Detailed_prior_sim_methods

2023-05-05

Detailed methods - compiling the realised trends

The USGS and Audubon publish trend estimates for the long-term, starting in the late 1960s. We will use these long-term trend estimates for the regions and time-periods that they are available.

```
bbs_trends <- read.csv("data/BBS_1966-2019_core_best_trend.csv") %>%
  filter(Region == "SU1" |
           !grepl(x = Region, pattern = "[[:digit:]]")) %>% # state level names do not have numbers, al
  mutate(Region = ifelse(Region == "SU1","Survey_Wide",Region),
         Survey = "BBS",
         AOU = as.character(AOU),
         se_trend = (X97.5.CI - X2.5.CI)/4) %>%
  rename(trend = Trend)


cbc_trends <- read.csv("data/cbc_trends_abundance_indices_and_scaling_factors_v4.0_web_download_12Apr20
  filter(grepl(pattern = "RatioTrendAllYears",parameter),
         (stratum == "USACAN" |
            nchar(stratum) == 2)) %>%
  select(ebird_com_name,stratum,estimate_mean,estimate_ucl,estimate_lcl,parameter) %>%
  rename(AOU = ebird_com_name,
         Region = stratum,
         trend = estimate_mean) %>%
  mutate(Region = ifelse(Region == "USACAN","Survey_Wide",Region),
         Survey = "CBC",
         se_trend = (estimate_ucl-estimate_lcl/4))



save(list = c("cbc_trends","bbs_trends"),
     file = "data/CBC_BBS_published_trends.RData")
```

Supplementing the published trends with comparisons between years of population trajectories

To supplement these published trend estimates, we generated additional estimates of trends and variation in trends using the published population trajectories (collection of annual indices of relative abundance). These additional estimates of trend are a post-hoc summary of the population trajectories that will not include any assessment of uncertainty. For our purposes, the uncertainty of each additional trend estimate is much less important than the full collection of estimates across all time-periods, regions, and species. Because we have ignored the uncertainty of the trajectories in our calculations of trends, we have also removed some of the most poorly estimated annual indices of abundance from each data collection. We dropped all estimates of annual abundance that had a coefficient of variation > 100 (SD/mean > 100).

Estimates from the program area available at a number of different spatial scales, but for this analysis, we have selected the survey-wide estimates to represent the broadest species-level estimates, and the state/province estimates to represent the spatial variation within species because of their consistent treatment in the two programs.

```
## downloaded from https://www.mbr-pwrc.usgs.gov/ on June 13 2022
bbs_inds <- read.csv("data/Index_best_1966-2019_core_best.csv",
                             colClasses = c("integer",
                                            "character",
                                            "integer",
                                            "numeric",
                                            "numeric",
                                            "numeric")) %>%
rename(lci = X2.5..CI,
       uci = X97.5..CI) %>%
  mutate(cv = ((uci-lci)/4)/Index ) %>%
  select(AOU,Region,Year,Index,cv) %>%
  filter(Region == "SU1" |
           !grepl(x = Region, pattern = "[[:digit:]]")) %>% # state level names do not have numbers, al
  filter(cv < 100,
         !is.na(cv)) %>%  # drop extremely poorly estimated values
  mutate(Region = ifelse(Region == "SU1","Survey_Wide",Region),
         Survey = "BBS",
         AOU = as.character(AOU))


# provided by Audubon June 13 2022.
cbc_file <- "data/cbc_trends_abundance_indices_and_scaling_factors_v4.0_web_download_12Apr2022.csv"
cbc_inds <- read.csv(cbc_file) %>%
  filter(parameter == "AbundanceIndex",
         (stratum == "USACAN" |
            nchar(stratum) == 2)) %>% # selects the state/prov two-letter names
  mutate(cv = ((estimate_ucl-estimate_lcl)/4)/estimate_median ) %>%
select(ebird_com_name,stratum,count_year,estimate_median,cv) %>%
rename(AOU = ebird_com_name,
       Region = stratum,
       Year = count_year,
       Index = estimate_median) %>%
  filter(cv < 100) %>%  # drop extremely poorly estimated values
  mutate(Region = ifelse(Region == "USACAN","Survey_Wide",Region),
         Survey = "CBC",
         Year = as.integer(Year))

all_inds <- bind_rows(bbs_inds,cbc_inds)

save(list = "all_inds",
     file = "data/all_state_survey_wide_indices_BBS_CBC.RData")
```

Using the trajectories, we calculate all possible 1-year, 2-year, 5-year, 10-year, 20-year trends, and 50-year trends: the point estimates based on the comparison of posterior medians of annual indices. These trends are similar to the end-point trends used in the hierarchical models for the BBS and CBS.

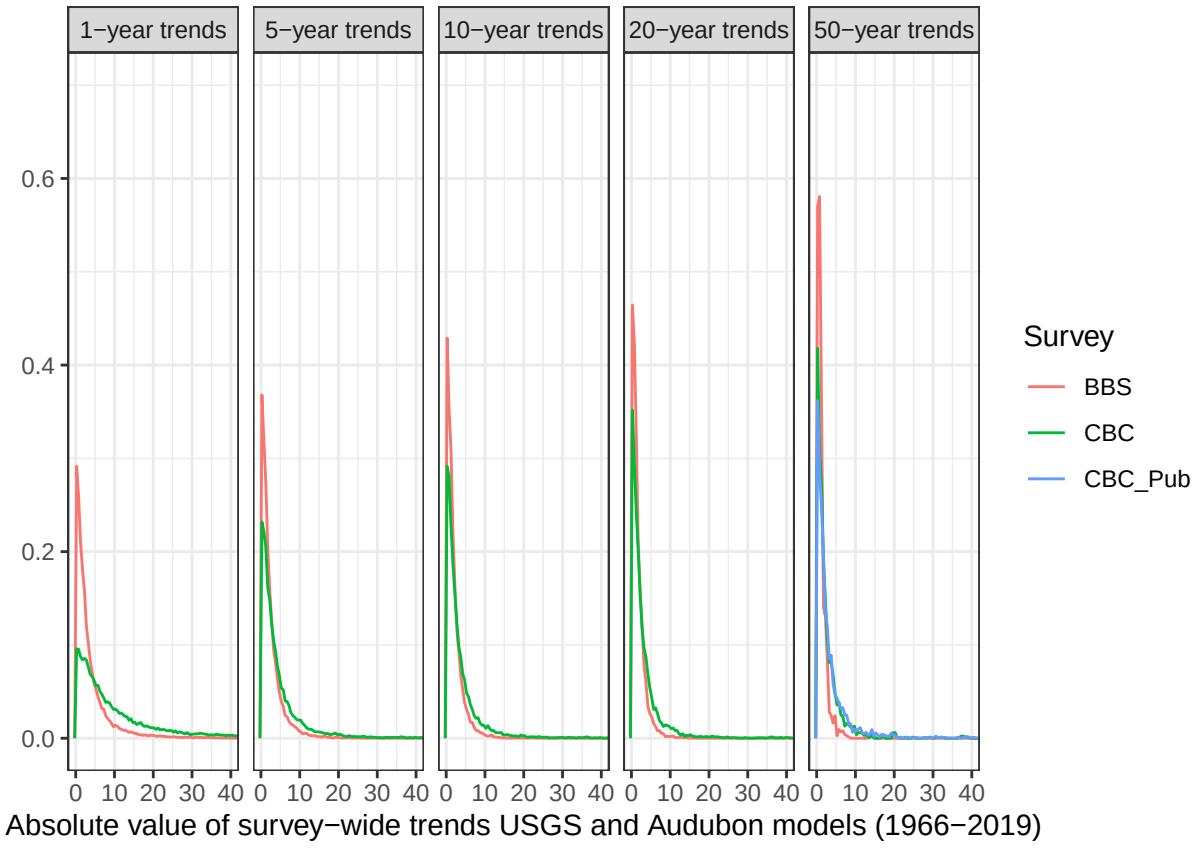
```r
load("data/all_state_survey_wide_indices_BBS_CBC.RData")

# function to calculate a %/year trend from a count-scale trajectory
trs <- function(y1,y2,ny){
  tt <- (((y2/y1)^(1/ny))-1)*100
}

miny = min(all_inds$Year)
maxy = max(all_inds$Year)
all_trends <- NULL

for(tl in c(2,6,11,21,51)){
  #estimating all possible 1-year, 2-year, 5-year, 10-year, 20-year,
  # and 50-year trends, with no uncertainty, just the point estimates
  #based on the comparison of posterior means fo annual indices
  ny = tl-1
  yrs1 <- seq(miny,(maxy-ny),by = 1)
  yrs2 <- yrs1+ny
  for(j in 1:length(yrs1)){
    y2 <- yrs2[j]
    y1 <- yrs1[j]

    nyh2 <- paste0("Y",y2)
    nyh1 <- paste0("Y",y1)

    tmp <- all_inds %>%
      filter(Year %in% c(y1,y2)) %>%
      select(AOU,Index,Year,Region,Survey) %>%
      pivot_wider(.,names_from = Year,
                  values_from = Index,
                  names_prefix = "Y") %>%
      rename_with(.,~gsub(pattern = nyh2,replacement = "YE", .x)) %>%
      rename_with(.,~gsub(pattern = nyh1,replacement = "YS", .x)) %>%
      drop_na() %>%
      group_by(AOU,Region,Survey) %>%
      summarise(trend = trs(YS,YE,ny),
                .groups = "keep")%>%
      mutate(first_year = y1,
             last_year = y2,
             nyears = ny,
             abs_trend = abs(trend),
             t_years = paste0(ny,"-year trends"))

    all_trends <- bind_rows(all_trends,tmp)
  }
}



  load("data/CBC_BBS_published_trends.RData")
```

```
  bbs_trends <- bbs_trends %>%
    rename(trend = Trend) %>%
    mutate(abs_trend = abs(trend),
           t_years = "50-year trends",
           Survey = "BBS_Pub",
           AOU = as.character(AOU)) %>%
    select(abs_trend,trend,Region,AOU,Survey,t_years)
  cbc_trends <- cbc_trends %>%
    rename(trend = Trend) %>%
    mutate(abs_trend = abs(trend),
           t_years = "50-year trends",
           Survey = "CBC_Pub") %>%
    select(abs_trend,trend,Region,AOU,Survey,t_years)

  all_trends <- all_trends %>%
    bind_rows(.,bbs_trends) %>%
    bind_rows(.,cbc_trends) %>%
  mutate(t_years = factor(t_years,
                          levels = c("1-year trends",
                                     "5-year trends",
                                     "10-year trends",
                                     "20-year trends",
                                     "50-year trends"),
                          ordered = TRUE))

  saveRDS(all_trends,file = "data/all_trends_bbs_cbc.rds")
```

Realised distribution of all trend estimates from CBC and BBS

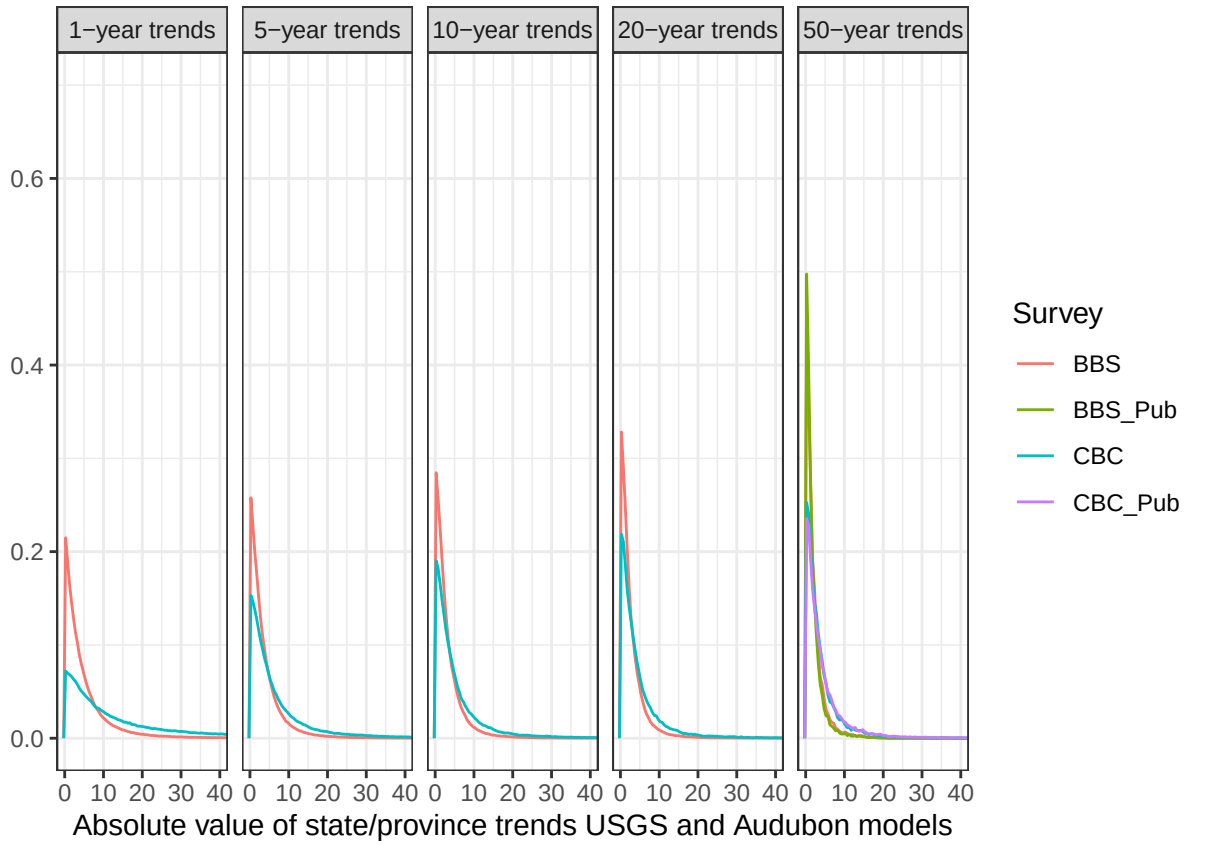
The distribution of survey wide trend estimates for CBC and BBS data.

```
all_trends <- readRDS("data/all_trends_bbs_cbc.rds")
all_continental_trends <- all_trends %>%
  filter(Region == "Survey_Wide")
mxabs = 2000#quantile(all_trends$abs_trend,0.9999) - facilitates setting breaks


realised_all_sw_freq <- ggplot(data = all_continental_trends,
                               aes(abs_trend,after_stat(density),
                                   group = Survey,
                                      colour = Survey))+
  geom_freqpoly(breaks = c(0,seq(0.5,mxabs,0.5)),center = 0)+
  xlab("Absolute value of survey-wide trends USGS and Audubon models (1966-2019)")+
  ylab("")+
  theme_bw()+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))+
  facet_wrap(vars(t_years),
             nrow = 1,
             ncol = 5)
print(realised_all_sw_freq)
```



The realised distribution of all state/provincial trend estimates for BBS and CBC.

```
#selecting the province/state level trends to assess spatial variation
all_politic_trends <- all_trends %>%
  filter(Region != "Survey_Wide")



realised_all_politic_freq <- ggplot(data = all_politic_trends,
                                    aes(abs_trend,after_stat(density),
                                        group = Survey,
                                           colour = Survey))+
  geom_freqpoly(breaks = c(0,seq(0.5,mxabs,0.5)),center = 0)+
  xlab("Absolute value of state/province trends USGS and Audubon models")+
  ylab("")+
  theme_bw()+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))+
  facet_wrap(vars(t_years),
             nrow = 1,
             ncol = 5)
print(realised_all_politic_freq)
```

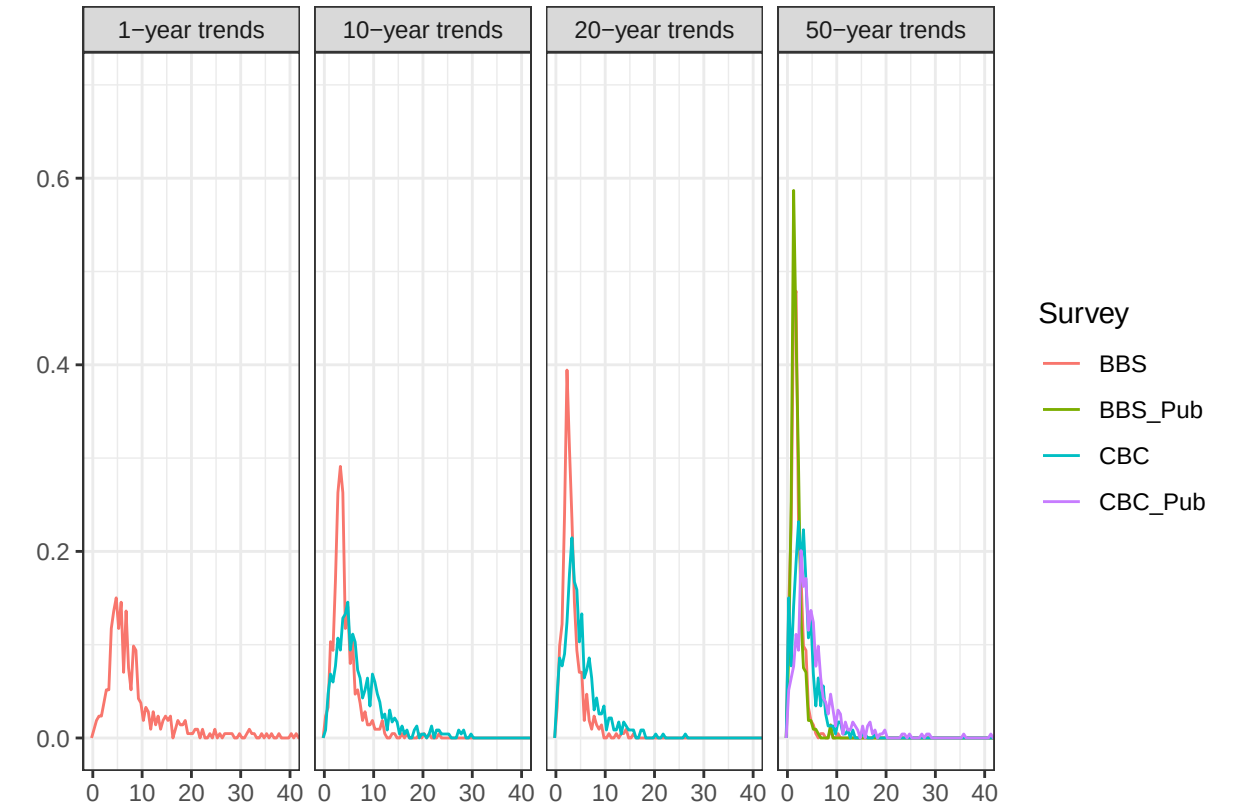


Examining the variation among regions within a given species

The published estimates of trends can also inform our prior expectations on the variation among regions in population trends at different temporal scales.

```
all_sd_trends <- all_politic_trends %>%
  filter(!(t_years == "1-year trends" & Survey == "CBC"),
         t_years != "5-year trends") %>%
  group_by(AOU,t_years,Survey) %>%
  summarise(sd_trends = sd(trend,na.rm = TRUE),
            min_trend = min(trend,na.rm = TRUE),
            max_trend = max(trend,na.rm = TRUE),
            q5_trend = quantile(trend,0.05,na.rm = TRUE),
            q95_trend = quantile(trend,0.95,na.rm = TRUE),
            .groups = "keep") %>%
  filter(is.finite(sd_trends))

realised_all_sd <- ggplot(data = all_sd_trends,
                                    aes(sd_trends,after_stat(density),
                                        group = Survey,
                                        colour = Survey))+
  geom_freqpoly(breaks = c(0,seq(0.5,mxabs,0.5)),center = 0)+
  xlab("SD (by species) of state/province trends USGS and Audubon models (1966-2019)")+
  ylab("")+
  theme_bw()+
```

```
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))+
  facet_wrap(vars(t_years),
             nrow = 1,
             ncol = 5)
print(realised_all_sd)
```



SD (by species) of state/province trends USGS and Audubon models (1966−2019)

Prior Simulation for the flexibility of the overall population trajectories

Setting up the basic data structure using Wood Thrush

Fitting the prior models

the following code fits a customized set of Stan models that estimate the population trajectories and trends based on the priors.

```
for(i in 1:nrow(tb_sims)){
  mod <- tb_sims[i,"model"]
  prior_B <- tb_sims[i,"prior_time"]
  prior_b <- tb_sims[i,"prior_sd_time"]
  prior_y <- tb_sims[i,"prior_yeareffects"]

  if(mod == "gamye"){
    stan_data <- data_gamye
```

```
  stan_data[["prior_scale_y"]] <- prior_y

}else{
  stan_data <- data_diff

}

stan_data[["prior_scale_b"]] <- prior_b

if(tb_sims[i,"hierarchical"]){
  hier <- ""
  stan_data[["prior_scale_B"]] <- prior_B

  if(tb_sims[i,"spatial"]){
    spat <- "spatial"

  }else{
  spat <- "non_spatial"
    stan_data[["N_edges"]] <- NULL
    stan_data[["node1"]] <- NULL
    stan_data[["node2"]] <- NULL

  }

}else{
  hier <- "non_hierarchical"
  spat <- ""
   stan_data[["N_edges"]] <- NULL
    stan_data[["node1"]] <- NULL
    stan_data[["node2"]] <- NULL

}


out_base <- paste("prior_sim",mod,spat,hier,sep = "_")


mod_file <- paste0("models/",tb_sims[i,"model_file"])

    # Fit model ----------------------------------------------------------------

    print(paste("beginning",out_base,Sys.time()))


    ## compile model
    model <- cmdstan_model(mod_file)



    stanfit <- model$sample(
      data=stan_data,
      refresh=100,
      chains=2, iter_sampling=1000,
```

```
        iter_warmup=500,
        parallel_chains = 2,
        #pars = parms,
        adapt_delta = 0.8,
        max_treedepth = 10,
        seed = 123)


      stanfit$save_object(file = paste0("output/",out_base,".rds"))







}
```

We then summarized the estimated trajectories as well as the 1, 5, 10, 20, and 50-year trends simulated from the alternative priors.

```
source("Functions/posterior_summary_functions.R")

n_out <- NULL
trends_out <- NULL
summ_out <- NULL




for(i in 1:nrow(tb_sims)){

  trends_out_tmp <- NULL

  mod <- tb_sims[i,"model"]
  prior_B <- tb_sims[i,"prior_time"]
  prior_b <- tb_sims[i,"prior_sd_time"]
  prior_y <- tb_sims[i,"prior_yeareffects"]


  if(tb_sims[i,"hierarchical"]){
    hier <- ""

    if(tb_sims[i,"spatial"]){
      spat <- "spatial"

    }else{
    spat <- "non_spatial"

    }

  }else{
    hier <- "non_hierarchical"
    spat <- ""
```

```
  }


  out_base <- paste("prior_sim",mod,spat,hier,sep = "_")

  stanfit <- readRDS(paste0("output/",out_base,".rds"))


summ = stanfit$summary()

summ <- summ %>%
  mutate(model = mod,
         spatial = spat,
         hierarchical = hier)

n_samples <- posterior_samples(stanfit,
                               parm = "n",
                               dims = c("strat","Year_Index"))


if(mod == "gamye"){
nsmooth_samples <- posterior_samples(stanfit,
                               parm = "nsmooth",
                               dims = c("strat","Year_Index"))



}




nyears = max(n_samples$Year_Index)
# function to calculate a %/year trend from a count-scale trajectory
trs <- function(y1,y2,ny){
  tt <- (((y2/y1)^(1/ny))-1)*100
}


for(tl in c(2,6,11,21,51)){ #estimating all possible 1-year, 10-year, and full trends
  ny = tl-1
  yrs1 <- seq(1,(nyears-ny),by = ny)
  yrs2 <- yrs1+ny
  for(j in 1:length(yrs1)){
    y2 <- yrs2[j]
    y1 <- yrs1[j]

nyh2 <- paste0("Y",y2)
nyh1 <- paste0("Y",y1)
trends <- n_samples %>%
  filter(Year_Index %in% c(y1,y2)) %>%
  select(.draw,.value,Year_Index,strat) %>%
```

```
  group_by(.draw,Year_Index) %>%
  summarise(.value = mean(.value),
            .groups = "keep") %>%
  pivot_wider(.,names_from = Year_Index,
              values_from = .value,
              names_prefix = "Y") %>%
  rename_with(.,~gsub(pattern = nyh2,replacement = "YE", .x)) %>%
  rename_with(.,~gsub(pattern = nyh1,replacement = "YS", .x)) %>%
  group_by(.draw) %>%
  summarise(trend = trs(YS,YE,ny),
            .groups = "keep")%>%
  mutate(model = mod,
         spatial = spat,
         hierarchical = hier,
         first_year = y1,
         last_year = y2,
         nyears = ny,
         scale = "Survey_Wide",
         type = "full")



trends_out_tmp <- bind_rows(trends_out_tmp,trends)

trends <- n_samples %>%
  filter(Year_Index %in% c(y1,y2)) %>%
  select(.draw,.value,Year_Index,strat) %>%
  pivot_wider(.,names_from = Year_Index,
              values_from = .value,
              names_prefix = "Y") %>%
  rename_with(.,~gsub(pattern = nyh2,replacement = "YE", .x)) %>%
  rename_with(.,~gsub(pattern = nyh1,replacement = "YS", .x)) %>%
  group_by(.draw,strat,
            .groups = "keep") %>%
  summarise(trend = trs(YS,YE,ny),
            .groups = "keep")%>%
  mutate(model = mod,
         spatial = spat,
         hierarchical = hier,
         first_year = y1,
         last_year = y2,
         nyears = ny,
         scale = "Regional",
         type = "full")

trends_out_tmp <- bind_rows(trends_out_tmp,trends)


if(mod == "gamye"){
  trends <- nsmooth_samples %>%
  filter(Year_Index %in% c(y1,y2)) %>%
  select(.draw,.value,Year_Index,strat) %>%
  group_by(.draw,Year_Index) %>%
```

```
  summarise(.value = mean(.value),
            .groups = "keep") %>%
  pivot_wider(.,names_from = Year_Index,
              values_from = .value,
              names_prefix = "Y") %>%
  rename_with(.,~gsub(pattern = nyh2,replacement = "YE", .x)) %>%
  rename_with(.,~gsub(pattern = nyh1,replacement = "YS", .x)) %>%
  group_by(.draw) %>%
  summarise(trend = trs(YS,YE,ny),
            .groups = "keep")%>%
  mutate(model = mod,
         spatial = spat,
         hierarchical = hier,
         first_year = y1,
         last_year = y2,
         nyears = ny,
         scale = "Survey_Wide",
         type = "smooth")



trends_out_tmp <- bind_rows(trends_out_tmp,trends)

trends <- nsmooth_samples %>%
  filter(Year_Index %in% c(y1,y2)) %>%
  select(.draw,.value,Year_Index,strat) %>%
  pivot_wider(.,names_from = Year_Index,
              values_from = .value,
              names_prefix = "Y") %>%
  rename_with(.,~gsub(pattern = nyh2,replacement = "YE", .x)) %>%
  rename_with(.,~gsub(pattern = nyh1,replacement = "YS", .x)) %>%
  group_by(.draw,strat) %>%
  summarise(trend = trs(YS,YE,ny),
            .groups = "keep")%>%
  mutate(model = mod,
         spatial = spat,
         hierarchical = hier,
         first_year = y1,
         last_year = y2,
         nyears = ny,
         scale = "Regional",
         type = "smooth")
trends_out_tmp <- bind_rows(trends_out_tmp,trends)


}

}
}
save(file = paste0("output/prior_sim_summary",out_base,".RData"),
     list = c("trends_out_tmp",
              "summ"))
```

```
summ_out <- bind_rows(summ_out,summ)
trends_out <- bind_rows(trends_out,trends_out_tmp)
  }#prior_scale

saveRDS(trends_out,file = "output/prior_sim_trends.rds")
saveRDS(summ_out,file = "output/prior_sim_summaries.rds")
```

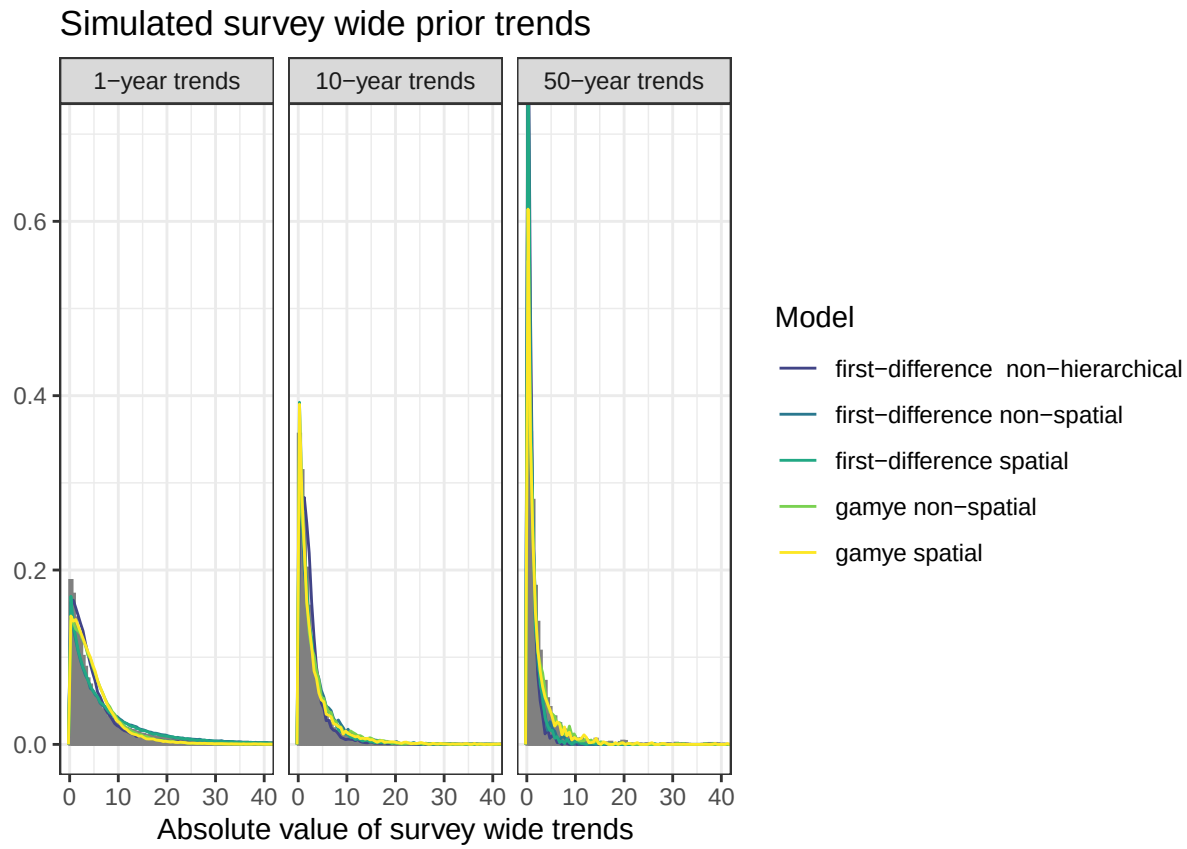
```
# loading the stored realised BBS and CBC trend estimates
 all_trends <- readRDS(file = "data/all_trends_bbs_cbc.rds")

all_sw_trends <- all_trends %>%
  filter(Region == "Survey_Wide")
all_politic_trends <- all_trends %>%
  filter(Region != "Survey_Wide")

mxabs = 2000#upper limit on the absolute trend estimates for the density plots below

## loading the prior simulated trends for all models
trends_out <- readRDS("output/prior_sim_trends.rds")


#summarising the trends for the political regions (not survey wide)
prior_trends_politic <- trends_out %>%
  filter(scale != "Survey_Wide") %>%
  mutate(abs_trend = abs(trend),
         t_years = paste(nyears,"year trends",sep = "-"),
         t_years = factor(t_years,
                          levels = c("1-year trends",
                                     "5-year trends",
                                     "10-year trends",
                                     "20-year trends",
                                     "50-year trends"),
                          ordered = TRUE),
         model_type = paste(model,spatial,hierarchical,sep = " "))

#setting up plotting label names for the models
mod_types <- unique(prior_trends_politic$model_type)
names(mod_types) <- gsub(mod_types,pattern = "(_)|[[:space:]]{2}",
                         replacement = " ")
realised_all_politic_freq <- vector(mode = "list",length = length(mod_types))
names(realised_all_politic_freq) <- mod_types

#looping through each model to generate the realised vs prior trend density plots
for(i in 1:length(mod_types)){
  mm = mod_types[i]
  mlab = names(mod_types)[i]

     tmp_sim <- prior_trends_politic %>%
    filter(model_type == mm)

tmp <- ggplot(data = all_politic_trends,
                                  aes(abs_trend,after_stat(density),
                                      groups = Survey))+
```

```
  geom_freqpoly(breaks = c(0,seq(0.5,mxabs,0.5)),center = 0,
                colour = grey(0.5))+
  geom_freqpoly(data = tmp_sim,
                aes(abs_trend,after_stat(density),
                colour = model_type),
                inherit.aes = FALSE,
                breaks = c(0,seq(0.5,mxabs,0.5)),center = 0)+
  scale_colour_viridis_d(begin = 0.8)+
  xlab("Absolute value of state/province trends USGS and Audubon models")+
  ylab("")+
  labs(title = paste0("Simulated state/province prior trends from ",mlab,"model"))+
  theme_bw()+
  theme(legend.position = "none")+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))

  if(grepl(pattern = "gam",mm)){ # gamye models need to plots to represent the full and smooth only tre
   tmp <- tmp+facet_wrap(vars(type,t_years),
              nrow = 2,ncol = 5)
}else{
 tmp <- tmp+facet_wrap(vars(t_years),
              ncol = 5)
}

realised_all_politic_freq[[mm]] <- tmp

}


# same plots and trend summaries as above, but for the survey-wide estimates
prior_trends_sw <- trends_out %>%
  filter(scale == "Survey_Wide") %>%
  mutate(abs_trend = abs(trend),
         t_years = paste(nyears,"year trends",sep = "-"),
         t_years = factor(t_years,
                          levels = c("1-year trends",
                                     "5-year trends",
                                     "10-year trends",
                                     "20-year trends",
                                     "50-year trends"),
                          ordered = TRUE),
         model_type = paste(model,spatial,hierarchical,sep = " "))

realised_all_sw_freq <- vector(mode = "list",length = length(mod_types))
names(realised_all_sw_freq) <- mod_types

# looping through models to generate plots
for(i in 1:length(mod_types)){
  mm = mod_types[i]
  mlab = names(mod_types)[i]
  tmp_sim <- prior_trends_sw %>%
    filter(model_type == mm)
tmp <- ggplot(data = all_sw_trends,
```

```
                                aes(abs_trend,after_stat(density),
                                    groups = Survey))+
  geom_freqpoly(breaks = c(0,seq(0.5,mxabs,0.5)),center = 0,
                colour = grey(0.5))+
  geom_freqpoly(data = tmp_sim,
                aes(abs_trend,after_stat(density),
                colour = model_type),
                inherit.aes = FALSE,
                breaks = c(0,seq(0.5,mxabs,0.5)),center = 0)+
  scale_colour_viridis_d(begin = 0.8)+
  xlab("Absolute value of survey wide trends USGS and Audubon models")+
  ylab("")+
  labs(title = paste0("Simulated survey wide prior trends from ",mlab,"model"))+
  theme_bw()+
  theme(legend.position = "none")+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))

  if(grepl(pattern = "gam",mm)){
   tmp <- tmp+facet_wrap(vars(type,t_years),
               nrow = 2,ncol = 5)
 }else{
  tmp <- tmp+facet_wrap(vars(t_years),
               ncol = 5)
}

realised_all_sw_freq[[mm]] <- tmp

}

saveRDS(realised_all_sw_freq,
        "output/realised_all_sw_freq.rds")

saveRDS(realised_all_politic_freq,
        "output/realised_all_politic_freq.rds")
```

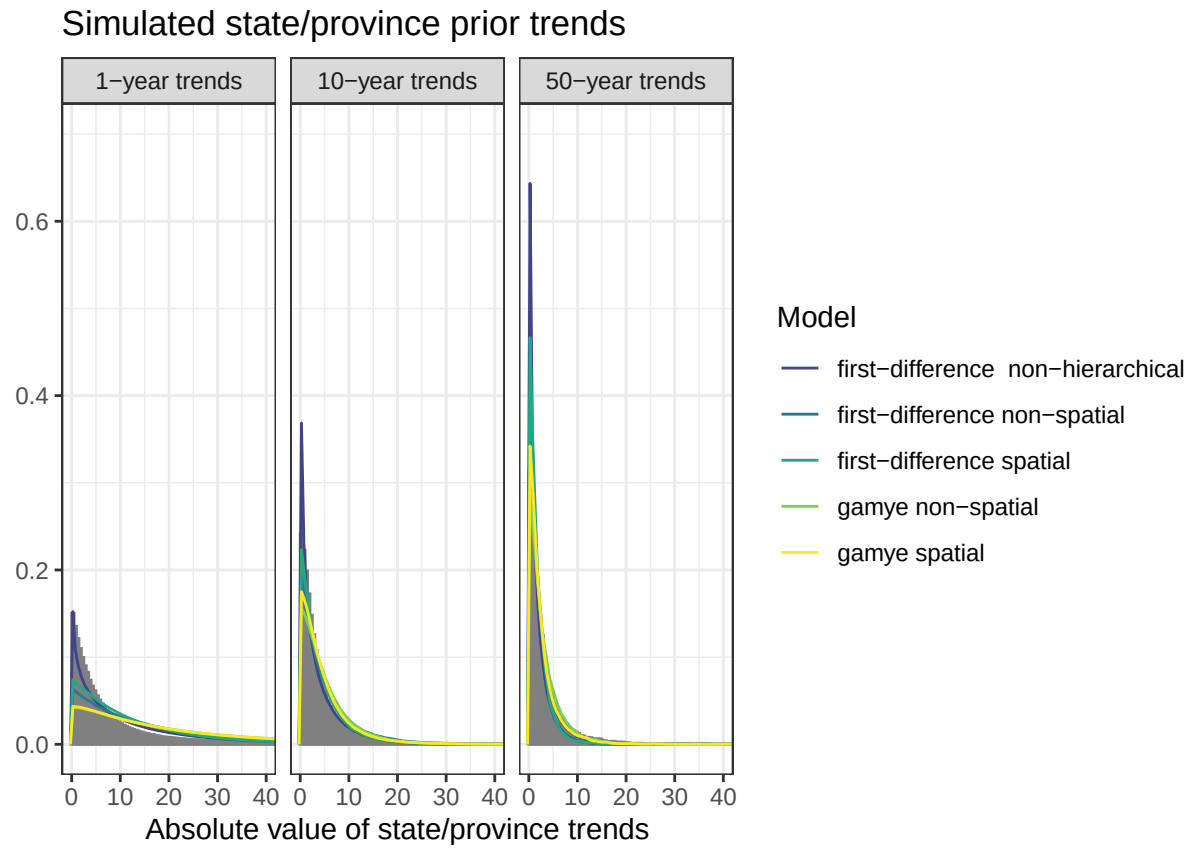
The code above plots the realised distribution of trends across the full time series of the BBS and CBC (black lines) along with the distribution of trends using the various priors used in the models in this paper.

```
  tmp_sim <- prior_trends_sw %>%
    filter(t_years %in% c("1-year trends",
                          "10-year trends",
                          "50-year trends"),
           type == "full") %>%
  mutate(Model = gsub(model_type,
                      pattern = "_",
                      replacement = "-"))




sub_sw_trends <- all_sw_trends %>%
    filter(t_years %in% c("1-year trends",
                          "10-year trends",
                          "50-year trends"))
sw_overview <- ggplot(data = sub_sw_trends,
                                aes(abs_trend,after_stat(density)))+
    geom_histogram(breaks = c(0,seq(0.5,mxabs,0.5)),
                colour = grey(0.5))+
  geom_freqpoly(data = tmp_sim,
                aes(abs_trend,after_stat(density),
                colour = Model),
                inherit.aes = FALSE,
                breaks = c(0,seq(0.5,mxabs,0.5)),center = 0)+
  scale_colour_viridis_d(begin = 0.2)+
  xlab("Absolute value of survey wide trends")+
  ylab("")+
  labs(title = paste0("Simulated survey wide prior trends"))+
  theme_bw()+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))+
  facet_wrap(vars(t_years),
             ncol = 5)

sw_overview
```




```
saveRDS(sw_overview,
        "output/survey_wide_overview.rds")


  tmp_sim <- prior_trends_politic%>%
    filter(t_years %in% c("1-year trends",
                          "10-year trends",
                          "50-year trends"),
           type == "full") %>%
  mutate(Model = gsub(model_type,
                      pattern = "_",
                      replacement = "-"))



sub_politic_trends <- all_politic_trends %>%
    filter(t_years %in% c("1-year trends",
                          "10-year trends",
                          "50-year trends"))

geo_overview <- ggplot(data = sub_politic_trends,
                                  aes(abs_trend,after_stat(density)))+
    geom_histogram(breaks = c(0,seq(0.5,mxabs,0.5)),
                colour = grey(0.5))+
  geom_freqpoly(data = tmp_sim,
                aes(abs_trend,after_stat(density),
```

```
                colour = Model),
                inherit.aes = FALSE,
                breaks = c(0,seq(0.5,mxabs,0.5)),center = 0)+
  scale_colour_viridis_d(begin = 0.2)+
  xlab("Absolute value of state/province trends")+
  ylab("")+
  labs(title = paste0("Simulated state/province prior trends"))+
  theme_bw()+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))+
  facet_wrap(vars(t_years),
             ncol = 5)

geo_overview
```




```
saveRDS(geo_overview,
        "output/state_province_overview.rds")
```

Exploring the standard deviation of trends among regions

```
# loading the stored realised BBS and CBC trend estimates
 all_trends <- readRDS(file = "data/all_trends_bbs_cbc.rds")
```

```r
all_sw_trends <- all_trends %>%
  filter(Region == "Survey_Wide")
all_politic_trends <- all_trends %>%
  filter(Region != "Survey_Wide")

realised_all_sd_freq <- vector(mode = "list",length = length(mod_types))
names(realised_all_sd_freq) <- mod_types

#function to calculate the inter-quartile interval
iq_func <- function(x,q = 0.5){
  q1 <- (1-q)/2
  q2 <- 1-q1
  iq <- quantile(x,q2) - quantile(x,q1)
  return(iq)
}

## summarizing the sd of realised trends
all_politic_sdtrends <- all_politic_trends %>%
  filter(Survey %in% c("BBS","CBC")) %>%
  group_by(Survey,AOU,first_year,last_year,nyears) %>%
  summarise(sd_trend = sd(trend),
            iq_range = iq_func(trend)) %>%
  mutate(t_years = paste(nyears,"year trends",sep = "-"),
         t_years = factor(t_years,
                          levels = c("1-year trends",
                                     "5-year trends",
                                     "10-year trends",
                                     "20-year trends",
                                     "50-year trends"),
                          ordered = TRUE))
```

```
## `summarise()` has grouped output by 'Survey', 'AOU', 'first_year', 'last_year'.
## You can override using the `.groups` argument.
```

```r
BBS_politic_sdtrends <- all_politic_trends %>%
  filter(Survey %in% c("BBS")) %>%
  group_by(Survey,AOU,first_year,last_year,nyears) %>%
  summarise(sd_trend = sd(trend),
            iq_range = iq_func(trend)) %>%
  mutate(t_years = paste(nyears,"year trends",sep = "-"),
         t_years = factor(t_years,
                          levels = c("1-year trends",
                                     "5-year trends",
                                     "10-year trends",
                                     "20-year trends",
                                     "50-year trends"),
                          ordered = TRUE))
```

```
## `summarise()` has grouped output by 'Survey', 'AOU', 'first_year', 'last_year'.
## You can override using the `.groups` argument.
```

```r
## loading the prior simulated trends for all models
trends_out <- readRDS("output/prior_sim_trends.rds")

#summarising the trends for the political regions (not survey wide)
prior_sdtrends_politic <- trends_out %>%
  filter(scale != "Survey_Wide") %>%
  group_by(.draw,model,spatial,hierarchical,first_year,last_year,nyears,type) %>%
  summarise(sd_trend = sd(trend),
            iq_range = iq_func(trend)) %>%
  mutate(t_years = paste(nyears,"year trends",sep = "-"),
         t_years = factor(t_years,
                          levels = c("1-year trends",
                                     "5-year trends",
                                     "10-year trends",
                                     "20-year trends",
                                     "50-year trends"),
                          ordered = TRUE),
         model_type = paste(model,spatial,hierarchical,sep = " "))
```

```
## `summarise()` has grouped output by '.draw', 'model', 'spatial',
## 'hierarchical', 'first_year', 'last_year', 'nyears'. You can override using the
## `.groups` argument.
```

```r
# looping through models to generate plots
for(i in 1:length(mod_types)){
  mm = mod_types[i]
  mlab = names(mod_types)[i]
  tmp_sim <- prior_sdtrends_politic %>%
    filter(model_type == mm)
tmp <- ggplot(data = all_politic_sdtrends,
                                 aes(sd_trend,after_stat(density),
                                     groups = Survey))+
  geom_freqpoly(breaks = c(seq(-mxabs,mxabs,0.5)),center = 0,
                colour = grey(0.5))+
  geom_freqpoly(data = tmp_sim,
    aes(sd_trend,after_stat(density),
    colour = model_type),
    inherit.aes = FALSE,
    breaks = c(seq(-mxabs,mxabs,0.5)),center = 0)+
  scale_colour_viridis_d(begin = 0.8)+
  xlab("SD of survey wide trends USGS and Audubon models")+
  ylab("")+
  labs(title = paste0("SD of simulated survey wide prior trends from ",mlab,"model"))+
  theme_bw()+
  theme(legend.position = "none")+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))

  if(grepl(pattern = "gam",mm)){
   tmp <- tmp+facet_wrap(vars(type,t_years),
              nrow = 2,ncol = 5)
 }else{
  tmp <- tmp+facet_wrap(vars(t_years),
```

```
            ncol = 5)
}

realised_all_sd_freq[[mm]] <- tmp

}

saveRDS(realised_all_sd_freq,
        "output/realised_all_sd_freq.rds")



## overview plot

 tmp_sim <- prior_sdtrends_politic %>%
    filter(((type == "full" & model == "first_difference")|
              (type == "full" & model == "gamye")),
           t_years %in% c("1-year trends",
                          "10-year trends",
                          "50-year trends")) %>%
   mutate(Model = gsub("_","-",model_type))

 sub_politic_sdtrends <- all_politic_sdtrends %>%
      filter(t_years %in% c("1-year trends",
                          "10-year trends",
                          "50-year trends"))

sd_overview <- ggplot(data = sub_politic_sdtrends,
                                  aes(sd_trend,after_stat(density)))+
  geom_histogram(breaks = c(seq(0,mxabs,0.5)),
                 colour = grey(0.5))+
  geom_freqpoly(data = tmp_sim,
    aes(sd_trend,after_stat(density),
    colour = Model),
    inherit.aes = FALSE,
    breaks = c(seq(0,mxabs,0.5)),center = 0)+
  scale_colour_viridis_d(begin = 0.2)+
  xlab("SD of survey wide trends")+
  ylab("")+
  labs(title = paste0("SD of simulated prior trends among regions"))+
  theme_bw()+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))+
  facet_wrap(vars(t_years),
             ncol = 5)

saveRDS(sd_overview,"output/sd_overview.rds")

 sub_politic_sdtrends <- BBS_politic_sdtrends %>%
      filter(t_years %in% c("1-year trends",
                          "10-year trends",
                          "50-year trends"))
```

```
sd_overview_bbs <- ggplot(data = sub_politic_sdtrends,
                                  aes(sd_trend,after_stat(density)))+
  geom_histogram(breaks = c(seq(0,mxabs,0.5)),
                 colour = grey(0.5))+
  geom_freqpoly(data = tmp_sim,
    aes(sd_trend,after_stat(density),
    colour = Model),
    inherit.aes = FALSE,
    breaks = c(seq(0,mxabs,0.5)),center = 0)+
  scale_colour_viridis_d(begin = 0.2)+
  xlab("SD of survey wide trends")+
  ylab("")+
  labs(title = paste0("SD of simulated prior trends among regions"))+
  theme_bw()+
  coord_cartesian(ylim = c(0,0.7),
                  xlim = c(0,40))+
  facet_wrap(vars(t_years),
             ncol = 5)

saveRDS(sd_overview_bbs,"output/sd_overview_bbs.rds")
```