**Appendix S1 – Example of a single simulation realization, and summary of 1000 repeated simulations**

Here, we illustrate a single realization of a simulation that was used to test our statistical model. As described in the main text of the manuscript, we simulated trajectories at each of 15 colonies as a combination of two temporally autocorrelated environmental processes: 1) an “unshared” environmental covariate that varied independently among colonies, and 2) a “shared” environmental covariate that affected all colonies simultaneously, imposing correlation among colony trajectories. In figure S1.1, we illustrate a single realization of a simulation in which colonies were affected by both environmental covariates; thus, colonies partially share a temporal trajectory. We randomly assigned each colony to be surveyed in 2-6 randomly selected years across the 50-year period, with at one survey occurring in the first and last 10 years of the simulation. Observation error was simulated according to equation 2 of the manuscript, with parameters chosen from the posterior mean from the empirical analysis for storm-petrels, where Counts were simulated as Poisson random variables using equation 1. The magnitude of observation error was then retained for 65% of surveys, to replicate the empirical data where estimates of error were not available for all surveys.

A screenshot of a graph

Description automatically generated

**Figure S1.1.** Example of simulated dynamics and observed counts at each of 15 colonies. In this example, colonies shared a time-varying environmental driver, resulting in partially shared temporal colony trajectories. Black dots indicate survey counts, and error bars indicate counts for which we assumed the magnitude of survey error was known. Open circles around dots indicate that survey error was unknown, and therefore would be estimated as part of the model-fitting process.

We fit the statistical model described in the manuscript to these simulated data, allowing us to estimate the temporal trajectories at each colony (shown in Figure S1.2).

A graph of a number of numbers and a line graph

Description automatically generated with medium confidence

**Figure S1.2.** Example of estimated temporal trajectories at each of the 15 simulated colonies. Blue lines indicate posterior median estimate, and blue ribbons represent 95% equal-tailed credible intervals around the prediction. Red lines represent the true colony-level population trajectories. Dots indicate survey counts and are interpreted as in Figure S1.1.

We used estimates of colony-level annual abundance (Figure S1.2) to calculate annual estimates of the regional population total. The trend in the estimated regional population was calculated using equation 7 in the manuscript. Figure S1.3 compares the estimated regional population trajectory (blue line) to the true population trajectory (red line), as well as a smooth fit to the true population trajectory (black line).

A graph with red and blue lines

Description automatically generated

**Figure S1.3.** Example of regional population trajectories at each of the 15 simulated colonies. Blue lines indicate posterior median estimate, and blue ribbons represent 95% equal-tailed credible intervals around the prediction. “True trend” is calculated as the geometric mean rate of change in the smoothed true trajectory (black line), while estimated trend is calculated as the geometric mean rate of change in the estimated trajectory (blue line).

We repeated this process 1000 times and compared true population trends to those estimated from our statistical model. We compared trend estimates calculated for the entire 50-year simulation (bottom row of Figure S1.4), as well as trend estimates for the final 10 years (top row of Figure S1.4). Results of these comparisons are illustrated in Figure S1.4.

A graph of different colored lines

Description automatically generated with medium confidence

**Figure S1.4.** Results of 1,000 simulated hypothetical scenarios of population change derived using a Bayesian GAMM approach based on 2-6 imprecise surveys from 15 colonies over a 50-year time period. Credible interval (CRI) coverage indicates whether the 95% credible interval of the estimated regional trend (vertical error bars) captured the “true” simulated regional trend for the most recent 10-year interval (top panels) or a full 50-year interval (bottom panels), under simulations where population dynamics at each colony were affected by an autocorrelated shared environmental covariate imposing correlations among the colony trajectories (left panels) or an unshared environmental covariate (right panels).

**Appendix S2 – Goodness-of-fit evaluation for statistical model**

We evaluated goodness-of-fit of our statistical model using several approaches. First, we visually compared predicted and observed counts for each colony, which are shown in Figures 2 and 3 in the main text of the manuscript. Second, we conducted posterior predictive checks, which evaluate whether the fitted statistical model can produce simulated datasets that “look like” the observed data. The degree to which a simulated dataset “looks like” the observed data is summarized using a discrepancy measure (see discussion in Kery and Royle 2016; pp 192-198). This approach entails generating an entirely new simulated dataset at each iteration of the MCMC fitting algorithm. Next, the simulated data is compared to the fitted values at each step of the MCMC algorithm using a discrepancy measure (in this study, we used root mean squared error; RMSE). The same discrepancy measure (RMSE) is also calculated between the observed data and the fitted values. The calculated discrepancy measures for the simulated datasets and observed dataset are then compared to evaluate whether the observed data has consistently higher or lower discrepancy with the fitted values than simulated datasets (which are, by definition, perfectly consistent with the fitted model).

Model mis-specification can be detected if the discrepancy between the observed data and the fitted model is consistently higher or lower than the discrepancy between the simulated data and the fitted model. The proportion of simulated datasets with higher discrepancy measures than the observed data are summarized using a “Bayesian p-value”, where a Bayesian p-value close to 0.5 implies the observed data are highly consistent with the fitted model, while p-values close to either 0 or 1 (and far from 0.50) indicate the observed data are not consistent with the fitted model.

Figures S2.1 and S2.2 illustrate the results of posterior predictive checks for storm petrels and puffins, respectively. The Bayesian p-value was 0.72 for the petrel model, and 0.43 for the puffin model. Both measures indicate that the observed data are consistent with fitted model.

A graph with a dotted line

Description automatically generated

**Figure S2.1**. Posterior predictive check for petrel model, comparing discrepancy measures (root mean squared error; RMSE) between the observed data and fitted model (x axis) to discrepancy measures between simulated data and fitted model (y axis). Correspondence along the 1:1 line indicates that simulated datasets have similar discrepancy to fitted model as the observed data, suggesting adequate model fit.

A graph with a dotted line

Description automatically generated

**Figure S2.2**. Posterior predictive check for puffin model, comparing discrepancy measures (root mean squared error; RMSE) between the observed data and fitted model (x axis) to discrepancy measures between simulated data and fitted model (y axis). Correspondence along the 1:1 line indicates that simulated datasets have similar discrepancy to fitted model as the observed data, suggesting adequate model fit.

**Appendix 3 – Results for Leach’s storm-petrel when omitting Machias Seal Island**

In our case study of Leach’s storm petrels, Machias Seal Island had substantially lower mean abundance than any other colony (mean count = 244 individuals; see Figure 2 in main text). Extremely small colonies can be subject to fundamentally different population dynamic processes than large colonies; thus, including Machias Seal Island may erode the ability of our model to share information on the shape of temporal trajectories among colonies. To examine this possibility, we re-fit the statistical model while omitting Machias Seal Island from the dataset. Fitted trajectories for each of the 12 storm petrel colonies are shown in Figure S3.1.

A graph of different types of data

Description automatically generated with medium confidence

**Figure S3.1.** Colony-level trajectories for Leach’s storm-petrel derived from the Bayesian hierarchical GAMM over the period 1966-2023 (ordered by latitude, 12 colony locations). Raw survey counts are shown as black points with standard error estimated by the GAMM (counts without raw estimated SE are indicated by open circles). The solid gray line and ribbons depict colony-level trajectories as posterior median and 95% confidence interval.

The overall regional trajectory is shown in Figure S3.2. Compared to the estimate presented in the main text of the manuscript which included Machias Seal Island (Figure 4), this estimate implies a higher degree of confidence that the regional population increased in the early portion of the time series (i.e., from 1970-1980).

A graph showing the evolution of the year

Description automatically generated with medium confidence

**Figure S3.2.** Regional-level trajectory for Leach’s storm-petrel derived from the Bayesian hierarchical GAMM over the period 1966-2023, when omitting Machias Seal Island from the dataset.

The estimated trend from 1966 to 2023 when omitting Machias Seal Island from the analysis is

A graph of a graph

Description automatically generated

**Appendix S4: JAGS code to fit the statistical model, and to generate simulations.**

Jags code to fit the statistical model is presented below:

model {

# --------------------------------

# Colony-level intercepts (note that GAM structure includes intercept for colony 1)

# --------------------------------

C[1] <- 0

for(k in 2:ncolony){

C[k] ~ dnorm(0,0.01)

}

# --------------------------------

# GAMM smooth

# --------------------------------

# Describes the variance among colonies in the shapes of their GAMs

sdbeta ~ dunif(0,5)

taubeta <- pow(sdbeta,-2)

nk1 <- nknots-1

nk2 <- ((nknots\*2)-2)

B.X[1] ~ dnorm(0,0.01)

## prior for s(year)...

K1 <- S1[1:nk1,1:nk1] \* lambda[1] + S1[1:nk1,(nknots:nk2)] \* lambda[2]

B.X[(2:nknots)] ~ dmnorm(zero[(2:nknots)],K1)

#K1 is the prior on the precisions of the mnorm B.X values (mean GAM parameters for a species)

## smoothing parameter

for(i in 1:2) {

lambda[i] ~ dgamma(0.05,0.005)

rho[i] <- log(lambda[i])

} # i

for(j in 1:nknots){ # Computation of GAM components

for(k in 1:ncolony){

# Draw betas for each colony (for this knot) from a shared distribution

beta.X[k,j] ~ dnorm(B.X[j],taubeta)

for ( i in 1:nyearspred ){

X.part[i,j,k] <- beta.X[k,j]\*(X[i,j])

} # i

} # k

} # j

# --------------------------------

# Colony-level dynamics

# --------------------------------

# Prior for annual process variation

ProcVar\_prior\_sd <- 0.2

ProcVar\_prior\_var <- pow(ProcVar\_prior\_sd,2)

ProcVar\_sd ~ dlnorm(log(0.2) - 0.5\*ProcVar\_prior\_var,1/ProcVar\_prior\_var)

ProcVar\_tau <- pow(ProcVar\_sd,-2)

for (i in 1:nyearspred){

for(k in 1:ncolony){

yeareffect[i,k] <- sum(X.part[i,1:nknots,k])

log\_mu[i,k] ~ dnorm(C[k] + yeareffect[i,k],ProcVar\_tau)

} # k

} # i

# --------------------------------

# Observation process

# --------------------------------

# priors on SE parameters

intercept\_SE ~ dnorm(0,0.1)

slope\_SE ~ dnorm(0,0.1)

sd\_SE ~ dunif(0,2)

tau\_SE <- pow(sd\_SE,-2)

for (i in 1:ncounts) {

# Filling in missing observation error for certain surveys

survey\_SE[i] ~ dlnorm(intercept\_SE + slope\_SE \* log(survey\_count[i]),tau\_SE)

survey\_tau[i] <- pow(survey\_SE[i],-2)

# Assume lambda is lognormally distributed

lambda\_count[i] ~ dnorm(exp(log\_mu[year[i],colony[i]]), survey\_tau[i])T(0,)

count[i] ~ dpois(lambda\_count[i])

}

# --------------------------------

# Derived parameters: predictions of annual expected counts

# --------------------------------

for (i in 1:nyearspred){

for(k in 1:ncolony){

population\_index[i,k] <- exp(C[k] + yeareffect[i,k] + 0.5\*pow(ProcVar\_sd,2))

}

} #i

# --------------------------------

# Goodness-of-fit evaluation (Posterior predictive check)

# Simulate new datasets under the fitted model

# Compare RMSE of simulated datasets to RMSE of empirical data

# --------------------------------

for (i in 1:ncounts) {

# Simulate new lambda and new counts under fitted model

simulated\_lambda\_count[i] ~ dnorm(exp(log\_mu[year[i],colony[i]]), survey\_tau[i])T(0,)

simulated\_count[i] ~ dpois(simulated\_lambda\_count[i])

# Calculate discrepancy measures (squared error) for actual and simulated data

sqE\_actual[i] <- pow(count[i] - population\_index[year[i],colony[i]] ,2)

sqE\_simulated[i] <- pow(simulated\_count[i] - population\_index[year[i],colony[i]],2)

}

# Overall root mean squared error of empirical data

RMSE\_actual <- sqrt(mean(sqE\_actual[]))

# Overall root mean squared error of simulated data

RMSE\_simulated <- sqrt(mean(sqE\_simulated[]))

}

Below, we include code to conduct simulations:

my\_packs = c('tidyverse',

'jagsUI',

'mgcv',

'scales',

'ggthemes')

if (any(!my\_packs %in% installed.packages()[, 'Package'])) {install.packages(my\_packs[which(!my\_packs %in% installed.packages()[, 'Package'])],dependencies = TRUE)}

lapply(my\_packs, require, character.only = TRUE)

rm(list=ls())

# ------------------------------------------------

# Set working directory

# ------------------------------------------------

setwd("")

# ------------------------------------------------

# ggplot theme

# ------------------------------------------------

CustomTheme <- theme\_set(theme\_bw())

CustomTheme <- theme\_update(legend.key = element\_rect(colour = NA),

legend.key.height = unit(1.2, "line"),

panel.grid.major = element\_line(colour = 'transparent'),

panel.grid.minor = element\_line(colour = 'transparent'),

panel.border = element\_rect(linetype = "solid",

colour = "black",

size = 1, fill = NA),

axis.line = element\_line(colour = "black"),

strip.text = element\_text(size = 12, colour = "black"),

strip.background = element\_rect(colour = "black",

fill = "lightblue2",

linetype = "solid"),

axis.title.y = element\_text(margin = margin(0,10,0,0)),

axis.title.x = element\_text(margin = margin(10,0,0,0)),

panel.background = element\_rect(fill = "white"))

`%!in%` <- Negate(`%in%`)

# ----------------------------------------------------------

# Part 1: Simulate an entire 50-year time series for each of 15 colonies

# ----------------------------------------------------------

simulation\_results <- data.frame()

for (run in 1:1000){

# How variable is the "shared" component of environmental variation?

for (sd\_shared in c(0,0.1)){

set.seed(run)

# Load results that have completed so far

if (file.exists("../output/model\_output/simulation\_results\_new.rds")) simulation\_results <- readRDS("../output/model\_output/simulation\_results\_new.rds")

# Skip this iteration, if it has already been run

if (nrow(simulation\_results) > 0 & sum(simulation\_results$run == run & simulation\_results$sd\_shared == sd\_shared)) next

ncolony <- 15

nyears <- 50

# Time-varying environmental covariate

env\_shared <- cumsum(rnorm(nyears,0,sd\_shared))

N\_matrix <- matrix(NA,nrow=ncolony,ncol = nyears)

# Initial abundance of each colony

N\_matrix[,1] <- rlnorm(ncolony, meanlog = log(10000), sdlog = 0.5) %>% sort()

# Simulate trajectories at each colony

for (i in 1:ncolony){

env\_unshared <- cumsum(rnorm(nyears,0,0.1))

# Generate a random trajectory at the colony (random walk)

for (t in 2:nyears){

N\_matrix[i,t] <- exp(log(N\_matrix[i,1]) + env\_shared[t] + env\_unshared[t] + rnorm(1,0,0.1))

}

}

# Convert to dataframe (to plot with ggplot)

N\_df <- reshape2::melt(N\_matrix) %>%

rename(Colony = Var1,Year=Var2,N=value)

# Plot dynamics (on log10 scale)

ggplot()+

geom\_line(aes(x = 1:nyears, y = colSums(N\_matrix), col = "Regional Sum"),linewidth = 1)+

geom\_line(data = N\_df, aes(x = Year, y = N, col = factor(Colony)))+

theme\_few()+

scale\_y\_continuous(labels = comma)+

ggtitle("Simulated trajectories at each of 15 colonies")+

ylab("Abundance")+

xlab("Year")

ggplot()+

geom\_line(aes(x = 1:nyears, y = colSums(N\_matrix), col = "Regional Sum"),linewidth = 1)+

theme\_few()+

scale\_y\_continuous(labels = comma, trans = "log10")+

ggtitle("Simulated trajectories at each of 9 colonies")+

ylab("Abundance")+

xlab("Year")

# ----------------------------------------------------------

# Part 2: Simulate intermittent surveys (2-6 surveys at each colony)

# ----------------------------------------------------------

intercept\_SE = -0.95

slope\_SE = 0.86

sd\_SE = 0.45

# Magnitude of observation error varies among surveys

N\_df$survey\_SE <- rlnorm(nrow(N\_df),intercept\_SE + slope\_SE \* log(N\_df$N),sd\_SE)

N\_df$lambda\_obs <- rnorm(nrow(N\_df), N\_df$N, N\_df$survey\_SE)

N\_df$lambda\_obs[N\_df$lambda\_obs <0] <- 0

# -------------------------------

# Which surveys were actually conducted?

# -------------------------------

N\_df$SurveyCount <- NA

for (i in 1:ncolony){

# Rows in N\_df corresponding to observations at this colony

colony\_rows <- which(N\_df$Colony == i)

survey\_years <- c()

# Simulate one count in first 10 years of surveys

survey\_years <- c(survey\_years, sample(1:10,1))

# Simulate one count in final 10 years of surveys

survey\_years <- c(survey\_years, sample(nyears:(nyears-10),1))

# Simulate 2-4 additional surveys

survey\_years <- c(survey\_years, sample(6:(nyears-6),sample(0:4,1)))

# Poisson observations

N\_df$SurveyCount[(N\_df$Colony == i) &

(N\_df$Year %in% survey\_years)] <- rpois(n = length(survey\_years),

lambda = N\_df$lambda\_obs[(N\_df$Colony == i) & (N\_df$Year %in% survey\_years)])

}

# Omit 35% of standard errors to mimick missing information in empirical data

survey\_rows <- which(!is.na(N\_df$SurveyCount))

SEs\_to\_drop <- sample(survey\_rows,round(length(survey\_rows)\*0.35))

N\_df$survey\_SE[SEs\_to\_drop] <- NA

# Confidence intervals on counts

N\_df$SurveyCount\_lci <- N\_df$SurveyCount - 1.96\*N\_df$survey\_SE

N\_df$SurveyCount\_uci <- N\_df$SurveyCount + 1.96\*N\_df$survey\_SE

# Plot survey counts at each of the colonies

ggplot()+

geom\_line(data = N\_df, aes(x = Year, y = N), col = "gray80")+

geom\_errorbar(data = N\_df, aes(x = Year, ymin = SurveyCount\_lci, ymax = SurveyCount\_uci), col = "black", width = 0)+

geom\_point(data = N\_df, aes(x = Year, y = SurveyCount), col = "black")+

geom\_point(data = subset(N\_df, is.na(survey\_SE)), aes(x = Year, y = SurveyCount), col = "black", pch = 1, size =5)+

theme\_few()+

facet\_wrap(Colony~.)+

scale\_y\_continuous(labels = comma, trans = "log10")+

ggtitle("Simulated surveys at colonies")

# Print at 8x6 for appendix

# ----------------------------------------------------------

# Part 3: Fit model to simulated survey data

# ----------------------------------------------------------

spdat <- N\_df %>% subset(!is.na(SurveyCount)) %>% dplyr::select(Colony,Year,SurveyCount,survey\_SE)

# Data for import into jags

nknots = 6

year <- spdat$Year

ymax <- nyears

colony = spdat$Colony

count <- spdat$SurveyCount

ncounts = length(count)

obs\_tau = 1/spdat$log\_observation\_SE^2

# Use jagam to prepare basis functions

nyearspred = length(1:ymax)

preddat = data.frame(yrs = 1:ymax,count = 1)

form = as.formula(paste("count ~ s(yrs,k =",nknots,")"))

gamprep = jagam(formula = form,

data = preddat,

file = "tempgam.txt",

centred = T)

# Package data into a list for JAGS

jags\_data = list(X = gamprep$jags.data$X,

S1 = gamprep$jags.data$S1,

zero = gamprep$jags.data$zero,

colony = colony,

ncounts = ncounts,

ncolony = ncolony,

count = count,

nknots = nknots,

nyearspred = nyearspred,

year = year,

survey\_count = count,

survey\_SE = spdat$survey\_SE)

# Fit model using JAGS

parameters.to.save = c("intercept\_SE","slope\_SE","sd\_SE","population\_index")

out <- jags(data = jags\_data,

parameters.to.save = parameters.to.save,

inits = NULL,

n.iter = 200000,

n.burnin = 100000,

n.thin = 50,

model.file = "Seabird\_Model.jags",

n.chains = 3,

parallel = TRUE)

out$mcmc.info$elapsed.mins # ~1.5 mins

# ----------------------------------------------------------

# Part 4: Summarize predictions and compare to true (i.e., simulated) trajectories

# ----------------------------------------------------------

# Extract predictions in dataframe format

fit\_samples = reshape2::melt(out$sims.list$population\_index) %>%

rename(samp = Var1, Year = Var2, Colony = Var3, N\_pred = value)

N\_summary\_colony = fit\_samples %>%

group\_by(Colony, Year) %>%

summarize(q025 = quantile(N\_pred,0.025),

q50 = quantile(N\_pred,0.500),

mean = mean(N\_pred),

q975 = quantile(N\_pred,0.975)) %>%

# Join with true values

full\_join(N\_df)

# Plot estimates

ggplot()+

geom\_ribbon(data = N\_summary\_colony, aes(x = Year, ymin = q025, ymax = q975), alpha = 0.2, fill = "dodgerblue")+

geom\_line(data = N\_summary\_colony, aes(x = Year, y = q50, col = "Estimate"))+

geom\_line(data = N\_df, aes(x = Year, y = N, col = "True Trajectory"))+

geom\_errorbar(data = N\_df, aes(x = Year, ymin = SurveyCount\_lci, ymax = SurveyCount\_uci), col = "black", width = 0)+

geom\_point(data = N\_df, aes(x = Year, y = SurveyCount), col = "black")+

geom\_point(data = subset(N\_df, is.na(survey\_SE)), aes(x = Year, y = SurveyCount), col = "black", pch = 1, size =5)+

theme\_few()+

facet\_wrap(Colony~.)+

scale\_y\_continuous(trans="log10", labels = comma)+

ggtitle("Simulated regional trajectory")+

scale\_color\_manual(values = c("dodgerblue","red"), name = "")+

ylab("Index of abundance")

# ----------------------------------------------------------

# Part 5: Calculate regional annual indices as sum of individual colony annual indices

# ----------------------------------------------------------

# True regional annual indices

regional\_df <- N\_df %>%

group\_by(Year) %>%

summarize(N = sum(N))

# Fit a GAM through the regional indices.

# This is the "true" trajectory we are trying to estimate

regional\_df$log\_N <- log(regional\_df$N)

gam\_true <- gam(log\_N~s(Year, k = -1), data = regional\_df)

regional\_df$population\_index <- exp(predict(gam\_true))

# Estimated regional annual indices

regional\_samples = fit\_samples %>%

group\_by(Year,samp) %>%

summarize(N\_pred = sum(N\_pred))

# Summary (mean and 95% CI)

N\_summary\_regional <- regional\_samples %>%

group\_by(Year) %>%

summarize(q025 = quantile(N\_pred,0.025),

q50 = quantile(N\_pred,0.500),

mean = mean(N\_pred),

q975 = quantile(N\_pred,0.975)) %>%

# Join with true values

full\_join(regional\_df)

# ----------------------------------------------------------

# 50-year trend estimate

# ----------------------------------------------------------

baseline\_year <- 1

trend\_true\_50yr <- 100\*((regional\_df$population\_index[regional\_df$Year == nyears]/regional\_df$population\_index[regional\_df$Year == baseline\_year])^(1/(nyears-baseline\_year))-1)

trend\_est\_50yr <- 100\*((regional\_samples$N\_pred[regional\_samples$Year == nyears]/regional\_samples$N\_pred[regional\_samples$Year == baseline\_year])^(1/(nyears-baseline\_year))-1)

trend\_est\_50yr <- quantile(trend\_est\_50yr,c(0.025,0.5,0.975))

regional\_plot\_50yr <- ggplot()+

geom\_ribbon(data = N\_summary\_regional, aes(x = Year, ymin = q025, ymax = q975), alpha = 0.2, fill = "dodgerblue")+

geom\_line(data = N\_summary\_regional, aes(x = Year, y = q50, col = "Estimate"))+

geom\_line(data = regional\_df, aes(x = Year, y = N, col = "True trajectory"))+

geom\_line(data = regional\_df,aes(x = Year, y = population\_index, col = "True trajectory (smoothed)"))+

theme\_few()+

scale\_y\_continuous(labels = comma)+

ggtitle("Regional trajectory")+

scale\_color\_manual(values = c("dodgerblue","red","black"), name = "")+

ylab("Index of abundance")+

geom\_text(aes(x = 0,

y = max(c(N\_summary\_regional$q975,N\_summary\_regional$N))),

label = paste0("True trend = ",round(trend\_true\_50yr,2),"% per year\nEst trend = ",round(trend\_est\_50yr[2],2),"% (",round(trend\_est\_50yr[1],2)," to ",round(trend\_est\_50yr[3],2),")"), hjust=0)

regional\_plot\_50yr

# ----------------------------------------------------------

# 10-year trend estimate

# ----------------------------------------------------------

baseline\_year <- 41

trend\_true\_10yr <- 100\*((regional\_df$population\_index[regional\_df$Year == nyears]/regional\_df$population\_index[regional\_df$Year == baseline\_year])^(1/(nyears-baseline\_year))-1)

trend\_est\_10yr <- 100\*((regional\_samples$N\_pred[regional\_samples$Year == nyears]/regional\_samples$N\_pred[regional\_samples$Year == baseline\_year])^(1/(nyears-baseline\_year))-1)

trend\_est\_10yr <- quantile(trend\_est\_10yr,c(0.025,0.5,0.975))

regional\_plot\_10yr <- ggplot()+

geom\_ribbon(data = N\_summary\_regional, aes(x = Year, ymin = q025, ymax = q975), alpha = 0.2, fill = "dodgerblue")+

geom\_line(data = N\_summary\_regional, aes(x = Year, y = q50, col = "Estimate"))+

geom\_line(data = regional\_df, aes(x = Year, y = N, col = "True trajectory"))+

geom\_line(data = regional\_df,aes(x = Year, y = population\_index, col = "True trajectory (smoothed)"))+

theme\_few()+

scale\_y\_continuous(labels = comma)+

ggtitle("Regional trajectory")+

scale\_color\_manual(values = c("dodgerblue","black","red"), name = "")+

ylab("Index of abundance")+

geom\_text(aes(x = 0,

y = max(c(N\_summary\_regional$q975,N\_summary\_regional$N))),

label = paste0("True trend = ",round(trend\_true\_10yr,2),"% per year\nEst trend = ",round(trend\_est\_10yr[2],2),"% (",round(trend\_est\_10yr[1],2)," to ",round(trend\_est\_10yr[3],2),")"), hjust=0)

# ----------------------------------------------------------

# Append results for this simulation run to dataframe

# ----------------------------------------------------------

if (file.exists("../output/model\_output/simulation\_results\_new.rds")) simulation\_results <- readRDS("../output/model\_output/simulation\_results\_new.rds")

simulation\_results <- rbind(simulation\_results,data.frame(run = run,

sd\_shared = sd\_shared,

trend\_true\_50yr = trend\_true\_50yr,

trend\_est\_50yr\_q025 = trend\_est\_50yr[1],

trend\_est\_50yr\_q500 = trend\_est\_50yr[2],

trend\_est\_50yr\_q975 = trend\_est\_50yr[3],

cov\_50yr = trend\_true\_50yr > trend\_est\_50yr[1] & trend\_true\_50yr < trend\_est\_50yr[3],

trend\_true\_10yr = trend\_true\_10yr,

trend\_est\_10yr\_q025 = trend\_est\_10yr[1],

trend\_est\_10yr\_q500 = trend\_est\_10yr[2],

trend\_est\_10yr\_q975 = trend\_est\_10yr[3],

cov\_10yr = trend\_true\_10yr > trend\_est\_10yr[1] & trend\_true\_10yr < trend\_est\_10yr[3],

max\_Rhat = max(out$Rhat$population\_index)))

saveRDS(simulation\_results, file = "../output/model\_output/simulation\_results\_new.rds")

} # sd\_shared

# ----------------------------------------------------------

# Plot results

# ----------------------------------------------------------

simulation\_results <- readRDS("../output/model\_output/simulation\_results\_new.rds")

lim = range(simulation\_results[,c("trend\_true\_50yr","trend\_est\_50yr\_q025","trend\_est\_50yr\_q975",

"trend\_true\_10yr","trend\_est\_10yr\_q025","trend\_est\_10yr\_q975")])

trend\_plot\_50yr <- ggplot(data = simulation\_results, aes(x = trend\_true\_50yr, y = trend\_est\_50yr\_q500, ymin = trend\_est\_50yr\_q025, ymax = trend\_est\_50yr\_q975,col=cov\_50yr))+

geom\_abline(intercept=0,slope=1,col="gray85")+

geom\_errorbar(width=0)+

geom\_point()+

coord\_cartesian(ylim=lim,xlim=lim)+

theme\_bw()+

xlab("True (simulated) regional 50-year trend")+

ylab("Estimated regional 50-year trend")+

scale\_color\_manual(values=c("red","dodgerblue"), name = "Coverage")+

ggtitle("50-year trend estimates")+

facet\_grid(sd\_shared~.)

trend\_plot\_10yr <- ggplot(data = simulation\_results, aes(x = trend\_true\_10yr, y = trend\_est\_10yr\_q500, ymin = trend\_est\_10yr\_q025, ymax = trend\_est\_10yr\_q975,col=cov\_10yr))+

geom\_abline(intercept=0,slope=1,col="gray85")+

geom\_errorbar(width=0)+

geom\_point()+

coord\_cartesian(ylim=lim,xlim=lim)+

theme\_bw()+

xlab("True (simulated) regional 10-year trend")+

ylab("Estimated regional 10-year trend")+

scale\_color\_manual(values=c("red","dodgerblue"), name = "Coverage")+

ggtitle("10-year trend estimates")+

facet\_grid(sd\_shared~.)

#print(trend\_plot\_50yr)

}

# ----------------------------------------------------------

# Summarize results across repeated simulations

# ----------------------------------------------------------

simulation\_results <- readRDS("../output/model\_output/simulation\_results\_new.rds")

# Remove runs that failed to converge

simulation\_results\_converged <- subset(simulation\_results, max\_Rhat <= 1.05)

# ----------------------------------------------------------

# Plot results

# ----------------------------------------------------------

lim = range(simulation\_results\_converged[,c("trend\_true\_50yr","trend\_est\_50yr\_q025","trend\_est\_50yr\_q975",

"trend\_true\_10yr","trend\_est\_10yr\_q025","trend\_est\_10yr\_q975")])

trend\_plot\_50yr <- ggplot(data = simulation\_results\_converged, aes(x = trend\_true\_50yr, y = trend\_est\_50yr\_q500, ymin = trend\_est\_50yr\_q025, ymax = trend\_est\_50yr\_q975,col=cov\_50yr))+

geom\_abline(intercept=0,slope=1,col="gray85")+

geom\_errorbar(width=0)+

geom\_point()+

coord\_cartesian(ylim=lim,xlim=lim)+

theme\_bw()+

xlab("True (simulated) regional 50-year trend")+

ylab("Estimated regional 50-year trend")+

scale\_color\_manual(values=c("red","dodgerblue"), name = "Coverage")+

ggtitle("50-year trend estimates")+

facet\_grid(sd\_shared~.)

print(trend\_plot\_50yr)

trend\_plot\_10yr <- ggplot(data = simulation\_results\_converged, aes(x = trend\_true\_10yr, y = trend\_est\_10yr\_q500, ymin = trend\_est\_10yr\_q025, ymax = trend\_est\_10yr\_q975,col=cov\_10yr))+

geom\_abline(intercept=0,slope=1,col="gray85")+

geom\_errorbar(width=0)+

geom\_point()+

coord\_cartesian(ylim=lim,xlim=lim)+

theme\_bw()+

xlab("True (simulated) regional 10-year trend")+

ylab("Estimated regional 10-year trend")+

scale\_color\_manual(values=c("red","dodgerblue"), name = "Coverage")+

ggtitle("10-year trend estimates")+

facet\_grid(sd\_shared~.)

print(trend\_plot\_10yr)

# Summarize accuracy and bias for 50-year trend estimates

simulation\_results\_converged %>%

group\_by(sd\_shared) %>%

summarize(n = n(),

mean\_bias = mean(trend\_est\_50yr\_q500 - trend\_true\_50yr),

SE\_bias = sd(trend\_est\_50yr\_q500 - trend\_true\_50yr),

coverage = mean(cov\_50yr))

# Summarize accuracy and bias for 10-year trend estimates

simulation\_results\_converged %>%

group\_by(sd\_shared) %>%

summarize(n = n(),

mean\_bias = mean(trend\_est\_10yr\_q500 - trend\_true\_10yr),

SE\_bias = sd(trend\_est\_10yr\_q500 - trend\_true\_10yr),

coverage = mean(cov\_10yr))