SEACAR Coral Analysis

Last compiled on 30 June, 2022

# Important Notes

All scripts and outputs can be found on the SEACAR GitHub repository:

<https://github.com/FloridaSEACAR/SEACAR_Panzik>

This script is based off of code originally wrtten by Katie May Laumann

# Libraries and Settings

Loads libraries used in the script. Loads the Segoe UI font for use in the figures. The inclusion of scipen option limits how frequently R defaults to scientific notation. Sets default settings for displaying warning and messages in created document, and sets figure dpi.

library(knitr)  
library(data.table)  
library(dplyr)  
library(lubridate)  
library(ggplot2)  
library(scales)  
library(tidyr)  
library(gridExtra)  
library(tidyverse)  
library(hrbrthemes)  
library(nlme)  
windowsFonts(`Segoe UI` = windowsFont('Segoe UI'))  
options(scipen=999)  
opts\_chunk$set(warning=FALSE, message=FALSE, dpi=200)

# File Import

Imports percent cover files for each region for analysis.

file\_list <- list.files("data", pattern="Percent Cover", full=TRUE)  
  
file\_in <- file\_list[grep("SE FL", file\_list)]  
CorSE <- fread(file\_in, sep=",", header=TRUE, stringsAsFactors=FALSE,  
 na.strings="")  
  
file\_in <- file\_list[grep("DRY TORT", file\_list)]  
CorDRYTORTUGAS <- fread(file\_in, sep=",", header=TRUE, stringsAsFactors=FALSE,  
 na.strings="")  
  
file\_in <- file\_list[grep("FLA KEYS", file\_list)]  
CorFLKEYS <- fread(file\_in, sep=",", header=TRUE, stringsAsFactors=FALSE,  
 na.strings="")

# Data Filtering

The processing and filtering that is done to the data is as follows for each region:

1. Removes non-coral species
2. [PercentCover-SpeciesComposition\_%] column is renamed perccov
3. Removes data that contains NA values in percent cover
4. Create variable with full species name
5. Write data to file

Afterwards, the data from each region is combined into a single data frame.

#We have more than corals here: see Groups 1 and 2.  
#We only want corals, so filter by Group 2.  
CorSE<-CorSE%>%filter(SpeciesGroup2=="Octocoral"|SpeciesGroup2=="Unspecifiedcoral")  
#Current datasets do not include sampling method, which was requested. If sampling method is added to the dataset,  
#seperate analyses by sampling method by first building a df for each method:  
#CorSEMethod1<-CorSE%>%MethodRowName=="Method1"  
#CorSEMethod2<-CorSE%>%MethodRowName=="Method2"  
#then by running all of the following analyses for each method individually.   
#This could be simplified by using lapply, if desired.  
  
#rename columns to more easily work with them  
colnames(CorSE)[colnames(CorSE) == "[PercentCover-SpeciesComposition\_%]"] <- "Perccov"  
  
#remove % cover NAs  
CorSE <- CorSE[!is.na(CorSE$Perccov), ]  
  
#convert to a numeric to be able to do math  
CorSE$Perccov<-as.numeric(CorSE$Perccov)  
CorSE <- CorSE[!is.na(CorSE$Perccov), ]  
  
#combine genus and spp into one column in case you want to refine analyses to the species level.  
CorSE$gensp<-paste(CorSE$GenusName, CorSE$SpeciesName, sep=" ")  
CorSE<-CorSE%>%filter(gensp!="NA NA")  
  
#Write to file  
write.csv(CorSE,"data/CorSE.csv")  
  
#We have more than corals here: see Groups 1 and 2.  
#We only want corals, so filter by Group 2.  
CorFLKEYS<-CorFLKEYS%>%filter(SpeciesGroup2=="Octocoral"|SpeciesGroup2=="Unspecifiedcoral")  
  
#rename columns to more easily work with them  
colnames(CorFLKEYS)[colnames(CorFLKEYS) == "[PercentCover-SpeciesComposition\_%]"] <- "Perccov"  
  
#remove % cover NAs  
CorFLKEYS <- CorFLKEYS[!is.na(CorFLKEYS$Perccov), ]   
  
#convert to a numeric to be able to do math  
CorFLKEYS$Perccov<-as.numeric(CorFLKEYS$Perccov)  
CorFLKEYS <- CorFLKEYS[!is.na(CorFLKEYS$Perccov), ]   
  
#combine genus and spp into one column in case you want to refine analyses to the species level.  
CorFLKEYS$gensp<-paste(CorFLKEYS$GenusName, CorFLKEYS$SpeciesName, sep=" ")  
CorFLKEYS<-CorFLKEYS%>%filter(gensp!="NA NA")  
  
#Write to file  
write.csv(CorFLKEYS,"data/CorFLKeys.csv")  
  
#We have more than corals here: see Groups 1 and 2.  
#We only want corals, so filter by Group 2.  
CorDRYTORTUGAS<-CorDRYTORTUGAS%>%filter(SpeciesGroup2=="Octocoral"|SpeciesGroup2=="Unspecifiedcoral")  
  
#rename columns to more easily work with them  
colnames(CorDRYTORTUGAS)[colnames(CorDRYTORTUGAS) == "[PercentCover-SpeciesComposition\_%]"] <- "Perccov"  
  
#remove % cover NAs  
CorDRYTORTUGAS <- CorDRYTORTUGAS[!is.na(CorDRYTORTUGAS$Perccov), ]   
  
#convert to a numeric to be able to do math  
CorDRYTORTUGAS$Perccov<-as.numeric(CorDRYTORTUGAS$Perccov)  
CorDRYTORTUGAS <- CorDRYTORTUGAS[!is.na(CorDRYTORTUGAS$Perccov), ]   
  
#combine genus and spp into one column in case you want to refine analyses to the species level.  
CorDRYTORTUGAS$gensp<-paste(CorDRYTORTUGAS$GenusName, CorDRYTORTUGAS$SpeciesName, sep=" ")  
CorDRYTORTUGAS<-CorDRYTORTUGAS%>%filter(gensp!="NA NA")  
  
#Write to file  
write.csv(CorDRYTORTUGAS,"data/CorDryTortugas.csv")  
  
#Combine data  
CorSE <- CorSE %>%  
 filter(ManagedAreaName=="Coral ECA")  
CorFLKEYS <- CorFLKEYS %>%  
 filter(ManagedAreaName=="Coral ECA"|ManagedAreaName=="Florida Keys NMS"|  
 ManagedAreaName=="Coupon Bight")  
CorDRYTORTUGAS <- CorDRYTORTUGAS %>%  
 filter(ManagedAreaName=="Florida Keys NMS")  
  
Cor <- rbind(CorSE,CorFLKEYS)  
Cor <- rbind(Cor,CorDRYTORTUGAS)  
  
#Remove duplicates  
Cor <- Cor%>%filter(MADup!=2)

# Linera Mixed Effects Models

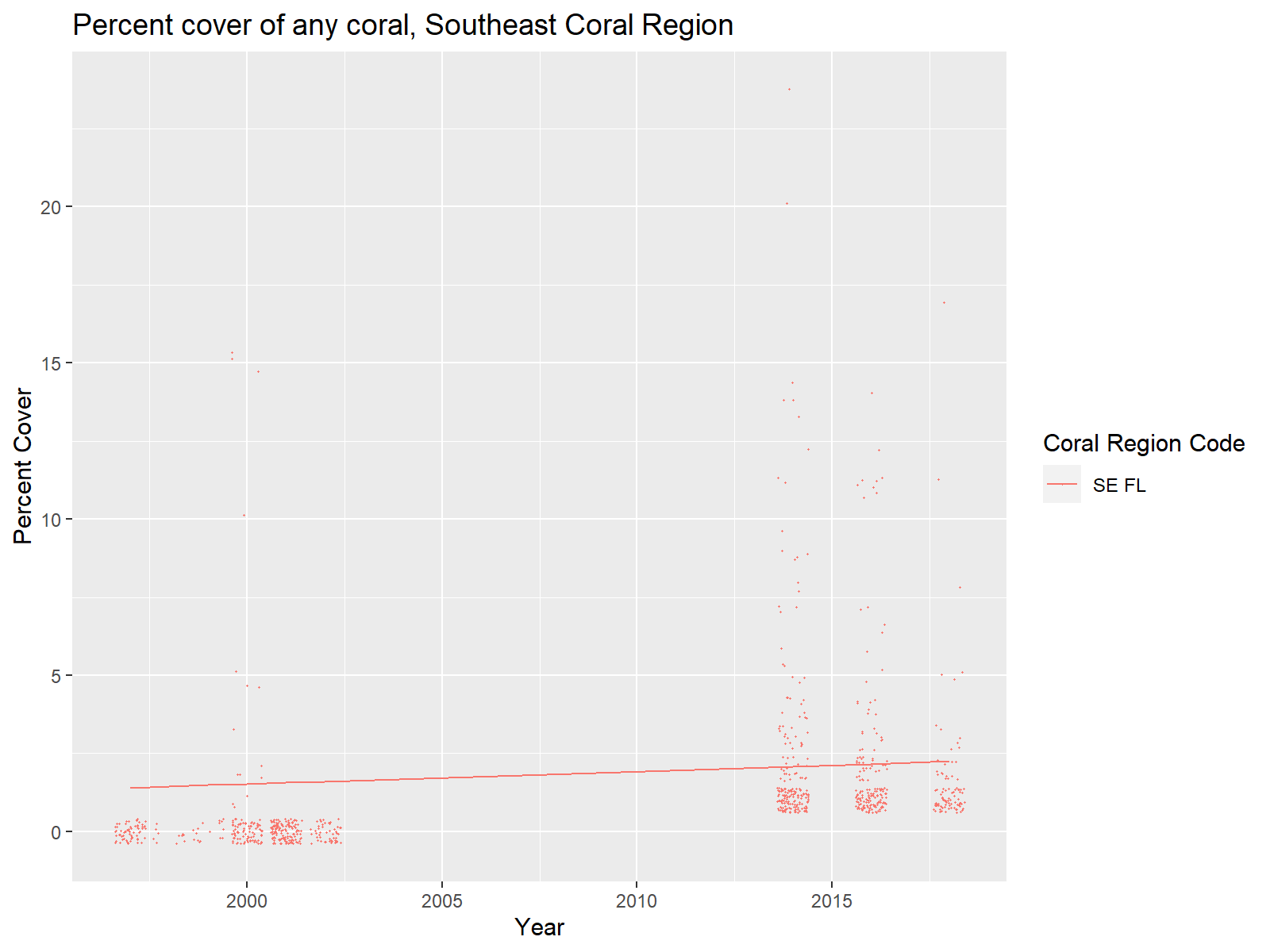
Performs a linear mixed effects (LME) model on each region between percent cover and year.

coralperccov <- Cor%>%  
 group\_by(ManagedAreaName,Year)%>%  
 summarise(mean(Perccov))  
  
colnames(coralperccov) [3] <- "MeanPercCov"  
  
# coralmodel<-lme(GenericRichness ~ MeanPercCov,  
# random =~1|Year,  
# na.action = na.omit,  
# data = coralperccov)  
  
CorSE\_AnyCoral<-lme(Perccov ~ Year,  
 random =~1|ProgramLocationID,  
 na.action = na.omit,  
 data = CorSE)  
  
#Here is where you examine your options for setting random variables and modifying your model.  
#For example, there are only 2 programs sampling, but many program location IDs. For example,   
#you may want to allow for different starting amounts of coral in each program location ID.   
#To run multiple lmes at different levels, for example by SpeciesGroup2, do the following:  
#Use the same type of model as above, but run it for each level (specified as i in the script)  
#and print them to a list using lapply.  
  
#In the following, we will allow for variation by program location ID.   
CorSE\_AnyCoral\_lmelist2 <- lapply(unique(CorSE$Coral\_Region),   
 function(i)summary(  
 lme(Perccov ~ Year,   
 random =~1|ProgramLocationID,  
 na.action = na.omit,  
 data = subset(CorSE,  
 Coral\_Region == i))))  
  
#create and add a names list to the output, with each name corresponding to one  
#unique taxon for the MA/in the model  
names(CorSE\_AnyCoral\_lmelist2) <- unique(CorSE$Coral\_Region)  
  
  
##FL Keys  
CorFLKEYS\_AnyCoral<-lme(Perccov ~ Year,  
 random =~1|ProgramLocationID,  
 na.action = na.omit,  
 data = CorFLKEYS)  
  
#Here is where you examine your options for setting random variables and modifying your model.  
#For example, there are only 2 programs sampling, but many program location IDs. For example,   
#you may want to allow for different starting amounts of coral in each program location ID.   
#To run multiple lmes at different levels, for example by SpeciesGroup2, do the following:  
#Use the same type of model as above, but run it for each level (specified as i in the script)  
#and print them to a list using lapply.  
  
#In the following, we will allow for variation by program location ID.   
CorFLKEYS\_AnyCoral\_lmelist2<-lapply(unique(CorFLKEYS$Coral\_Region),   
 function(i)summary(lme(Perccov ~ Year,  
 random =~1|ProgramLocationID,  
 na.action = na.omit, data = subset(CorFLKEYS, Coral\_Region == i))))  
  
#create and add a names list to the output, with each name corresponding to one  
#unique taxon for the MA/in the model  
names(CorFLKEYS\_AnyCoral\_lmelist2) <- unique(CorFLKEYS$Coral\_Region)  
  
#Dry Tortugas  
CorDRYTORTUGAS\_AnyCoral<-lme(Perccov ~ Year,  
 random =~1|ProgramLocationID,  
 na.action = na.omit,  
 data = CorDRYTORTUGAS)  
  
#Here is where you examine your options for setting random variables and modifying your model.  
#For example, there are only 2 programs sampling, but many program location IDs. For example,   
#you may want to allow for different starting amounts of coral in each program location ID.   
#To run multiple lmes at different levels, for example by SpeciesGroup2, do the following:  
#Use the same type of model as above, but run it for each level (specified as i in the script)  
#and print them to a list using lapply.  
  
#In the following, we will allow for variation by program location ID.   
CorDRYTORTUGAS\_AnyCoral\_lmelist2<-lapply(unique(CorDRYTORTUGAS$Coral\_Region),   
 function(i)summary(  
 lme(Perccov ~ Year,  
 random =~1|ProgramLocationID,  
 na.action = na.omit,  
 data = subset(CorDRYTORTUGAS,  
 Coral\_Region == i))))  
  
#create and add a names list to the output, with each name corresponding to one  
#unique taxon for the MA/in the model  
names(CorDRYTORTUGAS\_AnyCoral\_lmelist2) <- unique(CorDRYTORTUGAS$Coral\_Region)

# Appendix I: Plots

Plots are created for the LME results for each region and saved to a png file.

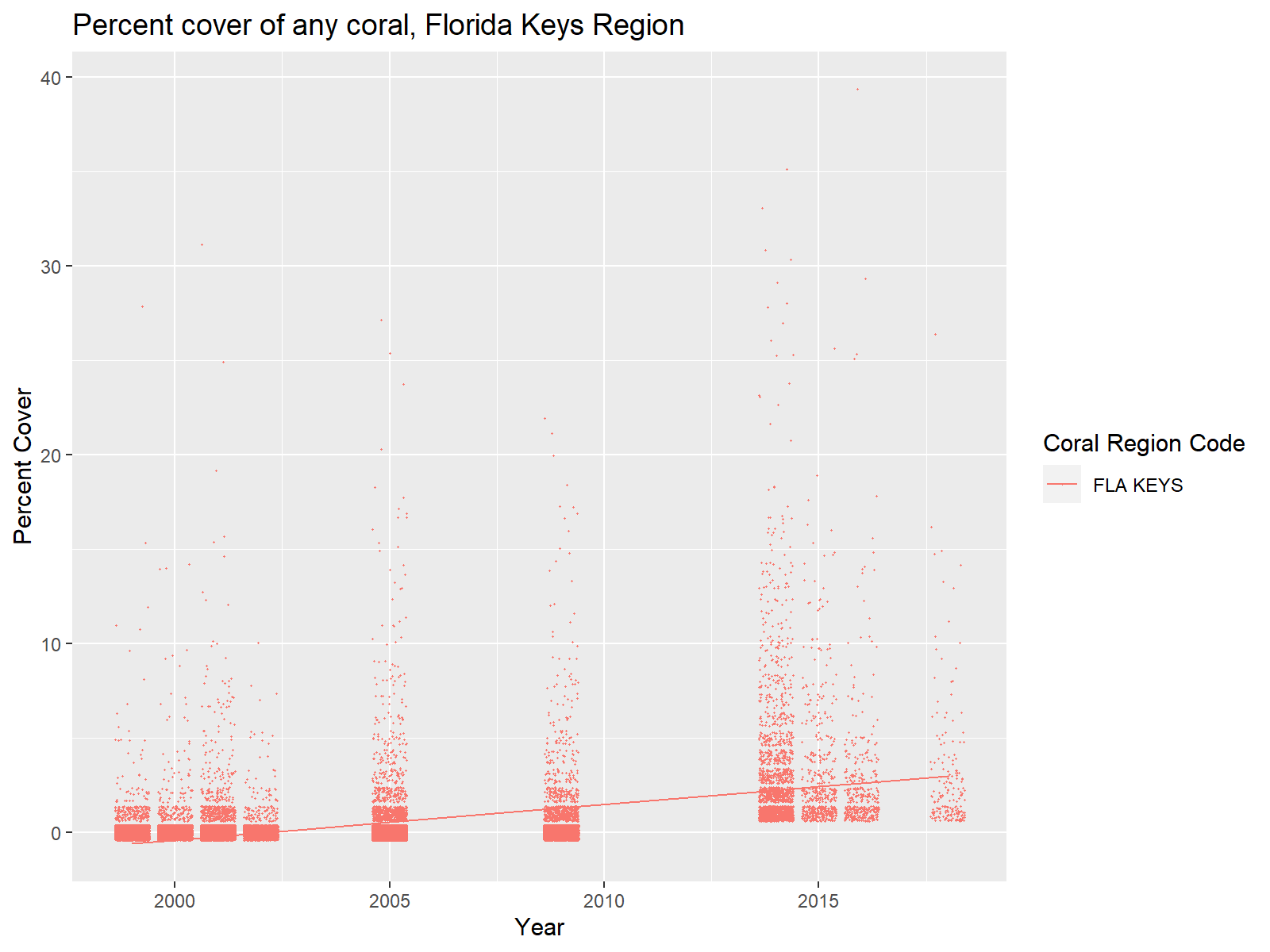
##SE FL  
preds <- do.call(rbind, lapply(names(CorSE\_AnyCoral\_lmelist2), function(i) {  
 mod <- CorSE\_AnyCoral\_lmelist2[[i]]  
 pred <- data.frame(Coral\_Region = i, Year = mod$data$Year, pred = predict(mod, level = 0))  
}))  
#using that dataframe, plot, for each taxon, the observed values in each programlocationID  
#for each year. Plot these as 'jittered' points.   
#create a line that is the fitted model for each taxon, as well.  
Plot\_CorSE\_AnyCoral<-ggplot() +  
 geom\_jitter(data = CorSE, aes(x = Year, y = Perccov, col = Coral\_Region),size=0.1) +  
 geom\_line(data = preds, aes(x = Year, y = pred, group = Coral\_Region, col = Coral\_Region))  
#label the plot and y axis (x was already labeled above)  
Plot\_CorSE\_AnyCoral<-Plot\_CorSE\_AnyCoral + labs(y = "Percent Cover")  
Plot\_CorSE\_AnyCoral<-Plot\_CorSE\_AnyCoral + labs(title = "Percent cover of any coral, Southeast Coral Region")  
#color code your points and lines by taxon  
Plot\_CorSE\_AnyCoral<-Plot\_CorSE\_AnyCoral + labs(colour = "Coral Region Code")  
Plot\_CorSE\_AnyCoral



#print to a png  
png("output/CorallmePlot\_AnyCoral\_SERegion.png",  
 width = 8,  
 height = 4,  
 units = "in",  
 res = 200)  
print(Plot\_CorSE\_AnyCoral)  
dev.off()

png 2

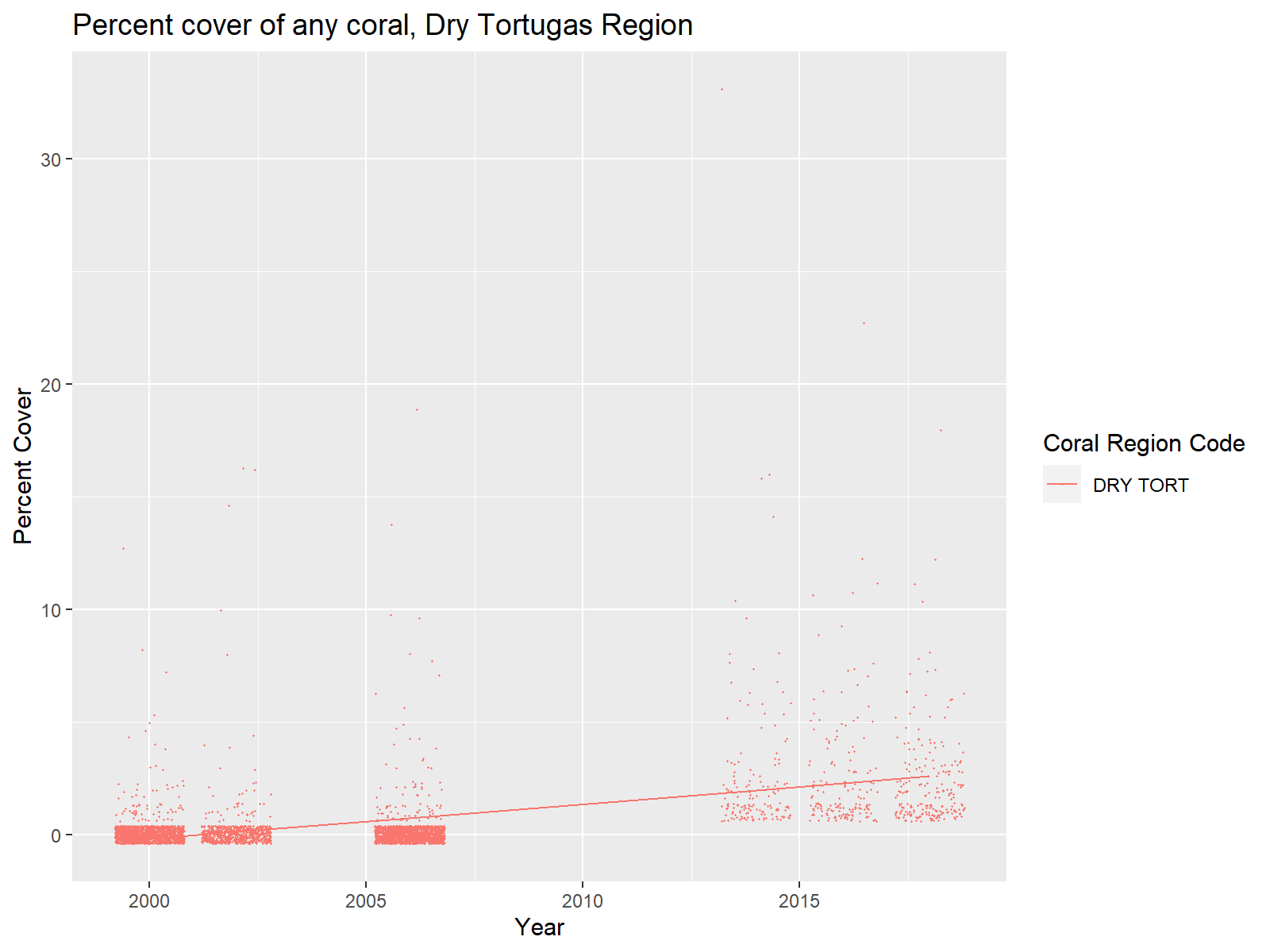
##FL Keys  
preds <- do.call(rbind, lapply(names(CorFLKEYS\_AnyCoral\_lmelist2), function(i) {  
 mod <- CorFLKEYS\_AnyCoral\_lmelist2[[i]]  
 pred <- data.frame(Coral\_Region = i, Year = mod$data$Year, pred = predict(mod, level = 0))  
}))  
#using that dataframe, plot, for each taxon, the observed values in each programlocationID  
#for each year. Plot these as 'jittered' points.   
#create a line that is the fitted model for each taxon, as well.  
Plot\_CorFLKEYS\_AnyCoral<-ggplot() +  
 geom\_jitter(data = CorFLKEYS, aes(x = Year, y = Perccov, col = Coral\_Region),size=0.1) +  
 geom\_line(data = preds, aes(x = Year, y = pred, group = Coral\_Region, col = Coral\_Region))  
#label the plot and y axis (x was already labeled above)  
Plot\_CorFLKEYS\_AnyCoral<-Plot\_CorFLKEYS\_AnyCoral + labs(y = "Percent Cover")  
Plot\_CorFLKEYS\_AnyCoral<-Plot\_CorFLKEYS\_AnyCoral + labs(title = "Percent cover of any coral, Florida Keys Region")  
#color code your points and lines by taxon  
Plot\_CorFLKEYS\_AnyCoral<-Plot\_CorFLKEYS\_AnyCoral + labs(colour = "Coral Region Code")  
Plot\_CorFLKEYS\_AnyCoral



#print to a png  
png("output/CorallmePlot\_AnyCoral\_FLKEYSRegion.png",  
 width = 8,  
 height = 4,  
 units = "in",  
 res = 200)  
print(Plot\_CorFLKEYS\_AnyCoral)  
dev.off()

png 2

##Dry Tortugas  
preds <- do.call(rbind, lapply(names(CorDRYTORTUGAS\_AnyCoral\_lmelist2), function(i) {  
 mod <- CorDRYTORTUGAS\_AnyCoral\_lmelist2[[i]]  
 pred <- data.frame(Coral\_Region = i, Year = mod$data$Year, pred = predict(mod, level = 0))  
}))  
#using that dataframe, plot, for each taxon, the observed values in each programlocationID  
#for each year. Plot these as 'jittered' points.   
#create a line that is the fitted model for each taxon, as well.  
Plot\_CorDRYTORTUGAS\_AnyCoral<-ggplot() +  
 geom\_jitter(data = CorDRYTORTUGAS, aes(x = Year, y = Perccov, col = Coral\_Region),size=0.1) +  
 geom\_line(data = preds, aes(x = Year, y = pred, group = Coral\_Region, col = Coral\_Region))  
#label the plot and y axis (x was already labeled above)  
Plot\_CorDRYTORTUGAS\_AnyCoral<-Plot\_CorDRYTORTUGAS\_AnyCoral + labs(y = "Percent Cover")  
Plot\_CorDRYTORTUGAS\_AnyCoral<-Plot\_CorDRYTORTUGAS\_AnyCoral + labs(title = "Percent cover of any coral, Dry Tortugas Region")  
#color code your points and lines by taxon  
Plot\_CorDRYTORTUGAS\_AnyCoral<-Plot\_CorDRYTORTUGAS\_AnyCoral + labs(colour = "Coral Region Code")  
Plot\_CorDRYTORTUGAS\_AnyCoral



#print to a png  
png("output/CorallmePlot\_AnyCoral\_DRYTORTUGASRegion.png",  
 width = 8,  
 height = 4,  
 units = "in",  
 res = 200)  
print(Plot\_CorDRYTORTUGAS\_AnyCoral)  
dev.off()

png 2