Case Study 1

This case study was developed primarily by E.Wolkovich, D.Loughnan and X.Wang with the input of the full manuscript author team.

Data analysis

The following code performs data analysis and visualization using both the traditional Fisherian approach and the Bayesian approach. For the Fisherian approach, we use an `lm` model and plot the p-values for different populations, indicating whether the increase or decrease is statistically significant. For the Bayesian approach, we use Stan code with priors and plot the posterior distribution.

```
# Seed we selected from data simulation
set.seed(1546)
# Dropping -1600, 1600 with p-value smaller than 0.05.
a <- c(-2000, -1600, -1400,  600, 1400 , 1600,  2000)
t <- 10
time <- 1:t
b <- seq(40000, 101000, by = 10000)
noise_sd <- c(7000, 6600, 6200, 5800, 5400, 5000,4600)
```

Traditional Fisherian approach

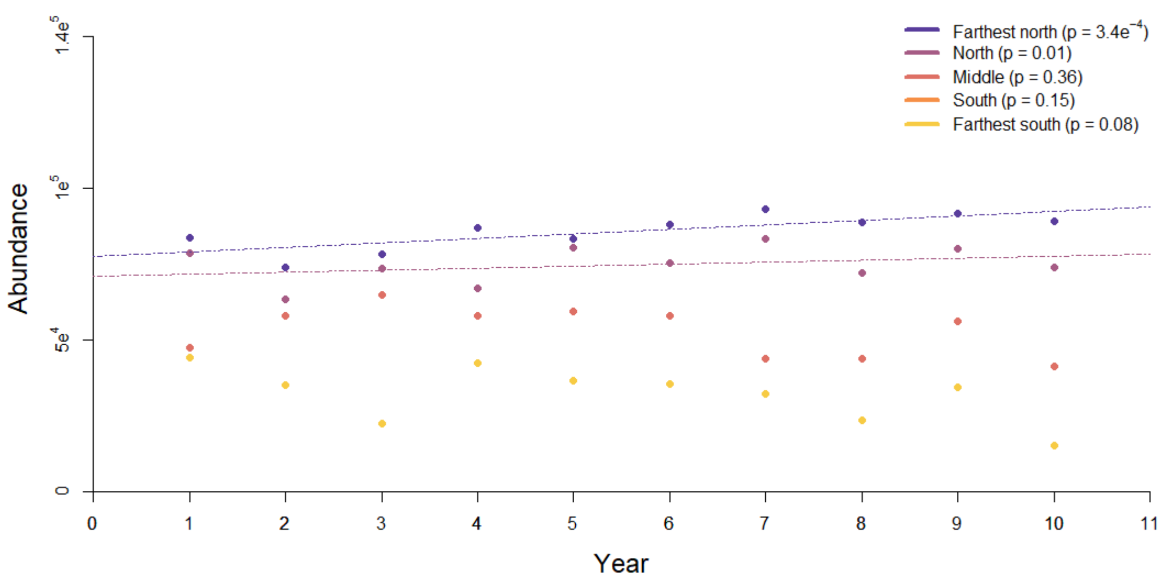
With traditional Fisherian approach, we use lm model to plot

```
output <- data.frame(iter = seq(1:(length(a)*length(time))),
                     pop = rep(1:length(a), each = length(time)),
                     year = rep(1:t, times = length(a)))
output$pred <- NA
```

```
abund <- vector()
for(i in 1:length(a)){
  y <- numeric(t)
  time <- 1:t
  y <- a[i]*time + b[i] + rnorm(t, 0, noise_sd[i])
  abund <- rbind(abund, y)
}

abund <- data.frame(reshape2::melt(t(abund)))
output$pred <- abund$value
```

Visualization



Bayesian approach

With Bayesian approach, we run the stan code:

```
data {
  int<lower=0> N; //No. obs
```

```
  int<lower=0> Ngrp; //No. in group---population or species
  int group[N]; // Group type
  vector[N] year;
  real ypred[N]; //response
}

parameters {
  real a[Ngrp] ;
  real b[Ngrp];
  real mu_a;
  real<lower=0> sigma_a;
  real mu_b;
  real<lower=0> sigma_b;
  real<lower=0> sigma_y;

}

model {

real mu_y[N];

for(i in 1:N){
     mu_y[i] = a[group[i]] + b[group[i]] * year[i];
  }

a ~ normal(mu_a, sigma_a);
b ~ normal(mu_b, sigma_b);

//Priors
mu_a ~ normal(188, 50);
sigma_a ~ normal(0,50);
mu_b ~ normal(0,10);
sigma_b ~ normal(0,10);
sigma_y ~ normal(0,10);

ypred ~ normal(mu_y, sigma_y);
}
```

```
'data.frame':   50 obs. of  4 variables:
 $ iter: int  1 2 3 4 5 6 7 8 9 10 ...
 $ pop : int  1 1 1 1 1 1 1 1 1 1 ...
 $ year: int  1 2 3 4 5 6 7 8 9 10 ...
```

```
$ pred: num  43951 34961 22390 42250 36469 ...
```

```
mdlPop <- stan("partialPoolSimMdl.stan",
               data = datalistGrp)

sum <- summary(mdlPop)$summary

intercept <- sum[grep("a\\[", rownames(sum)), "mean"]
slopes <- sum[grep("b\\[", rownames(sum)), "mean"]

# Posterior distribution
post <- rstan::extract(mdlPop)
```

Visualization