

Appendix

```
library(epidemia)
plot.cases <- function(method, data) {
  # reproduction number
  rt <- epirt(
    formula = R(Nation, date) ~ 1 + Protest +
      schools_universities + Lockdown,
    prior = shifted_gamma(
      shape=2, scale = 1/3, shift = log(1.05)/6
    ),
    prior_covariance = decov(
      shape = c(2, rep(0.5, 5)),scale=0.25
    ),
    prior_intercept = rstanarm::normal(0.5,1),
    link = scaled_logit(5.7)
  )

  # the infection model
  inf <- epiinf(
    gen = EuropeCovid2$si,
    seed_days = 5
  )
  if (method == "deaths") {
    observation <- epiobs(
      formula = deaths ~ 1,
      i2o = EuropeCovid2$inf2death,
      prior_intercept = normal(0, 1),
      link = scaled_logit(0.02)
    )
  } else if (method == "Beds") {
    observation <- epiobs(
      formula = Beds ~ 1 + inhospital,
      i2o = dlnorm(1:20, 1.921, 0.428),

      link = scaled_logit(0.276),
      prior_intercept = normal(0, 0.5),
      center = TRUE
    )
  } else if (method == "deaths_inacurate") {
    observation <- epiobs(
      formula = deaths ~ 1+inhospital,
      i2o = EuropeCovid2$inf2death,
      prior_intercept = normal(0, 0.3),
      link = scaled_logit(0.02)
    )
    method <- "deaths"
  }
}
```

```

args <- list(
  rt=rt, inf=inf, obs=observation,
  data=data, seed=12345, refresh=0
)
options(mc.cores = parallel::detectCores())
pr_args <- c(
  args,
  list(
    algorithm="sampling", iter=1e4,
    prior_PD=TRUE, control = list(adapt_delta = 0.99)
  )
)

# prior R_t
fm_prior <- do.call(epim, pr_args)
p1 <- plot_rt(fm_prior, levels = c(30, 60, 95))
args$algorithm <- "fullrank"
args$iter <- 50000
args$tol_rel_obj <- 1e-8
fm <- do.call(epim, args)
p2 <- plot_rt(
  fm, step = T, levels = c(30, 60, 95)
)
p3 <- plot_obs(
  fm, type = method, step = T, levels = c(30, 60, 95)
)
p4 <- plot_infections(
  fm, step = T, levels = c(30, 60, 95)
)
p5 <- plot_infections(
  fm, cumulative = TRUE, step = T, levels = c(30, 60, 95)
)
return(list(prior = p1, rt = p2, obs = p3, infection = p4, cum = p5))
}

```