(y=Cases),shape=21,size=0.5) +
  geom_line() +
  geom_line(data=out, aes(y=value),color="brown") +
  geom_line(data=out.start, aes(y=value),color="steelblue") +
  geom_line(data=out.manual, aes(y=value),color="purple") +
  scale_x_date(breaks="month",date_labels = "%b")
```