Impacts of global climate stressors and marine protected areas on biodiversity risk

Casey O'Hara

March 19, 2018

knitr::opts\_chunk$set(fig.width = 6, fig.height = 4, fig.path = 'Figs/',  
 echo = TRUE, message = FALSE, warning = FALSE)  
  
library(raster)  
library(kableExtra)  
  
source('https://raw.githubusercontent.com/oharac/src/master/R/common.R')  
  
dir\_git <- '~/github/spp\_health\_dists'  
  
### goal specific folders and info  
dir\_data <- file.path(dir\_git, 'data')  
dir\_spatial <- file.path(dir\_git, 'spatial')  
dir\_anx <- file.path(dir\_M, 'git-annex')  
dir\_o\_anx <- file.path(dir\_O, 'git-annex/spp\_health\_dists')  
# ### provenance tracking  
# library(provRmd); prov\_setup()  
  
### support scripts  
# source('https://raw.githubusercontent.com/oharac/src/master/R/rast\_tools.R'))   
 ### raster plotting and analyzing scripts

# Introduction

Our global oceans provide valuable social, economic, and ecological benefits for people (REF: Halpern et al 2012, 2015, etc), but no area of the ocean is untouched, directly or indirectly, by negative effects of human activity (REF: Halpern et al 2008, other CHI). Cumulative impact assessments have greatly improved our understanding of the spatial distributions of anthropogenic stressors, though most such assessments make simplifying assumptions about the relationships between impacts and pressures, such as additivity of multiple pressures (which may miss synergistic or antagonistic effects among pressures) and linearity of impacts with pressure (which may miss non-linear responses to cumulative pressures) (REF <https://www.frontiersin.org/articles/10.3389/fmars.2016.00153/full>).

Anthropogenic pressures can impact species directly through physiological harms such as fishing mortality or exposure to pollution and marine debris, or indirectly through ecosystem interactions such as food web dynamics, habitat loss, or loss of ecosystem services. Some studies have examined the impact of stressors directly on species or key ecosystem components such as marine predators (REF <https://www.nature.com/articles/ncomms3688.pdf?origin=ppub>), but to capture broader impacts on biodiversity, we must examine more inclusive assemblages of species.

Here we compare a suite of anthropogenic stressors to a global map of ecosystem health, as measured by species extinction risk relative to biodiversity, to better understand ecosystem responses to such cumulative impacts. By assessing stressors and biodiversity risk in conjunction with presence or absence of marine protected areas (MPAs), we can develop a general sense of the effectiveness of such protections in mitigating human impacts on biodiversity.

The remainder of this paper will be organized as follows: in the Methods section we will briefly outline the methodology behind the preparation of each of the data layers to be used in the regression; then we will progressively develop methods to calculate covariance matrices and standard errors for an OLS regression. In the Results section we will summarize regression coefficient estimates and standard errors calculated in a number of different ways, and calculate statistics to determine significance of each coefficient. The Discussion section examines the effects of each of the regressors in light of real-world expectations and explanations, as well as limitations of the analysis as performed here. Finally, in Next Steps, we outline potential improvements to the analysis methods, and suggest a number of variations to the regression model to explore and refine our understanding of the effects of various anthropogenic stressors on ecosystem health and biodiversity risk.

# Methods

## Preparing data layers

### Spatial distribution of biodiversity risk

For this study, we will define biodiversity risk as the average conservation status of all species within a particular location; a low score suggests a healthy ecosystem at low risk of disruption due to species extinction. This is largely based upon the methodology underlying the Ocean Health Index Biodiversity/Species goal.

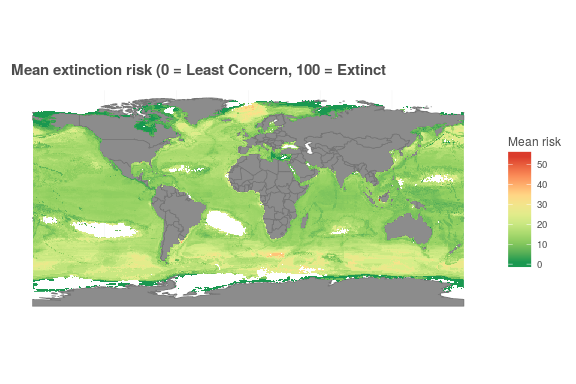
The International Union for Conservation of Nature (IUCN) Red List of Threatened Species provides the most comprehensive catalog of species conservation status, assessing the extinction risk of terrestrial species (XXX as of Version XXX) and marine species (XXX as of Version XXX) based upon quantitative criteria and expert opinion. We reclassified the six levels of extinction risk (excluding Data Deficient and Not Evaluated) to a linear 0 to 1 scale of relative risk, in which Least Concern is scored as 0, Near Threatened as 0.2, to Extinct as 1.0. This rescaling is based on the Biodiversity/Species goal of the Ocean Health Index (REF Halpern et al, 2012, 2015).

To determine spatial distribution of each species, we use AquaMaps modeled species distribution data to assign species presence to half-degree cells across the global oceans. AquaMaps determines a 'probability of occurrence' for each species/cell combination, based on the species' environmental preferences and the environmental conditions within that cell. we assign a cutoff of 0.60 probability of occurrence as the minimum threshold for 'presence' within a cell (REF AquaMaps marine mammal paper).

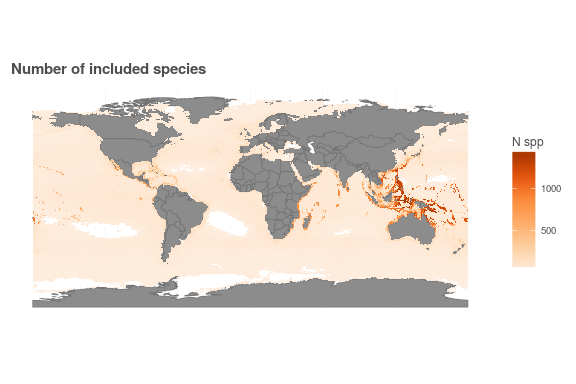
For each IUCN-assessed marine species included in the AquaMaps data set, we apply the species' risk score uniformly across all 'present' cells within its distribution. For each half-degree cell, we determine the mean species risk as simply the arithmetic mean of species risk scores for all species found within that cell. The mean risk was multiplied by a factor of 100. Cells with fewer than five assessed species were omitted from the analysis to ensure the mean risk is based upon some minimal size of species assemblage.

It is important to note that the set of marine species assessed by the IUCN Red List is only a small fraction of known marine species. This is further limited by the set of assessed species with predicted distributions in the AquaMaps dataset. The IUCN itself publishes 'extent of occurrence' maps for a subset of marine species, but the methodology differences between AquaMaps and IUCN maps make it challenging to incorporate both spatial datasets simultaneously (REF O'Hara et al 2017).

loiczid\_rast <- raster(file.path(dir\_git, 'spatial/loiczid\_raster.tif'))  
  
### gotta force the mean\_risk column to be double; there are lots of zero  
### values so will default to thinking that column is integer.  
risk\_by\_cell <- read\_csv(file.path(dir\_data, 'risk\_by\_cell\_all.csv'),  
 col\_types = 'ddddd') %>%  
 filter(n\_spp >= 5) %>%  
 mutate(mean\_risk = 100 \* mean\_risk)  
  
risk\_rast <- subs(loiczid\_rast, risk\_by\_cell, by = 'loiczid', which = 'mean\_risk') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
nspp\_rast <- subs(loiczid\_rast, risk\_by\_cell, by = 'loiczid', which = 'n\_spp') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
library(rnaturalearth)  
library(sf)  
ctrys50m <- rnaturalearth::ne\_countries(scale = 50, type = 'countries', returnclass = 'sf') %>%  
 select(iso\_a3, iso\_n3, admin)  
  
  
ggplot(risk\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 geom\_raster(aes(x, y, fill = layer)) +  
 scale\_fill\_distiller(palette = 'RdYlGn') +  
 labs(fill = 'Mean risk',  
 title = 'Mean extinction risk (0 = Least Concern, 100 = Extinct')



ggplot(nspp\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 geom\_raster(aes(x, y, fill = n\_spp)) +  
 scale\_fill\_distiller(palette = 'Oranges', direction = 1) +  
 labs(fill = 'N spp',  
 title = 'Number of included species')



### Spatial distribution of anthropogenic stressors

The most recent global cumulative impact study (Halpern et al 2015) includes fine-resolution spatial distributions of 19 anthropogenic stressors (intensity of activity or stressor) and pressures (impact of stressor on habitats, weighted by the vulnerability of the habitat to the stressor). For the purposes of this initial exploration, we use cumulative impact data layers for three climate stressors: sea surface temperature, ultraviolet radiation, and ocean acidification. Detailed methods of each data layer can be found in the supplemental information for the original studies (Halpern et al 2008, 2015) but are briefly outlined here.

#### Sea surface temperature (SST)

Weekly sea surface temperature data generated by NOAA based on satellite data provides a spatial and temporal distribution of sea surface temperature anomalies. A climatological SST mean and variance were calculated for each spatial point during the reference period of 1985-1990. For both the reference period and the study period of 2005-2010, the number of positive extreme SST events (weeks in which the local SST exceeded the climatological mean + 1 standard deviation) was tallied; the reference count was subtracted from the study period count to measure the change in frequency of extreme events.

#### Ultraviolet radiation (UV)

Similar to the SST data, Ultraviolet radiation data are generated, based on satellite data, in this case by NASA. A climatological mean and variance of UV radiation were calculated based on a reference period of (???), and then a count of positive extreme events was calculated for the study period of 2008 to 2012, in a similar manner as for SST. In the case of UV, the score was based only on the count of extreme events in the study period, rather than the difference between the study period and reference period, since the time between the reference period and study period is relatively short.

#### Ocean acidification (OA)

Ocean acidification from increases in atmospheric CO2 reduces the aragonite saturation state of ocean water. As decreases toward 1, calcifying organisms such as corals and shell-forming invertebrates are less able to create calcium carbonate structures. Below , existing shells begin to dissolve. OA scores are based on the difference between from a modeled pre-industrial value (circa 1870) and current values (2000-2009) (decreases are positive-valued). These values are then normalized by the difference between the 1870 value and a value of 1:

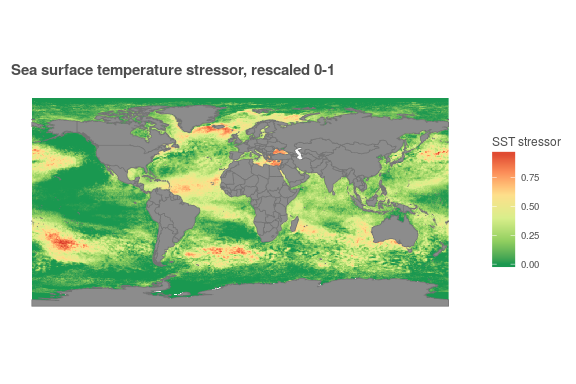
As an example, a 0.1 decrease in from 3.1 to 3.0 is expected to have far smaller impact than a 0.1 decrease from 1.2 to 1.1.

#### Rescaling and spatially transforming stressor layers

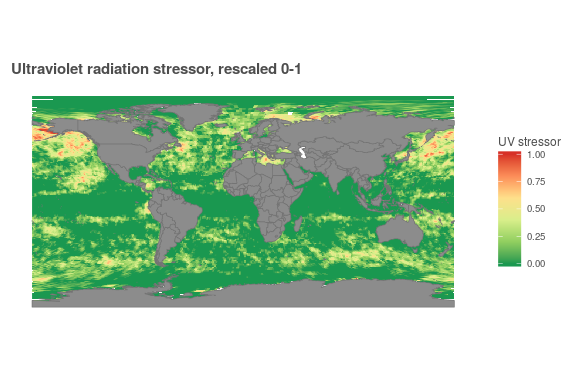
For all but one of the stressors studied in REF Halpern et al 2015, global values were highly skewed with a few instances of extremely high outliers that could tend to distort the model results. Each layer was therefore transformed using a transformation and then normalized to the 99.99th percentile log-transformed value, resulting in a score from 0 to 1. Additionally, for the OA stressor, any location with a raw was given a maximum stressor value of 1.

In the original cumulative impact assessments (REF Halpern et al 2008, 2015), each layer was spatially reprojected to a raster in Mollweide projection at approximately 1 km resolution. For the purposes of this analysis, these original layers were recalculated at a resolution to match the AquaMaps species distribution data, in a WGS84 projection at 0.5° cell resolution, using the mean to aggregate cell values from the finer resolution to the coarser resolution.

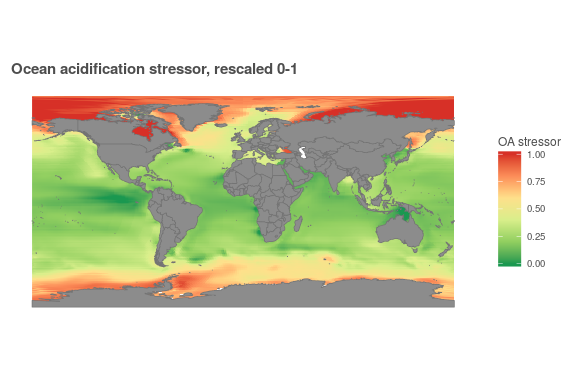
cc\_stressor\_files <- list.files('stressor\_to\_loiczid',   
 pattern = 'slr\_2016|sst\_2012|uv\_2016|oa\_2016',  
 full.names = TRUE)  
  
if(exists('cc\_stressor\_df')) rm(cc\_stressor\_df)  
for(stressor\_file in cc\_stressor\_files) { ### stressor\_file <- cc\_stressor\_files[3]  
 stressor\_name <- basename(stressor\_file) %>%  
 str\_replace('\_simple.csv', '')  
   
 # cat('Processing', stressor\_name, '...\n')  
   
 tmp <- read\_csv(stressor\_file, col\_types = 'ddddd') %>%  
 select(-n\_na) %>%  
 setNames(c('loiczid',   
 paste0(stressor\_name, '\_mean'), paste0(stressor\_name, '\_var'),   
 paste0(stressor\_name, '\_zeros')))  
   
 if(!exists('cc\_stressor\_df')) {  
 cc\_stressor\_df <- tmp ### create it the first time through the loop  
 } else {  
 cc\_stressor\_df <- cc\_stressor\_df %>%  
 full\_join(tmp, by = 'loiczid') ### join it on subsequent times through the loop  
 }  
}  
sst\_rast <- subs(loiczid\_rast, cc\_stressor\_df, by = 'loiczid', which = 'sst\_2012\_mean') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
ggplot(sst\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_raster(aes(x, y, fill = layer)) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 scale\_fill\_distiller(palette = 'RdYlGn') +  
 labs(fill = 'SST stressor',  
 title = 'Sea surface temperature stressor, rescaled 0-1')



uv\_rast <- subs(loiczid\_rast, cc\_stressor\_df, by = 'loiczid', which = 'uv\_2016\_mean') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
ggplot(uv\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_raster(aes(x, y, fill = layer)) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 scale\_fill\_distiller(palette = 'RdYlGn') +  
 labs(fill = 'UV stressor',  
 title = 'Ultraviolet radiation stressor, rescaled 0-1')



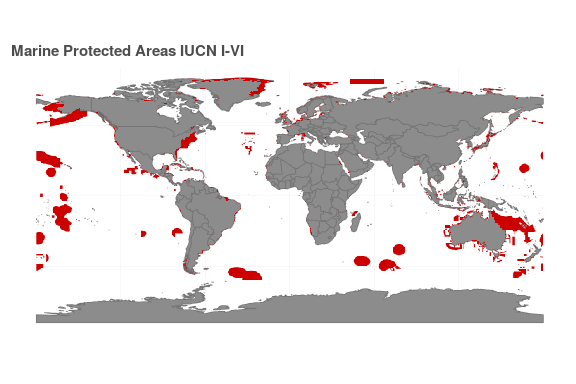
oa\_rast <- subs(loiczid\_rast, cc\_stressor\_df, by = 'loiczid', which = 'oa\_2016\_mean') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
ggplot(oa\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_raster(aes(x, y, fill = oa\_2016\_mean)) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 scale\_fill\_distiller(palette = 'RdYlGn') +  
 labs(fill = 'OA stressor',  
 title = 'Ocean acidification stressor, rescaled 0-1')



### Spatial distribution of marine protected areas

The World Database on Protected Areas (WDPA), a joint project between the United Nations Environmental Programme (UNEP) and the IUCN, offers a comprehensive global database of terrestrial and marine protected areas. For the purposes of this analysis, we include all protected areas that are officially designated, and that are categorized under the IUCN's protected area categories I through VI. Park boundaries were rasterized to the native resolution of the stressor layers (1 km Mollweide), and then aggregated back to the 0.5° cells for analysis, with the cell values representing the proportion of the cell's marine area under such protection.

mpa\_df <- read\_csv(file.path(dir\_data, 'wdpa\_i\_vi\_lookup.csv'), col\_types = 'ddd') %>%  
 left\_join(read\_csv(file.path(dir\_data, 'ocean\_area\_lookup.csv'), col\_types = 'dd')) %>%  
 group\_by(loiczid) %>%  
 summarize(mpa\_pct = sum(wdpa\_yr\_km2) / first(ocean\_area\_km2))  
  
mpa\_rast <- subs(loiczid\_rast, mpa\_df, by = 'loiczid', which = 'mpa\_pct') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
ggplot(mpa\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 geom\_raster(aes(x, y, alpha = mpa\_pct), fill = 'red3', show.legend = FALSE) +  
 labs(title = 'Marine Protected Areas IUCN I-VI')



## Developing the regression methodology

In this section of the Methods, we will focus on the development of the regression methodology rather than simply the final regression method used in the analysis, as the development seems more relevant for the purposes of this class project. Each step along the way we'll build on the basic OLS regression and show some intermediate results, to see how these results change as we account for heteroskedasticity-robust standard errors and cluster-robust standard errors.

To examine the effects of climate stressors on the mean extinction risk of species in a particular location, we will use OLS estimation to estimate the projection coefficients on the regressors. The regressors we will examine include:

* Ultraviolet radiation:
* Sea surface temperature:
* Ocean acidification:
* Marine protected areas:
* abs(Latitude): : Climate preferences of marine species typically include strong preferences for temperature, which are dependent on distance from equator/toward poles. Including absolute latitude roughly allows for changes in mean risk due to changes in species composition.
* Latitude < 0 (dummy): : Including a southern latitude dummy variable allows for asymmetric effects of latitude (i.e. northern and southern hemisphere) on mean risk and interactions with covariates.
* Constant term

The response variable is the mean extinction risk of all species within a particular spatial cell, i.e. biodiversity risk.

To be able to run the regression manually, we eliminate any cells with NA values in , , , and/or . This filter, combined with filtering for , leaves 143,890 observations out of 159,535 total observations.

### Regression

Defining as the matrix of regressors noted above, and as our response variable, we can estimate our regression coefficients :

Without considering spatial autocorrelation, we calculate residuals from

cc\_stressor\_files <- list.files('stressor\_to\_loiczid',   
 pattern = 'slr|sst|uv|oa',  
 full.names = TRUE)  
  
if(exists('cc\_stressor\_df')) rm(cc\_stressor\_df)  
for(stressor\_file in cc\_stressor\_files) { ### stressor\_file <- cc\_stressor\_files[3]  
 stressor\_name <- basename(stressor\_file) %>%  
 str\_replace('\_simple.csv', '')  
   
 # cat('Processing', stressor\_name, '...\n')  
   
 tmp <- read\_csv(stressor\_file, col\_types = 'ddddd') %>%  
 select(-n\_na) %>%  
 setNames(c('loiczid',   
 paste0(stressor\_name, '\_mean'), paste0(stressor\_name, '\_var'),   
 paste0(stressor\_name, '\_zeros')))  
   
 if(!exists('cc\_stressor\_df')) {  
 cc\_stressor\_df <- tmp ### create it the first time through the loop  
 } else {  
 cc\_stressor\_df <- cc\_stressor\_df %>%  
 full\_join(tmp, by = 'loiczid') ### join it on subsequent times through the loop  
 }  
}  
  
risk\_df <- read\_csv(file.path(dir\_data, 'risk\_by\_cell\_all.csv'),  
 col\_types = 'ddddd')  
  
lat\_df <- read\_csv(file.path(dir\_data, 'latlong\_lookup.csv'), col\_types = 'ddd') %>%  
 select(loiczid, lat) %>%  
 mutate(dummySouth = as.integer(lat < 0),  
 latAbs = abs(lat))  
  
mpa\_df <- read\_csv(file.path(dir\_data, 'wdpa\_i\_vi\_lookup.csv'), col\_types = 'ddd') %>%  
 left\_join(read\_csv(file.path(dir\_data, 'ocean\_area\_lookup.csv'), col\_types = 'dd')) %>%  
 group\_by(loiczid) %>%  
 summarize(mpa\_pct = sum(wdpa\_yr\_km2) / first(ocean\_area\_km2))  
  
  
risk\_v\_stressor\_df <- full\_join(risk\_df, cc\_stressor\_df, by = 'loiczid') %>%  
 left\_join(lat\_df, by = 'loiczid') %>%  
 filter(n\_spp >= 5) %>%  
 select(loiczid,   
 mean\_risk,  
 # log\_mean\_risk,   
 sst\_2012\_mean, oa\_2016\_mean, uv\_2016\_mean,   
 latAbs, dummySouth) %>%  
 # filter(!is.na(log\_mean\_risk)) %>%  
 filter(!is.na(mean\_risk)) %>%  
 filter(!is.na(sst\_2012\_mean)) %>%  
 filter(!is.na(oa\_2016\_mean)) %>%  
 filter(!is.na(uv\_2016\_mean)) %>%  
 left\_join(mpa\_df, by = 'loiczid') %>%  
 mutate(mpa\_pct = ifelse(is.na(mpa\_pct), 0, mpa\_pct),  
 mean\_risk = 100 \* mean\_risk)

y <- risk\_v\_stressor\_df$mean\_risk  
# y <- risk\_v\_stressor\_df$log\_mean\_risk  
loiczid\_vec <- risk\_v\_stressor\_df$loiczid  
  
n\_obs <- length(y)  
  
X <- risk\_v\_stressor\_df %>%  
 mutate(oa\_S = oa\_2016\_mean \* dummySouth,  
 sst\_S = sst\_2012\_mean \* dummySouth,  
 uv\_S = uv\_2016\_mean \* dummySouth,  
 latS = latAbs \* dummySouth,  
 const = 1) %>%  
 select(-loiczid, -mean\_risk) %>%  
 as.matrix()  
  
Xt\_X <- t(X) %\*% X  
Xt\_y <- t(X) %\*% y  
  
beta\_hat <- solve(Xt\_X) %\*% Xt\_y ### why 'solve' for inverse? jeez, R, be intuitive

### Estimation of error variance

Since is not directly observable, we can approximate the error variance

as

e\_hat <- y - X %\*% beta\_hat  
  
sig\_hat <- 1/n\_obs \* t(e\_hat) %\*% e\_hat

The estimate of error variance = 24.57368.

y\_bar <- mean(y)  
  
y\_hat <- X %\*% beta\_hat  
term1 <- sum((y - y\_bar)^2)  
  
term2 <- sum((y\_hat - y\_bar)^2)  
  
term3 <- sum(e\_hat^2)  
  
R\_2 <- term2 / term1

The coefficient of determination, = 0.3054.

### Prediction errors

Estimates to this point have been based on the basic OLS methodology, and as such our residuals are constructed based on the entire data set, including the point we are trying to predict. To get at a true prediction, we drop the value to be predicted to determine the leave-one-out OLS estimator.

and out-of-sample mean squared error:

We can calculate these using projection matrix but since we are using a dataset with , it seems unlikely that we will be able to work with these matrices... Instead we will directly calculate the diagonal values of , , to understand the leverage of each observation:

# P = X %\*% solve(t(X) %\*% X) %\*% t(X)   
# Error: cannot allocate vector of size 138.6 Gb  
  
h\_ii <- vector('numeric', length = n\_obs)  
for (i in seq\_along(y)) {  
   
 x\_i <- X[i, ]  
 h\_ii[i] <- t(x\_i) %\*% solve(Xt\_X) %\*% x\_i  
   
}  
# hist(h\_ii, main = 'leverage h\_ii')

e\_tilde <- (1 - h\_ii)^-1 \* e\_hat  
  
sig\_tilde <- 1/n\_obs \* sum(e\_tilde^2)

We calculated a value of = 24.57837, very close to = 24.573681, as would be expected with such a large . Removing one out of nearly 144,000 observations means the impact of that one observation will be very small, especially since the error terms are all within a close neighborhood of zero (+/- 0.5).

### Variance of least-squares estimator (homoskedastic)

If we can assume i.i.d. sampling and a homoskedastic linear regression model, we can calculate our covariance matrix estimator as

Our estimator is a biased estimator for , but since it should be very close; we can correct it using a scaling factor of . Our estimator from the leave-one-out OLS regression should be unbiased and need no adjustment. We can calculate covariance matrix estimators for each and determine standard errors for our regression coefficients from .

k\_regr <- length(beta\_hat)  
  
V\_hat1 <- (n\_obs)/(n\_obs - k\_regr) \* sig\_hat[1] \* solve(Xt\_X)  
V\_tilde1 <- sig\_tilde[1] \* solve(Xt\_X)  
  
se\_hat1 <- sqrt(diag(V\_hat1))  
se\_tilde1 <- sqrt(diag(V\_tilde1))  
  
beta\_se\_df <- data.frame(estimate = rownames(beta\_hat),  
 beta\_hat,  
 se\_hat = se\_hat1,  
 se\_tilde = se\_tilde1) %>%  
 mutate(t\_hat = round(beta\_hat / se\_hat, 2),  
 t\_tilde = round(beta\_hat / se\_tilde, 2),  
 se\_hat = round(se\_hat, 5),  
 se\_tilde = round(se\_tilde, 5))  
  
knitr::kable(beta\_se\_df, format = 'html', caption = 'Table i: Standard errors and t statistics',  
 col.names = c('Estimate', '$\\widehat\\beta$',   
 '$\\widehat\\sigma$',  
 '$\\widetilde\\sigma\_{loo}$',  
 '$\\hat t$',  
 '$\\widetilde t\_{loo}$')) %>%  
 kable\_styling('striped', full\_width = FALSE)

Table i: Standard errors and t statistics

Estimate

sst\_2012\_mean

4.6592640

0.10776

0.10776

43.24

43.24

oa\_2016\_mean

-16.9142263

0.15605

0.15606

-108.39

-108.38

uv\_2016\_mean

3.2213246

0.11916

0.11916

27.03

27.03

latAbs

0.1298950

0.00175

0.00175

74.12

74.12

dummySouth

-0.2912758

0.06475

0.06475

-4.50

-4.50

mpa\_pct

-1.9030904

0.06207

0.06207

-30.66

-30.66

oa\_S

-4.1289479

0.24910

0.24912

-16.58

-16.57

sst\_S

-2.5529086

0.14522

0.14523

-17.58

-17.58

uv\_S

-2.1906827

0.19667

0.19668

-11.14

-11.14

latS

0.1400159

0.00243

0.00243

57.53

57.53

const

12.2651278

0.04457

0.04458

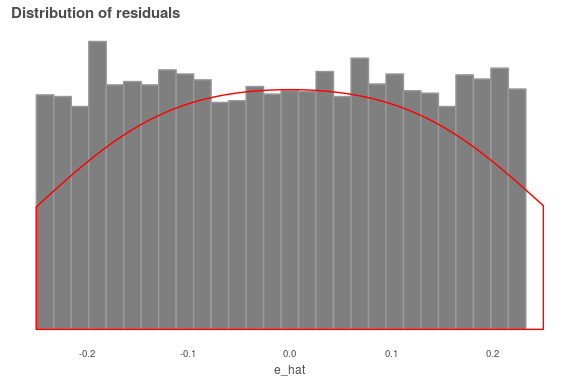
275.17

275.15

### Beyond basic OLS regression

We expect our residuals to be i.i.d. and normally distributed with mean 0, but these assumptions do not appear to be entirely valid.

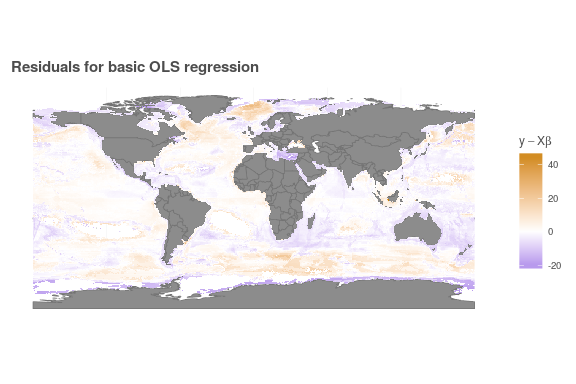
e\_hat\_df <- data.frame(e\_hat = e\_hat)  
ggplot(e\_hat\_df, aes(x = e\_hat, y = ..density..)) +  
 ggtheme\_plot() +  
 geom\_histogram(color = 'grey60', fill = 'grey50') +  
 geom\_density(adjust = 5, color = 'red') +  
 xlim(c(-.25, .25)) +  
 theme(axis.text.y = element\_blank(),  
 axis.title.y = element\_blank(),  
 panel.grid.major = element\_blank()) +  
 labs(title = 'Distribution of residuals')



ehat\_mean <- sum(e\_hat)/n\_obs  
# 1.708902e-12  
  
### calc skewness and kurtosis  
m2 <- sum((e\_hat - ehat\_mean)^2) / n\_obs  
m3 <- sum((e\_hat - ehat\_mean)^3) / n\_obs  
m4 <- sum((e\_hat - ehat\_mean)^4) / n\_obs  
  
skewness = m3 / (m2^(3/2))  
kurtosis = m4 / (m2^2)

While the residuals appear to be roughly normally distributed based on the histogram, we can calculate skewness = 0.0438 and kurtosis = 5.6861; a perfect normal distribution would have skewness = 0 and kurtosis = 3. High kurtosis indicates that the true variance will be proportionally that much greater (in this case, 5.6861 times greater) than the covariance matrix calculated under the homoskedastic assumption. In addition, plotting the residuals spatially shows significant spatial heterogeneity, so we will need to consider cluster- and heteroskedasticity-robust methods of estimating our standard errors.

ehat\_df <- data.frame(loiczid\_vec,  
 e\_hat)  
e\_hat\_rast <- loiczid\_rast %>%  
 subs(ehat\_df, by = 'loiczid\_vec', which = 'e\_hat') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
ggplot(e\_hat\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 geom\_raster(aes(x, y, fill = e\_hat)) +  
 scale\_fill\_gradient2(low = 'blue3', high = 'orange3') +  
 labs(title = 'Residuals for basic OLS regression',  
 fill = expression(y - X \* beta))



### Heteroskedasticity-robust standard errors

Earlier we calculated an estimate of error variance:

From Hansen, equation 4.24: $var(\widehat{e}\_i | \mathbf{X}) = \mathbb E [\widehat e\_i^2 | \mathbf{X}] = (1 - h\_{ii}) \sigma^2$:

As this variance is a function of and hence , the residuals are heteroskedastic even if the errors are homoskedastic. Notice as well that this implies is a biased estimator of .

We can address the bias as in equation 4.30:

### Covariance matrix estimation under heteroskedasticity

Hansen presents several good candidates for determining heteroskedasticity-robust estimators of the covariance matrix; here we will calculate two versions:

and

The degree-of-freedom-adjusted White covariance matrix is common (it is the default in Stata), but the leave-one-out covariance matrix estimator derived by Andrews, , is a conservative estimate and therefore may be a more easily defensible choice.

k\_regr <- ncol(X)  
  
ehat\_terms1 <- vector('list', length = n\_obs)  
ehat\_terms2 <- ehat\_terms1  
  
for(i in 1:n\_obs) { ### i <- 1  
 x\_i <- X[i, ]  
 ehat\_terms1[[i]] <- x\_i %\*% t(x\_i) \* (e\_hat[i])^2  
  
 ehat\_terms2[[i]] <- ehat\_terms1[[i]] / (1 - h\_ii[i])^2  
}  
  
sum\_ehats\_1 <- Reduce('+', ehat\_terms1)  
sum\_ehats\_2 <- Reduce('+', ehat\_terms2)  
  
V\_white <- (n\_obs/(n\_obs - k\_regr)) \* solve(Xt\_X) %\*% sum\_ehats\_1 %\*% solve(Xt\_X)  
  
V\_andrews <- solve(Xt\_X) %\*% sum\_ehats\_2 %\*% solve(Xt\_X)  
  
### max(abs(V\_hat - V\_tilde)) # shows that the biggest diff is only 4.10777e-07

These estimator matrices estimate the variance of the distribution of , and taking the square root of the diagonals, we can calculate the standard error terms:

Calculating for both adjusted White and Andrews robust standard errors, and comparing to the leave-one-out homoskedastic standard error () calculated earlier:

se\_white <- diag(V\_white) %>% sqrt()  
se\_andrews <- diag(V\_andrews) %>% sqrt()  
  
beta\_se\_df2 <- data.frame(estimate = rownames(beta\_hat),  
 beta\_hat,  
 se\_tilde = se\_tilde1,  
 se\_white,  
 se\_andrews) %>%  
 mutate(t\_tilde = round(beta\_hat / se\_tilde, 2),  
 t\_white = round(beta\_hat / se\_white, 2),  
 t\_andrews = round(beta\_hat / se\_andrews, 2),  
 se\_tilde = round(se\_tilde, 5),  
 se\_white = round(se\_white, 5),  
 se\_andrews = round(se\_andrews, 5))  
  
knitr::kable(beta\_se\_df2, format = 'html',   
 caption = 'Table ii: Standard errors and t statistics',  
 col.names = c('Estimate', '$\\widehat\\beta$',   
 '$\\widetilde\\sigma\_{loo}$',  
 '$\\widehat\\sigma\_{W}$',  
 '$\\widetilde\\sigma\_{A}$',  
 '$t\_{loo}$',  
 '$t\_{W}$',  
 '$t\_{A}$')) %>%  
 kable\_styling('striped', full\_width = FALSE)

Table ii: Standard errors and t statistics

Estimate

sst\_2012\_mean

4.6592640

0.10776

0.10899

0.10901

43.24

42.75

42.74

oa\_2016\_mean

-16.9142263

0.15606

0.19452

0.19456

-108.38

-86.95

-86.93

uv\_2016\_mean

3.2213246

0.11916

0.14262

0.14264

27.03

22.59

22.58

latAbs

0.1298950

0.00175

0.00240

0.00240

74.12

54.18

54.17

dummySouth

-0.2912758

0.06475

0.05639

0.05639

-4.50

-5.17

-5.17

mpa\_pct

-1.9030904

0.06207

0.05766

0.05767

-30.66

-33.01

-33.00

oa\_S

-4.1289479

0.24912

0.33336

0.33340

-16.57

-12.39

-12.38

sst\_S

-2.5529086

0.14523

0.13747

0.13748

-17.58

-18.57

-18.57

uv\_S

-2.1906827

0.19668

0.22310

0.22313

-11.14

-9.82

-9.82

latS

0.1400159

0.00243

0.00291

0.00291

57.53

48.15

48.15

const

12.2651278

0.04458

0.03506

0.03507

275.15

349.79

349.76

### Measures of Fit (4.17)

Previously we calculated for our OLS estimators under homoskedasticity:

But this can be improved upon. R2 always increases as we add regressors, so should be taken with a grain of salt... so we can look at *adjusted* R2, which partially corrects this problem:

or this one (well-adjusted?), which fully corrects this issue:

### Regular R^2 (repeated from above)  
y\_bar <- mean(y)  
  
y\_hat <- X %\*% beta\_hat  
  
R2 <- 1 - sum(e\_hat^2) / sum((y - y\_bar)^2)  
### .310315  
  
### Adjusted R^2  
adj\_R2 <- 1 - ((n\_obs - 1) \* sum(e\_hat^2)) / ((n\_obs - k\_regr) \* sum((y - y\_bar)^2))  
### .310275  
  
### Tilde R^2  
e\_tilde <- (1 - h\_ii)^-1 \* e\_hat  
  
tilde\_R2 <- 1 - sum(e\_tilde^2) / sum((y - y\_bar)^2)  
  
r2\_df <- data.frame('fit' = c('Std $R^2$', 'Adj $R^2$', 'Well-adj $R^2$'),  
 'value' = c(R2, adj\_R2, tilde\_R2))  
  
knitr::kable(r2\_df, format = 'html', caption = 'Table iii: Measures of fit',  
 col.names = c('Measure of fit', 'value')) %>%  
 kable\_styling('striped', full\_width = FALSE)

Table iii: Measures of fit

Measure of fit

value

Std

0.3054215

Adj

0.3053733

Well-adj

0.3052890

### Cluster-robust standard errors

From the map of standard errors above, it is clear that there is spatial heterogeneity in the distribution of errors. This may be due to spatial autocorrelation or some other spatially-distributed variable that is not accounted for in the regression. One way to deal with spatial autocorrelation is to calculate a spatial weights matrix, in which one cell is affected by neighboring cells based on some relationship, for example inverse distance. However, in this case the resulting matrix would contain on the order of 60 billion cells. Instead we will look at spatial clustering to develop a more robust understanding of the variance in our estimators.

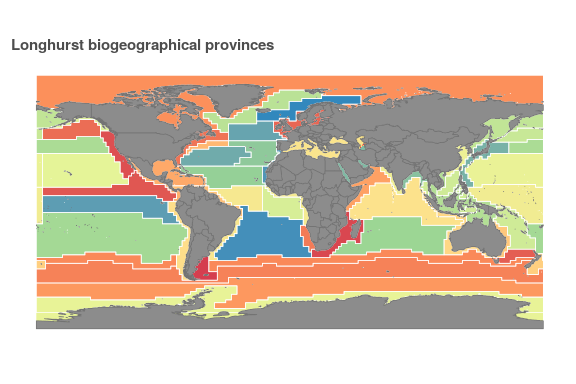
From Hansen:

The standard clustering assumption is that the clusters are known to the researcher and that the observations are independent across clusters.

Spatial clustering sets up an assumption that there is dependence among cells within a particular cluster - for example, cells within one cluster may share similar physical characteristics of depth, salinity, sea ice, etc. Conditions between two clusters, however, are assumed to be independent. This may be problematic since our spatial clusters will set up hard boundaries between neighboring clusters, while the physical reality is that there is no fixed boundary between such marine regions.

Some research efforts have roughly defined biogeographical boundaries within oceans, such as [Longhurst biogeographical provinces](http://www.marineregions.org/gazetteer.php?p=image&pic=64934), which may provide a meaningful starting point. However, these regions result in clusters with very different numbers of observations, which can be problematic for calculating cluster-robust errors. Additionally, there are only 54 Longhurst provinces; we may want to break the map into a larger number of more spatially compact clusters.

longh\_random <- data.frame(old = 1:54,  
 new = sample(1:54, size = 54, replace = FALSE))  
  
longh\_df <- read\_csv(file.path(dir\_git, 'data/longhurst\_cells.csv'),  
 col\_types = 'idcc') %>%  
 select(loiczid, old = longhurst) %>%  
 left\_join(longh\_random, by = 'old')  
  
if(!exists('loiczid\_rast')) {  
 loiczid\_rast <- raster(file.path(dir\_git, 'spatial/loiczid\_raster.tif'))  
}  
  
longhurst\_rast <- raster::subs(loiczid\_rast, longh\_df,  
 by = 'loiczid', which = 'new') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
longhurst\_sf <- read\_sf(file.path(dir\_spatial, 'longhurst\_v4\_2010/Longhurst\_world\_v4\_2010.shp'))  
  
ggplot(longhurst\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_raster(aes(x, y, fill = layer), show.legend = FALSE) +  
 geom\_sf(data = longhurst\_sf, fill = NA, color = 'white', size = .25, show.legend = FALSE) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 scale\_fill\_distiller(palette = 'Spectral') +  
 labs(title = 'Longhurst biogeographical provinces',  
 fill = expression(y - X \* beta))



longh\_df <- read\_csv(file.path(dir\_git, 'data/longhurst\_cells.csv'),  
 col\_types = 'idcc') %>%  
 select(longhurst, loiczid)  
  
risk\_clusters <- risk\_v\_stressor\_df %>%  
 left\_join(longh\_df, by = 'loiczid') %>%  
 arrange(loiczid) %>%  
 fill(longhurst, .direction = 'up') %>% ### Do a VERY rough interpolation  
 group\_by(longhurst) %>%  
 ungroup() %>%  
 arrange(longhurst, loiczid)

Clustering our observations by Longhurst biogeographical provinces, we can determine a clustered robust covariance matrix estimator analogous to the robust adjusted White estimator:

where

and

### Reset vars to make sure they're in appropriate cluster order  
  
y <- risk\_clusters$mean\_risk  
n\_obs <- length(y)  
clust\_vec <- risk\_clusters$longhurst  
clust\_ids <- unique(clust\_vec)  
  
X <- risk\_clusters %>%  
 mutate(oa\_S = oa\_2016\_mean \* dummySouth,  
 sst\_S = sst\_2012\_mean \* dummySouth,  
 uv\_S = uv\_2016\_mean \* dummySouth,  
 latS = latAbs \* dummySouth,  
 const = 1) %>%  
 select(-loiczid, -mean\_risk, -longhurst) %>%  
 as.matrix()  
  
### Calculate Beta\_hat, though should be identical to original beta\_hat  
Xt\_X <- solve(t(X) %\*% X)  
Xt\_y <- t(X) %\*% y  
beta\_hat\_clust <- Xt\_X %\*% Xt\_y  
k\_regr <- length(beta\_hat\_clust)   
  
### Calculate Omega\_n: remove g obs and calc beta hat; then calc  
### e\_tilde\_g for each group (store in a list object)  
G <- length(clust\_ids)  
omega\_g <- vector('list', length = G)  
for (g in clust\_ids) { ### g <- 1  
  
 y\_g <- y[clust\_vec == g]  
 X\_g <- X[clust\_vec == g, ]  
   
 e\_hat\_g <- y\_g - X\_g %\*% beta\_hat\_clust  
   
 omega\_g[[g]] <- t(X\_g) %\*% e\_hat\_g %\*% t(e\_hat\_g) %\*% X\_g  
}  
  
Omega\_n <- Reduce('+', omega\_g)  
  
a\_n <- (n\_obs - 1) / (n\_obs - k\_regr) \* G / (G - 1)  
  
V\_hat\_clust <- a\_n \* Xt\_X %\*% Omega\_n %\*% Xt\_X  
  
se\_hat\_clust <- sqrt(diag(V\_hat\_clust))

We can also calculate an alternative cluster-robust covariance matrix estimator based on cluster-level prediction errors and a leave-one-cluster-out type of process, as detailed in Hansen:

where

Previously we calculated values using leverage values: . Can we calculate leverage values at the group level and do the same? In the mean time we will use the brute-force approach and simply calculate for each group separately.

### Use X, y, beta, etc from above.  
  
### Calculate Beta\_hat (-g): remove g obs and calc beta hat; then calc  
### e\_tilde\_g for each group (store in a list object)  
e\_tilde\_g <- vector('list', length = length(clust\_ids))  
for (g in clust\_ids) { ### g <- 1  
 X\_minusg <- X[clust\_vec != g, ]  
 y\_minusg <- y[clust\_vec != g]  
 Xt\_X\_minusg <- t(X\_minusg) %\*% X\_minusg  
 Xt\_y\_minusg <- t(X\_minusg) %\*% y\_minusg  
  
 beta\_hat\_minusg <- solve(Xt\_X\_minusg) %\*% Xt\_y\_minusg  
   
 y\_g <- y[clust\_vec == g]  
 X\_g <- X[clust\_vec == g, ]  
   
 e\_tilde\_g[[g]] <- y\_g - X\_g %\*% beta\_hat\_minusg  
}  
  
Gsum\_terms <- vector('list', length = length(clust\_ids))  
  
for (g in clust\_ids) { ### g <- 1  
 X\_g <- X[clust\_vec == g, ]  
 e\_g <- e\_tilde\_g[[g]]  
   
 Gsum\_terms[[g]] <- t(X\_g) %\*% e\_g %\*% t(e\_g) %\*% X\_g  
}  
  
Gsum <- Reduce('+', Gsum\_terms)  
  
V\_tilde\_clust <- Xt\_X %\*% Gsum %\*% Xt\_X  
  
se\_tilde\_clust <- sqrt(diag(V\_tilde\_clust))

### Cluster robust standard errors

Calculating standard errors from the square root of the diagonal of the covariance matrix estimators based upon a simple Longhurst biogeographical province clustering:

Here we compare the standard error terms calculated using unclustered Andrews covariance matrix estimator , as well as standard and alternative (leave-one-out) clustered covariance matrix estimators and .

beta\_se\_df3 <- data.frame(estimate = rownames(beta\_hat),  
 beta\_hat,  
 se\_andrews,  
 se\_hat\_clust,  
 se\_tilde\_clust) %>%  
 mutate(t\_andrews = round(beta\_hat / se\_andrews, 2),  
 t\_hat\_clust = round(beta\_hat / se\_hat\_clust, 2),  
 t\_tilde\_clust = round(beta\_hat / se\_tilde\_clust, 2),  
 se\_andrews = round(se\_andrews, 5),  
 se\_hat\_clust = round(se\_hat\_clust, 5),  
 se\_tilde\_clust = round(se\_tilde\_clust, 5))  
  
knitr::kable(beta\_se\_df3, format = 'html',   
 caption = 'Table iv: Standard errors and t statistics',  
 col.names = c('Estimate', '$\\widehat\\beta$',   
 '$\\widetilde\\sigma\_{A}$',  
 '$\\widehat\\sigma\_{clust}$',  
 '$\\widetilde\\sigma\_{alt.clust}$',  
 '$t\_{A}$',  
 '$t\_{clust}$',  
 '$t\_{alt.clust}$')) %>%  
 kable\_styling('striped', full\_width = FALSE)

Table iv: Standard errors and t statistics

Estimate

sst\_2012\_mean

4.6592640

0.10901

2.35151

3.50407

42.74

1.98

1.33

oa\_2016\_mean

-16.9142263

0.19456

2.59175

11.54064

-86.93

-6.53

-1.47

uv\_2016\_mean

3.2213246

0.14264

2.60237

5.20329

22.58

1.24

0.62

latAbs

0.1298950

0.00240

0.03527

0.03445

54.17

3.68

3.77

dummySouth

-0.2912758

0.05639

1.76753

2.85586

-5.17

-0.16

-0.10

mpa\_pct

-1.9030904

0.05767

0.60285

0.69089

-33.00

-3.16

-2.75

oa\_S

-4.1289479

0.33340

11.04152

20.55364

-12.38

-0.37

-0.20

sst\_S

-2.5529086

0.13748

2.79787

4.16877

-18.57

-0.91

-0.61

uv\_S

-2.1906827

0.22313

4.23527

7.59171

-9.82

-0.52

-0.29

latS

0.1400159

0.00291

0.05015

0.07026

48.15

2.79

1.99

const

12.2651278

0.03507

0.79642

1.62047

349.76

15.40

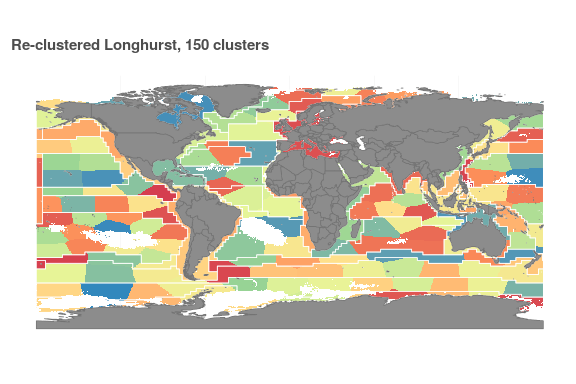
7.57

#### Effect of cluster number and size on standard errors

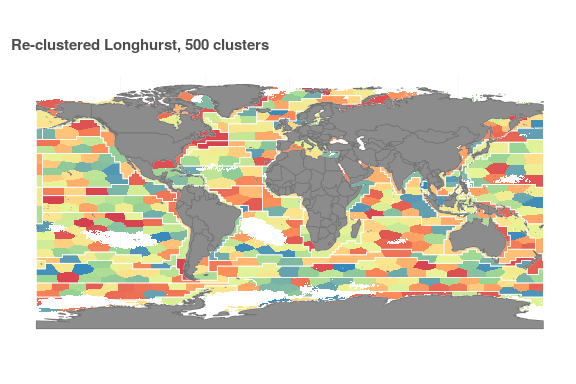
Hansen notes in section 4.21 that the number of clusters in a cluster-robust approach is analogous to the number of observations in a non-clustered approach, and also that cluster sizes are generally assumed to be homogeneous. In clustering on Longhurst provinces, we limit ourselves to only 54 clusters, which vary greatly in the number of in-cluster observations. For this final modification we will re-cluster our data using a K-means approach to create spatially compact clusters informed by the Longhurst boundaries, breaking apart large provinces and consolidating tiny provinces.

Here we will use the alternative cluster calculation (leave-one-cluster-out approach) and iterate over several analysis at different numbers of clusters to see the effect of increasing numbers of smaller clusters on the results.

latlong\_df <- read\_csv(file.path(dir\_data, 'latlong\_lookup.csv'), col\_types = 'ddd')  
  
cluster\_df <- risk\_clusters %>%  
 select(loiczid, longhurst) %>%  
 left\_join(latlong\_df, by = 'loiczid')  
   
set.seed(1234)  
  
### define helper function  
rescale <- function(x) {(x - min(x)) / (max(x) - min(x))}  
  
longh\_weight <- 2  
n\_clusts <- c(100, 150, 250, 500)  
n\_start <- 25  
  
### slow process; save results for future use (and consistency)  
clust\_id\_file <- file.path(dir\_data, 'longh\_clust\_ids.csv')  
  
if(!file.exists(clust\_id\_file)) {  
 cluster\_norm\_df <- cluster\_df %>%  
 select(-loiczid) %>% ### not spatially relevant  
 mutate(lat\_norm = rescale(lat),  
 long\_norm = rescale(long),  
 longh\_norm = rescale(longhurst),  
 longh\_norm = longh\_weight \* longh\_norm) %>%  
 select(-lat, -long, -longhurst)  
   
 set.seed(1234)  
 cluster\_ids\_df <- cluster\_df %>%  
 select(loiczid)  
   
 for(n\_clust in n\_clusts) {  
 fit\_km <- kmeans(cluster\_norm\_df,  
 n\_clust,   
 nstart = 25)  
 clust\_id <- data.frame(fit\_km$cluster) %>%  
 setNames(paste0('clust\_', n\_clust))  
 cluster\_ids\_df <- bind\_cols(cluster\_ids\_df, clust\_id)  
 }  
   
 write\_csv(cluster\_ids\_df, clust\_id\_file)  
} else {  
 cluster\_ids\_df <- read\_csv(clust\_id\_file)  
}  
   
clust\_rast <- raster::subs(loiczid\_rast, cluster\_ids\_df,   
 by = 'loiczid', which = 'clust\_150') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
ggplot(clust\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_raster(aes(x, y, fill = layer), show.legend = FALSE) +  
 geom\_sf(data = longhurst\_sf, fill = NA, color = 'white', size = .25, show.legend = FALSE) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 scale\_fill\_distiller(palette = 'Spectral') +  
 labs(title = 'Re-clustered Longhurst, 150 clusters',  
 fill = expression(y - X \* beta))



clust\_rast <- raster::subs(loiczid\_rast, cluster\_ids\_df,   
 by = 'loiczid', which = 'clust\_500') %>%  
 rasterToPoints() %>%  
 as.data.frame()  
  
ggplot(clust\_rast) +  
 ggtheme\_plot() +  
 theme(axis.title = element\_blank(),  
 axis.text = element\_blank()) +  
 geom\_raster(aes(x, y, fill = layer), show.legend = FALSE) +  
 geom\_sf(data = longhurst\_sf, fill = NA, color = 'white', size = .25, show.legend = FALSE) +  
 geom\_sf(data = ctrys50m, fill = 'grey55', color = 'grey45', size = .25) +  
 scale\_fill\_distiller(palette = 'Spectral') +  
 labs(title = 'Re-clustered Longhurst, 500 clusters',  
 fill = expression(y - X \* beta))



cluster\_df <- risk\_clusters %>%  
 left\_join(cluster\_ids\_df, by = 'loiczid')  
  
y <- cluster\_df$mean\_risk  
n\_obs <- length(y)  
  
X <- cluster\_df %>%  
 mutate(oa\_S = oa\_2016\_mean \* dummySouth,  
 sst\_S = sst\_2012\_mean \* dummySouth,  
 uv\_S = uv\_2016\_mean \* dummySouth,  
 latS = latAbs \* dummySouth,  
 const = 1) %>%  
 select(sst\_2012\_mean, oa\_2016\_mean, uv\_2016\_mean,   
 latAbs, dummySouth, mpa\_pct, oa\_S, sst\_S, uv\_S, latS, const) %>%  
 as.matrix()  
  
### Calculate Beta\_hat, though should be identical to original beta\_hat  
Xt\_X <- solve(t(X) %\*% X)  
Xt\_y <- t(X) %\*% y  
beta\_hat\_kmeans <- Xt\_X %\*% Xt\_y  
  
### Calculate Beta\_hat (-g): remove g obs and calc beta hat; then calc  
### e\_tilde\_g for each group (store in a list object)  
  
clust\_ids <- cluster\_df %>%  
 select(starts\_with('clust')) %>%  
 as.matrix  
  
se\_tilde\_kmeans <- vector('list', ncol(clust\_ids)) %>%  
 setNames(paste0('clust\_', n\_clusts))  
  
for(i in 1:ncol(clust\_ids)) { ### i <- 1  
 clust\_vec <- clust\_ids[ , i]  
 clust\_id\_vec <- unique(clust\_vec) %>% sort()  
   
 e\_tilde\_g <- vector('list', length = length(clust\_id\_vec))  
   
 for (g in clust\_id\_vec) { ### g <- 3  
 X\_minusg <- X[clust\_vec != g, ]  
 y\_minusg <- y[clust\_vec != g]  
 Xt\_X\_minusg <- t(X\_minusg) %\*% X\_minusg  
 Xt\_y\_minusg <- t(X\_minusg) %\*% y\_minusg  
   
 beta\_hat\_minusg <- solve(Xt\_X\_minusg) %\*% Xt\_y\_minusg  
   
 y\_g <- y[clust\_vec == g]  
 X\_g <- X[clust\_vec == g, ]  
   
 e\_tilde\_g[[g]] <- y\_g - X\_g %\*% beta\_hat\_minusg  
 }  
   
   
 Gsum\_terms <- vector('list', length = length(clust\_id\_vec))  
   
 for (g in clust\_id\_vec) { ### g <- 1  
 X\_g <- X[clust\_vec == g, ]  
 e\_g <- e\_tilde\_g[[g]]  
   
 Gsum\_terms[[g]] <- t(X\_g) %\*% e\_g %\*% t(e\_g) %\*% X\_g  
 }  
   
 Gsum <- Reduce('+', Gsum\_terms)  
   
 V\_tilde\_kmeans <- Xt\_X %\*% Gsum %\*% Xt\_X  
   
 se\_tilde\_kmeans[[i]] <- diag(V\_tilde\_kmeans) %>% sqrt()  
}

clust\_vec\_500 <- clust\_ids[ , 4]  
  
### Calculate Omega\_n: remove g obs and calc beta hat; then calc  
### e\_tilde\_g for each group (store in a list object)  
G <- length(unique(clust\_vec\_500))  
  
omega\_g <- vector('list', length = G)  
  
for (g in 1:G) { ### g <- 1  
  
 y\_g <- y[clust\_vec\_500 == g]  
 X\_g <- X[clust\_vec\_500 == g, ]  
   
 e\_hat\_g <- y\_g - X\_g %\*% beta\_hat\_kmeans  
   
 omega\_g[[g]] <- t(X\_g) %\*% e\_hat\_g %\*% t(e\_hat\_g) %\*% X\_g  
}  
  
Omega\_n <- Reduce('+', omega\_g)  
  
a\_n <- (n\_obs - 1) / (n\_obs - k\_regr) \* G / (G - 1)  
  
V\_hat\_clust\_500 <- a\_n \* Xt\_X %\*% Omega\_n %\*% Xt\_X  
  
se\_hat\_clust\_500 <- sqrt(diag(V\_hat\_clust\_500))

se\_tilde\_kmeans\_df <- se\_tilde\_kmeans %>%  
 setNames(paste0('se\_tilde\_', n\_clusts)) %>%  
 bind\_cols()  
  
beta\_se\_df4 <- data.frame(estimate = rownames(beta\_hat),  
 beta\_hat,  
 se\_tilde\_clust,  
 se\_hat\_clust\_500) %>%  
 bind\_cols(se\_tilde\_kmeans\_df) %>%  
 mutate(t\_54 = round(beta\_hat / se\_tilde\_clust, 2),  
 t\_150 = round(beta\_hat / se\_tilde\_150, 2),  
 t\_tilde\_clust\_500 = round(beta\_hat / se\_tilde\_500, 2),  
 t\_hat\_clust\_500 = round(beta\_hat / se\_hat\_clust\_500, 2),  
 se\_tilde\_100 = round(se\_tilde\_100, 5),  
 se\_tilde\_150 = round(se\_tilde\_150, 5),  
 se\_tilde\_250 = round(se\_tilde\_250, 5),  
 se\_tilde\_500 = round(se\_tilde\_500, 5),  
 se\_tilde\_clust = round(se\_tilde\_clust, 5),  
 se\_hat\_clust\_500 = round(se\_hat\_clust\_500, 5))  
  
knitr::kable(beta\_se\_df4, format = 'html',  
 caption = 'Table v: Standard errors and t statistics',  
 col.names = c('Estimate', '$\\widehat\\beta$',   
 '$\\widetilde\\sigma\_{alt.clust,54}$',  
 '$\\widetilde\\sigma\_{alt.clust,100}$',  
 '$\\widetilde\\sigma\_{alt.clust,150}$',  
 '$\\widetilde\\sigma\_{alt.clust,250}$',  
 '$\\widetilde\\sigma\_{alt.clust,500}$',  
 '$\\widehat\\sigma\_{clust,500}$',  
 '$t\_{alt.clust,54}$',  
 '$t\_{alt.clust,150}$',  
 '$t\_{alt.clust,500}$',  
 '$t\_{clust,500}$')) %>%  
 kable\_styling('striped', full\_width = FALSE)

Table v: Standard errors and t statistics

Estimate

sst\_2012\_mean

4.6592640

3.50407

1.25773

1.90478

1.91777

1.69038

1.29929

1.33

2.43

3.59

3.70

oa\_2016\_mean

-16.9142263

11.54064

2.16542

3.12460

2.75698

2.77265

2.27003

-1.47

-6.14

-7.45

-7.81

uv\_2016\_mean

3.2213246

5.20329

1.59462

2.18281

2.19606

1.91551

1.66325

0.62

1.47

1.94

2.02

latAbs

0.1298950

0.03445

0.02729

0.03620

0.03082

0.03172

0.02841

3.77

4.21

4.57

4.76

dummySouth

-0.2912758

2.85586

0.66415

1.13338

1.02117

0.88957

0.68097

-0.10

-0.29

-0.43

-0.44

mpa\_pct

-1.9030904

0.69089

0.44521

0.61218

0.65602

0.59897

0.46021

-2.75

-2.90

-4.14

-4.27

oa\_S

-4.1289479

20.55364

4.03081

6.19499

5.79046

5.27692

4.15928

-0.20

-0.71

-0.99

-1.02

sst\_S

-2.5529086

4.16877

1.57536

2.35140

2.32991

2.06576

1.62131

-0.61

-1.10

-1.57

-1.62

uv\_S

-2.1906827

7.59171

2.21171

3.18894

3.03787

2.64180

2.28641

-0.29

-0.72

-0.96

-0.99

latS

0.1400159

0.07026

0.03336

0.04671

0.04123

0.03956

0.03454

1.99

3.40

4.05

4.20

const

12.2651278

1.62047

0.40708

0.71002

0.63479

0.56462

0.41915

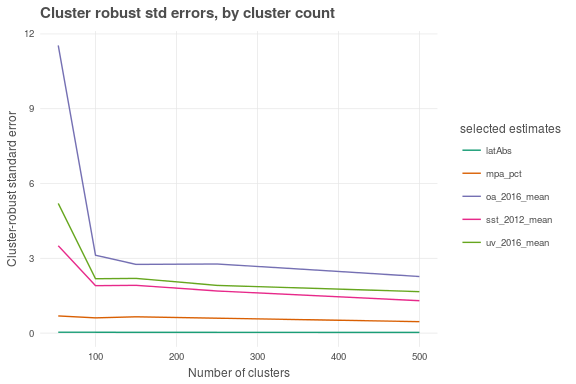
7.57

19.32

29.26

30.13

beta\_plot\_df <- beta\_se\_df4 %>%  
 gather(clust, se, starts\_with('se')) %>%  
 mutate(clust = str\_replace(clust, 'clust', '54'),  
 clust = str\_replace(clust, 'se\_tilde\_', ''),  
 clust = as.numeric(clust)) %>%  
 filter(str\_detect(estimate, 'mean|mpa|latAbs'))  
  
ggplot(beta\_plot\_df, aes(x = clust, y = se, color = estimate)) +  
 geom\_line() +  
 ggtheme\_plot() +  
 scale\_color\_brewer(palette = 'Dark2') +  
 labs(title = 'Cluster robust std errors, by cluster count',  
 color = 'selected estimates',  
 x = 'Number of clusters',  
 y = 'Cluster-robust standard error')



# Results

* table of beta and standard errors for classic OLS, leave-one-out homosked (pred errors), adjusted White heterosked, and cluster-robust leave-one-out

Due to the nature of this report, in which the methodology was developed in parallel with the analysis, we have already seen the results for a number of different assumptions and methods. Here we summarize resulting beta terms and the standard errors calculated from covariance matrices under six assumptions and methods:

1. classic OLS standard errors, assuming homoskedasticity (Hansen eq. 4.36, using estimator)
2. prediction errors (leave-one-out) assuming homoskedasticity (Hansen eq. 4.36, using estimator)
3. heteroskedasticity-robust adjusted White standard errors (4.38)
4. heteroskedasticity-robust leave-one-out prediction errors (4.40)
5. finite-sample-adjusted cluster-robust standard errors a la White (4.53)
6. cluster-robust prediction errors (leave-one-cluster-out) assuming heteroskedasticity (described in Hansen section 4.20, no equation number)

Clustered errors calculated using 500 clusters based upon latitude, longitude, and Longhurst province boundaries.

results\_df <- beta\_se\_df %>%  
 left\_join(beta\_se\_df2) %>%  
 left\_join(beta\_se\_df3) %>%  
 left\_join(beta\_se\_df4) %>%  
 select(estimate, beta\_hat,   
 se\_hat, t\_hat,  
 se\_tilde, t\_tilde,  
 se\_white, t\_white,  
 se\_andrews, t\_andrews,  
 se\_hat\_clust\_500, t\_hat\_clust\_500,   
 se\_tilde\_500, t\_tilde\_clust\_500) %>%  
 mutate(beta\_hat = round(beta\_hat, 3))  
   
knitr::kable(results\_df, format = 'html',  
 caption = 'Table 1',  
 col.names = c('Estimate', '$\\widehat\\beta$',   
 '$se\_1$',  
 '$t.stat\_{1}$',  
 '$se\_2$',  
 '$t.stat\_2$',  
 '$se\_3$',  
 '$t.stat\_3$',  
 '$se\_4$',  
 '$t.stat\_4$',  
 '$se\_5$',  
 '$t.stat\_5$',  
 '$se\_6$',  
 '$t.stat\_6$')) %>%  
 kable\_styling('striped', full\_width = FALSE) %>%  
 column\_spec(c(2, 4, 6, 8, 10, 12), border\_right = TRUE) %>%  
 add\_header\_above(c(' ' = 2,  
 '$\\mathbf{\\widehat V^0\_{\\hat\\beta}}$' = 2,  
 '$\\mathbf{\\widetilde V^0\_{\\hat\\beta}}$' = 2,  
 '$\\mathbf{\\widehat V\_{\\hat\\beta}}$' = 2,  
 '$\\mathbf{\\widetilde V\_{\\hat\\beta}}$' = 2,  
 '$\\mathbf{\\widehat V\_{\\hat\\beta}}\_{clust}$' = 2,  
 '$\\mathbf{\\widetilde V\_{\\hat\\beta}}\_{alt.clust}$' = 2))

Table 1

Estimate

sst\_2012\_mean

4.659

0.10776

43.24

0.10776

43.24

0.10899

42.75

0.10901

42.74

1.25773

3.70

1.29929

3.59

oa\_2016\_mean

-16.914

0.15605

-108.39

0.15606

-108.38

0.19452

-86.95

0.19456

-86.93

2.16542

-7.81

2.27003

-7.45

uv\_2016\_mean

3.221

0.11916

27.03

0.11916

27.03

0.14262

22.59

0.14264

22.58

1.59462

2.02

1.66325

1.94

latAbs

0.130

0.00175

74.12

0.00175

74.12

0.00240

54.18

0.00240

54.17

0.02729

4.76

0.02841

4.57

dummySouth

-0.291

0.06475

-4.50

0.06475

-4.50

0.05639

-5.17

0.05639

-5.17

0.66415

-0.44

0.68097

-0.43

mpa\_pct

-1.903

0.06207

-30.66

0.06207

-30.66

0.05766

-33.01

0.05767

-33.00

0.44521

-4.27

0.46021

-4.14

oa\_S

-4.129

0.24910

-16.58

0.24912

-16.57

0.33336

-12.39

0.33340

-12.38

4.03081

-1.02

4.15928

-0.99

sst\_S

-2.553

0.14522

-17.58

0.14523

-17.58

0.13747

-18.57

0.13748

-18.57

1.57536

-1.62

1.62131

-1.57

uv\_S

-2.191

0.19667

-11.14

0.19668

-11.14

0.22310

-9.82

0.22313

-9.82

2.21171

-0.99

2.28641

-0.96

latS

0.140

0.00243

57.53

0.00243

57.53

0.00291

48.15

0.00291

48.15

0.03336

4.20

0.03454

4.05

const

12.265

0.04457

275.17

0.04458

275.15

0.03506

349.79

0.03507

349.76

0.40708

30.13

0.41915

29.26

Basing significance on prediction errors calculated from the alternative cluster-robust covariance matrix estimator , it appears that the dummy for south latitudes, , as well as the interaction terms , , and , are not statistically significant (). In addition, just misses the critical value for .

# Discussion

The global cumulative impact assessment (REF Halpern et al 2008, 2015) predicts impacts to marine ecosystems based upon intensity of stressors and vulnerability of specific habitat types. Here we reverse that, comparing a set of spatially distributed stressors -- in this case only a subset of the entire suite of stressors included in the overall cumulative impact framework -- to a measure of ecosystem health in order to infer the impacts of stressors. The direction and magnitude of regression coefficients between ecosystem health and stressor intensity can lend validity to the cumulative impacts approach, and where the coefficients do not match expectations (e.g. ocean acidification, as noted below) we can search for underlying causes that confound our expectations.

It must be noted that there may be some direct dependence between our measure of biodiversity risk and our stressors. IUCN Red List assessments focus primarily on risks associated with changes in population, range, and habitat intactness, but Criteria A1 also includes '(d) actual or potential levels of exploitation' and '(e) the effects of introduced taxa, hybridization, pathogens, pollutants, competitors or parasites.' While these are just a small part of the criteria used to categorize extinction risk of individual species, there is an element of double-counting that can potentially amplify the significance of the results.

### Climate stressors

The conditional mean of biodiversity risk should be expected to increase for increases in any of the three climate stressors included in this study. At a species level, any particular stressor may have little to no observable direct effect. However, at the ecosystem level, small impacts across many species may accumulate and make the ecosystem less resilient to future stresses and disturbances.

In examining the coefficients on each stressor, recall that each stressor value is a normalized score from 0 (no increased stress) to 1 (maximum observed increased stress), and the biodiversity risk score is a value from 0 (all species at Least Concern) to 100 (all species extinct); every 20 point increase indicates a change to the next higher risk category.

For sea surface temperature, a shift from the lowest and highest valued cells correlates with an increase in the conditional mean of biodiversity risk by 4.66 points; high ultraviolet radiation exposure, while not quite significant at the level, correlates with an increase in the conditional mean of 3.22 points. The direction of these effects is unsurprising.

Very surprising, however, is the large negative coefficient for ocean acidification. A change in ocean acidification pressure from none at all to maximum stress correlates strongly with a *decrease* in risk of 16.9 points, an improvement by nearly a full IUCN extinction risk category. Examining the maps of mean risk and ocean acidification we can visually detect the inverse trend. Arctic and Antarctic waters are more vulnerable to ocean acidification at least in part because colder waters have a higher capacity to absorb carbon dioxide, and a small decrease in for areas where is already near 1 will result in higher ocean acidification stressor values. Yet biodiversity risk in polar regions is apparently lower; this seems in part to be due to a high proportion of healthy and wide-ranging species, especially polar marine birds, driving the mean risk down.

oa\_max\_cells <- oa\_rast %>%   
 filter(oa\_2016\_mean == 1) %>%  
 left\_join(latlong\_df, by = c('x' = 'long', 'y' = 'lat'))  
oa\_spp\_cells <- read\_csv(file.path(dir\_o\_anx, 'am\_files', 'am\_spp\_cells.csv'))   
  
oa\_spp <- oa\_spp\_cells %>%  
 filter(loiczid %in% oa\_max\_cells$loiczid) %>%  
 select(am\_sid) %>%  
 distinct()  
   
dir\_am\_data <- file.path(dir\_M, 'git-annex/globalprep/\_raw\_data/aquamaps/d2017')  
oa\_spp\_info <- read\_csv(file.path(dir\_am\_data, 'csv', 'speciesoccursum.csv')) %>%  
 filter(SPECIESID %in% oa\_spp$am\_sid)  
  
table(oa\_spp\_info$iucn\_code)  
 # EN LC LR/lc NT VU   
 # 5 80 1 6 8   
# (81 \* 0 + 6 \* .2 + 8 \* .4 + 5 \* .6) / 100 ### = .074

### Marine protected areas

The coefficient on MPA presence indicates that for a given cell, a change from entirely unprotected to entirely protected correlates with a 1.90 point decrease in the conditional mean of biodiversity risk. While this could be due simply to the designation of MPAs in places with already healthy ecosystems, REF Lester et al. determined that generally, MPAs tend to be placed in areas of poorer health. We cannot draw a causal connection based on our study, however, as there are many unobservables that may not be independent of our treatment (MPA presence), so we have no hope of justifying a conditional independence assumption.

When considering the apparent improvement by the designation of MPAs, it is important to note that MPAs necessarily fall within national Exclusive Economic Zones (EEZs), the swath of ocean extending out 200 miles from a country's coast (or to the nearest EEZ of another country). This leaves massive portions of the open ocean waters unprotected. In addition, most MPAs are limited in size and thus afford more protection to species with limited mobility or limited home ranges. MPAs are therefore likely to provide more benefit to coastal species rather than highly mobile pelagics; further revisions of our analysis will need to examine the differential effect of MPA protection based on species range or taxonomic group. Finally, our analysis includes MPAs at all levels of IUCN classification (IUCN I-VI), though not all MPAs are created equal. We must consider in future analyses differences between designated no-take zones, conservation-oriented categories (IUCN I-IV), and human use-oriented categories (IUCN V, VI), as well as the age of the MPA.

### Spatial dependence

In any regression on spatial analysis, spatial correlations and dependence are likely to arise, due to the effects of unobserved spatially distributed variables or spillover effects across soft boundaries (look into REFS from Davenport: Openshaw and Taylor 1979; Openshaw 1984; Anselin 2002). In our regression of marine biodiversity risk against multiple climate stressors, we have not directly accounted for potential regional variations in ocean conditions that may affect species assemblages or ecosystem resilience, such as salinity, net primary productivity, ocean currents, and other more intangible factors. Effects of such variables therefore fall within our error term and create spatial dependence of our residuals.

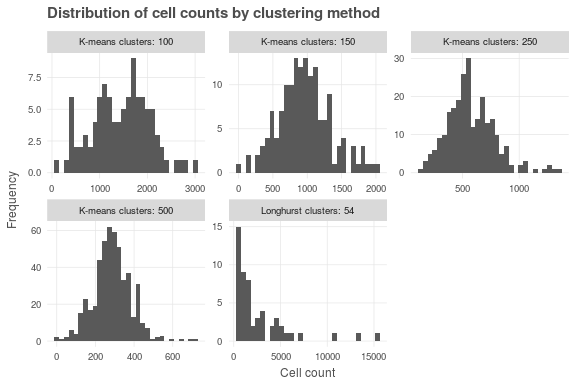
A standard way of accounting for spatial dependence is a spatial weights matrix, which essentially links every observation to every other observation through spatial connectivity or distance. In the case of this analysis, we have divided the oceans into half-degree cells, and even after removing land cells and cells with fewer than five assessed species, we are working with 143,890 observations. An spatial weights matrix in this case would comprise over 20 billion cell-to-cell elements. Since in future extensions of this analysis, we hope to work with data at the ~1 km 1 km resolution of the stressor layers, each layer including approximately 420 million cells; at 1.7 x 1017 elements, a spatial weights matrix at this resolution is unreasonably large.

Spatial clustering is another approach that can help account for spatial dependence in our regression, perhaps not as elegant as a spatial weights matrix, but far more tractable in terms of processing and memory limitations. In terms of our analysis here, I have implemented a simplified clustering scheme that bears some caveats. As we have implemented it, spatial clustering assumes a flat, rectangular earth rather than a spherical earth; this results in a discontinuity at the east-most (centered at 179.75°E) and west-most cells (at 179.75° W), which are actually adjacent on the globe. Additionally, our analysis is based upon cells defined as 0.5° in latitude and longitude; this results in cells with very different areas, around 4,200 km2 at the equator and nearing zero at the poles. While these differences may not create substantial differences in our regression results, they are worth exploring in a sensitivity analysis, or avoiding by using spatially-explicit clustering techniques on a raster in an equal area projection such as Mollweide (EPSG:54009).

To implement spatial clustering, we began with the boundaries of Longhurst biogeographical provinces, which roughly account for biogeochemical similarities in marine basins, intending to implicitly capture some of the spatially dependent unobserved variables. Other spatial groupings of ocean basins exist that may correlate more strongly with risk, such as Marine Ecoregions of the World (REF MEOW). However, by breaking up the 54 Longhurst provinces into 500 separate spatially compact clusters, presumably the cluster-robust errors can account for even finer scale patterns, and the original Longhurst boundaries become less relevant.

Since our clusters are based not on well-defined external factors but on arbitrary spatially compact neighborhoods, we have leeway to define cluster count and size that will result in the best balance between in-cluster results and between-cluster results. One factor to consider is that cluster robust approaches generally assume a similar number of observations in each cluster. We can examine the distribution of cell-counts per cluster for various clustering schemes, beginning with the Longhurst provinces which show a very skewed distribution; .

cluster\_count\_df <- cluster\_ids\_df %>%  
 left\_join(longh\_df) %>%  
 group\_by(longhurst) %>%  
 mutate(longhurst\_count = n()) %>%  
 group\_by(clust\_100) %>%  
 mutate(clust\_100\_count = n()) %>%  
 group\_by(clust\_150) %>%  
 mutate(clust\_150\_count = n()) %>%  
 group\_by(clust\_250) %>%  
 mutate(clust\_250\_count = n()) %>%  
 group\_by(clust\_500) %>%  
 mutate(clust\_500\_count = n()) %>%  
 ungroup() %>%  
 select(-loiczid) %>%  
 distinct()  
   
cluster\_count\_df2 <- cluster\_count\_df %>%  
 gather(clust\_100:longhurst, key = 'cluster', value = 'id') %>%  
 gather(longhurst\_count:clust\_500\_count, key = 'cluster2', value = 'count') %>%  
 filter(str\_detect(cluster2, cluster)) %>%  
 select(-cluster2) %>%  
 distinct() %>%  
 mutate(cluster = str\_replace(cluster, 'clust\_', 'K-means clusters: '),  
 cluster = str\_replace(cluster, 'longhurst', 'Longhurst clusters: 54'))  
  
ggplot(cluster\_count\_df2, aes(x = count)) +  
 ggtheme\_plot() +  
 geom\_histogram() +  
 facet\_wrap( ~ cluster, scales = 'free') +  
 labs(title = 'Distribution of cell counts by clustering method',  
 x = 'Cell count', y = 'Frequency')



What is the "right" number of clusters? Clusters essentially take the place of individual observations in the error calculation; so for between-cluster comparisons, maximizing the number of clusters is desirable. But for reliable in-cluster results, each cluster needs an appropriate (and approximately equal) number of observations. Intuitively this would mean the ideal number of clusters would be - in this case, with 143,890 observations, we'd want ~380 clusters. In the limit, as cluster count approaches the count of individual cells , we should probably approach the results for the unclustered analysis (in this case, the adjusted White heteroskedastic-robust analysis).

# Next steps

Based on the methods, results, and discussion, there is still a long way to go in this project. We found some reassuring results in the correlations between sea surface temperature and ultraviolet exposure with our definition of biodiversity risk; on the other hand, the correlation between risk and ocean acidification wildly confounds initial expectations. While we see positive results due to the protection afforded by MPAs, we must be cautious in inferring causal connections without further examination.

Now that we have established a basic methodology for this regression analysis, we can look forward to future modifications and improvements to improve confidence in our results.

### Methodological improvements

* Look into other methods of accounting for spatial dependence, including spatially explicit clustering methods and tractable methods for incorporating spatial weights matrices.
* Perform the analysis on a finer scale and with observations based upon equal area, for example the 934m cell resolution in Mollweide equal-area projection at which the cumulative impact stressor layers are available.
* Identify analysis methods already native to R, rather than coding regressions from scratch. While from a student perspective it has been pedagogically more interesting and informative to code the regressions and error calculations based upon matrix manipulations, if peer-reviewed methods already exist to perform these analyses, we should take advantage, both to ensure accuracy and to improve readability of code.

### Alternate models for regression

* Include the full suite of cumulative impact stressors. While climate stressors are by far the most broad-reaching, other human activities such as fishing, pollution, and shipping are likely to have direct and consequential impacts on biodiversity risk.
* Analyze cells limited to coastal regions, for example by clipping all stressors and species risk layers to EEZ boundaries. This would be complementary to the full-ocean analysis we have already performed. Note that this will impact the cluster count and size decisions since the number of observations will shrink drastically when the analysis is confined to coastal regions.
* Analyze species for small, medium, and large ranges separately; if species mobility data can be found, this would be another useful filter. Most species have a single IUCN risk category for their entire range. Especially for wide-ranging and highly mobile species such as marine mammals, marine birds, and large pelagic fishes, local scale spikes in stressor intensity are unlikely to have a measurable impact on the IUCN risk categorization. For the small subset of species with subpopulation IUCN assessments, we may be able to assign spatially distinct risk categories.
* Analyze by taxonomic groups - plants, invertebrates, bony fish, cartilaginous fish, marine mammals, marine reptiles, and birds are all likely to suffer different responses to stressors. Examining stressors vs. risk by taxonomic category may provide additional insights, for example helping to understand the unexpected apparent benefits of ocean acidification.

To paraphrase the advice of a wise econometrics professor: now that we've found some potentially interesting results, we must try our best to make those results disappear. If the results persist, then maybe we have something worth talking about.