C. californicus Spatiotemporal Variation of Shell Size and Shape

Malloy and Ul-Hasan and Lewis et al

Mar 24, 2019

Last edited [Sabah Ul-Hasan]: Mar 24, 2019

<https://github.com/MichaelMalloy/C_californicus_Morphology>

Contributors of data by order of contribution: MM, SUH, CAO, SC Contributors of code by order of contribution: SUH, LL, MM

SuppTabs 2-4 Data, post-processing: Raw -> Clean -> SoCal Lines 23 - 147 Figure 1a-b Map of C. californicus range Lines 149 - 263 Fig 2b Shell allometry (shell manual) Lines 265 - 306 Figure 3 Bergmann’s rule: Morphology ~ temperature Lines 309 - 373 Figure 4 Morphology ~ Bergmann’s rule (temperature) - coastal v island  
Figure 5 Morphology ~ human population density Table 1 Table of what’s correlated vs not to morphology (cont v not)

Overview of data and post-processing

###### Upload data ######  
# install.packages("RCurl")  
library("RCurl") # package needed for downloading data from Github repo

## Warning: package 'RCurl' was built under R version 3.5.2

## Loading required package: bitops

# packageVersion("RCurl") # v1.95.4.12   
  
# Raw <- read.csv(text=getURL("https://raw.githubusercontent.com/MichaelMalloy/C\_californicus\_Morphology/master/SupplementalTable2\_Data-Raw.csv")) not working, revisit  
# Ensure blanks are read as NAs  
  
  
  
# Alternative upload  
setwd("~/Desktop/MSa/Mar24-2019")  
Raw = read.csv("SupplementalTable2\_Data-Raw.csv", header=T, na.strings=c("","NA")) # 2996 obs, recognize blank cells as NA  
###### Upload data ######  
  
  
###### Data structure ######  
str(Raw) # Quick view of RawData

## 'data.frame': 2996 obs. of 39 variables:  
## $ Specimen\_Number : int 88 89 90 91 98 99 100 101 102 92 ...  
## $ Collection : Factor w/ 9 levels "ANSP","CAS","iNat",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Lot\_Number : Factor w/ 390 levels "1 atm Gland",..: 103 103 103 103 105 105 105 105 105 97 ...  
## $ Collector : Factor w/ 114 levels "alex\_bairstow",..: NA NA NA NA 38 38 38 38 38 97 ...  
## $ Ecoregion : Factor w/ 2 levels "Northern California",..: 1 1 1 1 2 2 2 2 2 2 ...  
## $ County : Factor w/ 10 levels "Baja California",..: 3 3 3 3 2 2 2 2 2 2 ...  
## $ Location : Factor w/ 20 levels "ANA","BS","CAT",..: 6 6 6 6 3 3 3 3 3 3 ...  
## $ Site : Factor w/ 151 levels "All Saints Bay",..: 39 39 39 39 105 105 105 105 105 103 ...  
## $ Land\_Category : Factor w/ 3 levels "Coastal","Island",..: 1 1 1 1 2 2 2 2 2 2 ...  
## $ Island\_Size : Factor w/ 3 levels "Large","Medium",..: NA NA NA NA 1 1 1 1 1 1 ...  
## $ SoCal\_Group : Factor w/ 8 levels "CAT","Los Angeles",..: NA NA NA NA 1 1 1 1 1 1 ...  
## $ MPA : Factor w/ 2 levels "no","yes": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Latitude : num 36.6 36.6 36.6 36.6 33.4 ...  
## $ Longitude : num -122 -122 -122 -122 -119 ...  
## $ Mean\_SST : num 13.1 13.1 13.1 13.1 16 ...  
## $ Mean\_SST\_Sdev : num 1.55 1.55 1.55 1.55 1.66 ...  
## $ Year : int 1892 1892 1892 1892 1901 1901 1901 1901 1901 1901 ...  
## $ Season : Factor w/ 4 levels "Fall","Spring",..: NA NA NA NA 2 2 2 2 2 NA ...  
## $ Month : int NA NA NA NA 5 5 5 5 5 NA ...  
## $ Epoch : Factor w/ 4 levels "Anthropocene",..: 2 2 2 2 2 2 2 2 2 2 ...  
## $ Formation : Factor w/ 19 levels "12th Terrace",..: NA NA NA NA NA NA NA NA NA NA ...  
## $ Length\_mm : num 17.8 22.2 23.8 23.6 17.6 ...  
## $ Width\_mm : num 10 12.51 13.93 13.91 9.96 ...  
## $ Globosity\_WoverL: num 0.56 0.56 0.59 0.59 0.57 0.55 0.58 0.53 0.55 0.53 ...  
## $ Aperture\_mm : num 2.2 2.59 3.02 3.26 2.12 3.05 3.19 3.33 3.51 1.52 ...  
## $ Thickness\_mm : num 0.3 0.48 0.53 0.54 0.39 0.48 0.77 0.6 0.46 0.25 ...  
## $ Habitat : Factor w/ 4 levels "algae forest",..: NA NA NA NA 4 4 4 4 4 4 ...  
## $ Depth : Factor w/ 3 levels "deep","intertidal",..: NA NA NA NA 3 3 3 3 3 1 ...  
## $ BRY : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ BAR : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ LMP : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ TWS : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ COR : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ ALG : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ SCR : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ BOR\_TAP : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ BOR\_STR : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ BOR\_INC : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ Image\_url : Factor w/ 99 levels "https://static.inaturalist.org/photos/10375087/medium.jpg?1505096528",..: NA NA NA NA NA NA NA NA NA NA ...

# Plots of shell morphology measurements as a point of reference for outliers   
morphos = c("Globosity\_WoverL", "Length\_mm", "Width\_mm", "Aperture\_mm", "Thickness\_mm")   
  
# install.packages("PerformanceAnalytics")  
library("PerformanceAnalytics") # Need package for correlation charts

## Loading required package: xts

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

##   
## Attaching package: 'PerformanceAnalytics'

## The following object is masked from 'package:graphics':  
##   
## legend

# packageVersion("PerformanceAnalytics") # v1.5.2  
  
# install.packages("tidyverse")  
library("tidyverse") # Need package pipes ( %>% or "then" )

## ── Attaching packages ────────────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.1.0 ✔ purrr 0.3.0   
## ✔ tibble 2.0.1 ✔ dplyr 0.8.0.1  
## ✔ tidyr 0.8.3 ✔ stringr 1.4.0   
## ✔ readr 1.3.1 ✔ forcats 0.3.0

## Warning: package 'tibble' was built under R version 3.5.2

## Warning: package 'tidyr' was built under R version 3.5.2

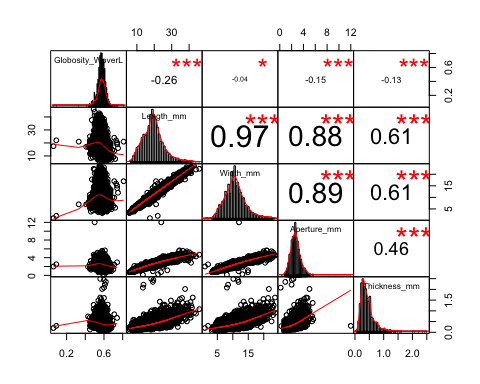
## Warning: package 'purrr' was built under R version 3.5.2

## Warning: package 'dplyr' was built under R version 3.5.2

## Warning: package 'stringr' was built under R version 3.5.2

## ── Conflicts ───────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ tidyr::complete() masks RCurl::complete()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::first() masks xts::first()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ dplyr::last() masks xts::last()

# packageVersion("tidyverse") # v1.2.1  
   
chart.Correlation(Raw %>% select(morphos)) # we can see outliers here



###### Data structure ######  
  
  
###### Clean-up ######  
# Subset for shells without aperture / thickness (iNat will be included)  
Raw\_globoshells = subset(Raw, is.na(Raw$Thickness\_mm))  
Raw\_iNat = subset(Raw, Raw$Collection=="iNat") # 99 specifically from iNat  
  
# Filter outliers (min and max values identified by viewing Raw)  
# Confirmed outliers from operator error (measuring or datasheet input)  
Clean\_OutliersRemoved = Raw %>% # 2946 obs   
 filter(Thickness\_mm < 2) %>% # low thickness  
 filter(!(Width\_mm < 15 & Thickness\_mm > 2)) %>% # too thick for width  
 filter(Aperture\_mm < 6) %>% # small aperture  
 filter(!(Length\_mm > 20 & Aperture\_mm <2)) %>% # aperture too small for length  
 rbind(Raw\_globoshells) %>% # adds 775 shells back in  
 filter(Width\_mm > 2) %>% # too narrow (note this gets rid of iNat)  
 rbind(Raw\_iNat) %>% # add iNat back in  
 filter(Globosity\_WoverL > 0.42 & Globosity\_WoverL < 0.71 ) %>%   
 data.frame # ensure it's a data frame  
# Remove 25 shells from Pliocene (too old, few)  
Clean\_OutliersRemoved=subset(Clean\_OutliersRemoved, !Epoch=="Pliocene")  
  
# 50 shells of original 2996 removed (2946 obs)  
length(Raw %>% filter(Thickness\_mm >= 2) %>% .$Specimen\_Number)

## [1] 6

paste(length(Raw$Specimen\_Number)-length(Clean\_OutliersRemoved$Specimen\_Number), "removed of", length(Raw$Specimen\_Number), "shells")

## [1] "50 removed of 2996 shells"

# Ensure updated Clean dataset has 3 or more specimens per lot  
Clean = Clean\_OutliersRemoved %>%   
 group\_by(Lot\_Number) %>%   
 mutate(Specimens\_per\_Lot = length((Lot\_Number))) %>%  
 ungroup %>%   
 filter(Specimens\_per\_Lot > 2 | Collection == "iNat") # keep catalogs with > 2 in lot and iNat (1 per 'lot')  
  
# 61 shells of original 2996 removed, 11 from Clean\_OutliersRemoved (2935 obs)  
paste(length(Raw$Specimen\_Number)-length(Clean$Specimen\_Number), "removed of", length(Raw$Specimen\_Number), "shells")

## [1] "61 removed of 2996 shells"

# Validation check for NA values for final data set  
cat("\n\n",  
 sum(is.na(Clean$Globosity\_WoverL)), "have no globosity ratio,\n", # 0  
 sum(is.na(Clean$Width\_mm)), "have no width measurement,\n", # 99 [iNat]  
 sum(is.na(Clean$Length\_mm)), "have no length measurement,\n", # 99 [iNat]  
 sum(is.na(Clean$Thickness\_mm)), "have no thickness measurement,\n", # 742 = 99 [iNat] + 338 PR + 273 Pleisto  
 sum(is.na(Clean$Aperture\_mm)), "have no aperture measurement" ) # 742 = 99 [iNat] + 338 PR + 273 Pleisto

##   
##   
## 0 have no globosity ratio,  
## 99 have no width measurement,  
## 99 have no length measurement,  
## 742 have no thickness measurement,  
## 742 have no aperture measurement

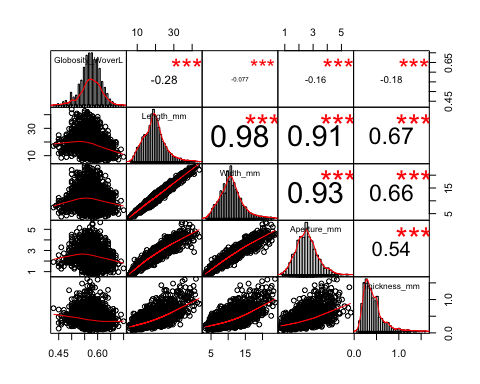
# Order the data  
Clean %>% group\_by(Lot\_Number) %>% summarise(N = length(Specimen\_Number))

## # A tibble: 379 x 2  
## Lot\_Number N  
## <fct> <int>  
## 1 1 atm Gland 5  
## 2 100 PSI Gland 5  
## 3 100023 10  
## 4 10032.43 4  
## 5 10035.15 3  
## 6 10130.21 31  
## 7 10167.44 8  
## 8 10307939 1  
## 9 10341412 1  
## 10 105150 10  
## # … with 369 more rows

Clean %>% group\_by(Epoch) %>% summarise(N = length(Specimen\_Number))

## # A tibble: 3 x 2  
## Epoch N  
## <fct> <int>  
## 1 Anthropocene 1492  
## 2 Holocene 1138  
## 3 Pleistocene 305

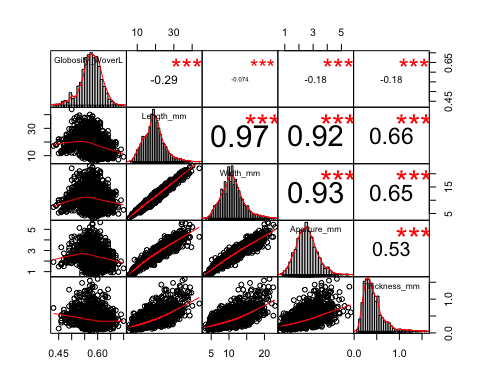
# Re-do of correlation chart  
chart.Correlation(Clean %>% select(morphos)) # much improved



# Parameters for SoCal points  
counties = c("Sonoma", "San Mateo", "Monterey", "San Luis Obispo", "Santa Barbara", "Ventura", "Los Angeles", "Orange", "San Diego", "Baja California")  
Raw$County = factor(Raw$County, levels=counties) # order counties  
  
socal = c("Santa Barbara", "Ventura", "Los Angeles", "Orange", "San Diego")  
Clean\_SoCal = Clean %>% filter(County %in% socal) # 2597 obs for SoCal   
  
socal\_locations = c("SB", "NLA", "LA", "SLA", "SD", "ANA", "SCZI", "SRI", "SMI", "CAT", "SBI", "SNI", "SCI") # socal location order  
Clean\_SoCal$Location = factor(Clean\_SoCal$Location, levels=socal\_locations) # order   
  
# Distribution of shells by county  
Clean\_SoCal %>% group\_by(County) %>% summarise(N = length(Epoch)) %>% data.frame

## County N  
## 1 Los Angeles 1363  
## 2 Orange 280  
## 3 San Diego 526  
## 4 Santa Barbara 345  
## 5 Ventura 83

# Check correlation for SoCal subset  
chart.Correlation(Clean\_SoCal %>% select(morphos)) # Also looks good



###### Clean-up ######  
  
  
###### Save cleaned-up data for downstream visualization and analysis ######  
  
# Data structure for Clean and SoCal subset  
sort(unique(Clean$Collection)) # Unique collections: 9

## [1] ANSP CAS iNat LACM-M LACM-M-IP MCZ PR   
## [8] SBNHM SIO   
## Levels: ANSP CAS iNat LACM-M LACM-M-IP MCZ PR SBNHM SIO

sort(unique(Clean$Catalog)) # Unique lots: 379

## Warning: Unknown or uninitialised column: 'Catalog'.

## NULL

sort(unique(Clean$County)) # Unique counties: 10

## [1] Baja California Los Angeles Monterey Orange   
## [5] San Diego San Luis Obispo San Mateo Santa Barbara   
## [9] Sonoma Ventura   
## 10 Levels: Baja California Los Angeles Monterey Orange ... Ventura

sort(unique(Clean$Location)) # Unique locations: 20

## [1] ANA BS CAT LA MB MO NBC NLA SB SBC SBI SCI SCZI SD   
## [15] SLA SM SMI SNI SO SRI   
## 20 Levels: ANA BS CAT LA MB MO NBC NLA SB SBC SBI SCI SCZI SD SLA ... SRI

sort(unique(Clean$Land\_Category)) # Unique location types: 3

## [1] Coastal Island Open   
## Levels: Coastal Island Open

sort(unique(Clean$Habitat)) # Unique habitats: 4

## [1] algae forest gravel   
## [3] rocky reef or intertidal sand   
## Levels: algae forest gravel rocky reef or intertidal sand

sort(unique(Clean$Depth)) # Unique depths: 3

## [1] deep intertidal subtidal   
## Levels: deep intertidal subtidal

sort(unique(Clean\_SoCal$Collection)) # Unique collections: 9

## [1] ANSP CAS iNat LACM-M LACM-M-IP MCZ PR   
## [8] SBNHM SIO   
## Levels: ANSP CAS iNat LACM-M LACM-M-IP MCZ PR SBNHM SIO

sort(unique(Clean\_SoCal$Catalog)) # Unique lots: 333

## Warning: Unknown or uninitialised column: 'Catalog'.

## NULL

sort(unique(Clean\_SoCal$County)) # Unique counties: 5

## [1] Los Angeles Orange San Diego Santa Barbara Ventura   
## 10 Levels: Baja California Los Angeles Monterey Orange ... Ventura

sort(unique(Clean\_SoCal$Location)) # Unique locations: 13

## [1] SB NLA LA SLA SD ANA SCZI SRI SMI CAT SBI SNI SCI   
## Levels: SB NLA LA SLA SD ANA SCZI SRI SMI CAT SBI SNI SCI

write.csv(Clean, file = "SupplementalTable3\_Data-Clean.csv")  
write.csv(Clean\_SoCal, file = "SupplementalTable4\_Data-SoCal.csv")   
  
###### Save cleaned-up data for downstream visualization and analysis ######

Figure 1a-b: Maps of shell locations (Clean and SoCal Data)

# install.packages("ggplot2")  
library("ggplot2")  
# packageVersion("ggplot2") # v3.1.0.9000  
# devtools::install\_github("dkahle/ggmap")   
library("ggmap")

## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.

## Please cite ggmap if you use it! See citation("ggmap") for details.

# packageVersion("ggmap") # v3.0.0.900  
# install.packages("rgdal")  
library("rgdal")

## Loading required package: sp

## rgdal: version: 1.3-6, (SVN revision 773)  
## Geospatial Data Abstraction Library extensions to R successfully loaded  
## Loaded GDAL runtime: GDAL 2.1.3, released 2017/20/01  
## Path to GDAL shared files: /Library/Frameworks/R.framework/Versions/3.5/Resources/library/rgdal/gdal  
## GDAL binary built with GEOS: FALSE   
## Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ\_VERSION: 493]  
## Path to PROJ.4 shared files: /Library/Frameworks/R.framework/Versions/3.5/Resources/library/rgdal/proj  
## Linking to sp version: 1.3-1

# packageVersion("rgdal") # v1.3.6  
  
# This code is personalized by account and only runs if you have a key  
# Instructions:https://github.com/dkahle/ggmap  
# Visit your personal Google Maps Platform to retrieve key if you have one  
ggmap::register\_google(key = "AIzaSyApgtoPtdKVGIi-y1EWLUZ90AD0k6wTMZk") # need to use API from PhD thesis (maps set up)  
  
# Create parameters and background maps for Clean and Clean\_SoCal  
# Parameters for All points  
# Subset data by Epoch to color code points  
Anthro = subset(Clean, Epoch=="Anthropocene") # 1492  
Holo = subset(Clean, Epoch=="Holocene") # 1138  
Pleisto = subset(Clean, Epoch=="Pleistocene") # 305  
  
All\_bbox <- make\_bbox(lat = Latitude, lon = Longitude, data = Clean)  
All\_bbox

## left bottom right top   
## -122.9015 25.8435 -114.7285 38.7465

# left bottom right top   
# -122.9015 25.8435 -114.7285 38.7465   
  
# Load polygons for ecoregions and call them after downloaded in your folder  
# shape files from http://www.marineregions.org/  
NC\_poly=readOGR( "northern\_ecoregions" , layer="ecoregions")

## OGR data source with driver: ESRI Shapefile   
## Source: "/Users/sul-hasan/Desktop/MSa/Mar24-2019/northern\_ecoregions", layer: "ecoregions"  
## with 1 features  
## It has 6 fields

SCB\_poly=readOGR( "southern\_ecoregions" , layer="ecoregions")

## OGR data source with driver: ESRI Shapefile   
## Source: "/Users/sul-hasan/Desktop/MSa/Mar24-2019/southern\_ecoregions", layer: "ecoregions"  
## with 1 features  
## It has 6 fields

MT\_poly=readOGR( "magdaelena\_ecoregions" , layer="ecoregions")

## OGR data source with driver: ESRI Shapefile   
## Source: "/Users/sul-hasan/Desktop/MSa/Mar24-2019/magdaelena\_ecoregions", layer: "ecoregions"  
## with 1 features  
## It has 6 fields

All=get\_map(location=c(-125,20,-105,40), # adjust to include regions  
 zoom = 5,   
 maptype = "terrain")

## Bounding box given to Google - spatial extent only approximate.

## Source : https://maps.googleapis.com/maps/api/staticmap?center=30,-115&zoom=5&size=640x640&scale=2&maptype=terrain&language=en-EN&key=xxx-y1EWLUZ90AD0k6wTMZk

SoCal\_bbox = make\_bbox(lat = Latitude, lon = Longitude, data = Clean\_SoCal)  
SoCal\_bbox

## left bottom right top   
## -120.6685 32.4635 -116.9615 34.5865

# left bottom right top   
# -120.6695 32.4425 -116.9405 34.5875   
  
SoCal=get\_map(location=c(-121,32,-116.5,35),   
 zoom = 7,   
 maptype = "terrain")

## Bounding box given to Google - spatial extent only approximate.

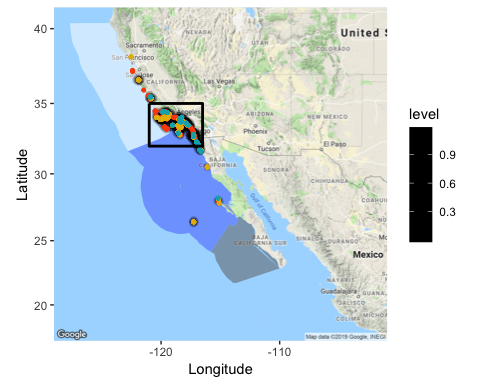
## Source : https://maps.googleapis.com/maps/api/staticmap?center=33.5,-118.75&zoom=7&size=640x640&scale=2&maptype=terrain&language=en-EN&key=xxx-y1EWLUZ90AD0k6wTMZk

# Figure 1a  
SoCal\_box = data.frame(x=c(-121,-121,-116.5,-116.5,-121),   
 y=c(35,32,32,35,35)) # create box for zoom of SoCal  
  
Clean\_map = ggmap(All) +  
 geom\_polygon(data = NC\_poly, aes(x = long, y = lat),  
 fill = "white", alpha = 0.5) + # white  
 geom\_polygon(data = SCB\_poly, aes(x = long, y = lat),  
 fill = "blue", alpha = 0.3) + # blue  
 geom\_polygon(data = MT\_poly, aes(x = long, y = lat),  
 fill = "#666666", alpha = 0.5) + # grey  
 stat\_density\_2d(data = Clean, # density  
 aes(x = Longitude,  
 y = Latitude,  
 fill = stat(level)),  
 alpha = .25,  
 bins = 250,  
 geom = "polygon") +  
 scale\_fill\_gradientn(colors = "black") + # color fill, orange  
 geom\_point(data = Anthro, aes(x = Longitude, y = Latitude),   
 color = "#FC4E07",  
 size = 1) + # aqua  
 geom\_point(data = Holo, aes(x = Longitude,y = Latitude),   
 color = "#E7B800",   
 size = 1) + # seagreen  
 geom\_point(data = Pleisto, aes(x = Longitude, y = Latitude),   
 color = "#00AFBB",  
 size = 1) + # blue   
 geom\_path(data=SoCal\_box, aes(x,y), color="black", lwd=1) + # insert box  
 labs(x="Longitude", y="Latitude")

## Regions defined for each Polygons

## Regions defined for each Polygons  
## Regions defined for each Polygons

Clean\_map



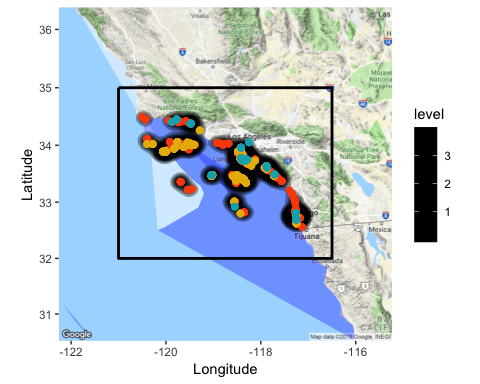
ggsave("Figure1a\_Data-Clean\_Map.png", dpi = 800)

## Saving 5 x 4 in image

# Figure 1b  
Anthro\_SoCal = subset(Clean\_SoCal, Epoch=="Anthropocene") # 1366  
Holo\_SoCal = subset(Clean\_SoCal, Epoch=="Holocene") # 1020  
Pleisto\_SoCal = subset(Clean\_SoCal, Epoch=="Pleistocene") # 211  
  
CleanSoCal\_map = ggmap(SoCal) +  
 geom\_polygon(data = NC\_poly, aes(x = long, y = lat),  
 fill = "white", alpha = 0.5) + # white  
 geom\_polygon(data = SCB\_poly, aes(x = long, y = lat),  
 fill = "blue", alpha = 0.3) + # blue  
 stat\_density\_2d(data = Clean\_SoCal, # density  
 aes(x = Longitude,  
 y = Latitude,  
 fill = stat(level)),  
 alpha = .35,  
 bins = 250,  
 geom = "polygon") +  
 scale\_fill\_gradientn(colors = "black") + # color fill, orange  
 geom\_point(data = Anthro\_SoCal, aes(x = Longitude, y = Latitude),   
 color = "#FC4E07",  
 size = 2) +   
 geom\_point(data = Holo\_SoCal, aes(x = Longitude,y = Latitude),   
 color = "#E7B800",   
 size = 2) +   
 geom\_point(data = Pleisto\_SoCal, aes(x = Longitude, y = Latitude),   
 color = "#00AFBB",  
 size = 2) +   
 geom\_path(data=SoCal\_box, aes(x,y), color="black", lwd=1) + # insert box  
 labs(x="Longitude", y="Latitude")

## Regions defined for each Polygons  
## Regions defined for each Polygons

CleanSoCal\_map



ggsave("Figure1b\_Data-Clean-SoCal\_Map.png", dpi = 800)

## Saving 5 x 4 in image

Figure 2b: Allometry - shell shape vs size (Clean)

# install.packages("GGally")  
library(GGally)

##   
## Attaching package: 'GGally'

## The following object is masked from 'package:dplyr':  
##   
## nasa

# packageVersion("GGally") # v1.4.0  
# install.packages("ggplot2")  
library(ggplot2)  
# packageVersion("ggplot2") # v3.1.0  
  
# Create a function for including linear regression line  
my\_fn <- function(data, mapping, ...){  
 p <- ggplot(data = data, mapping = mapping) +   
 geom\_point() +   
 geom\_smooth(method=lm, color="black", ...)   
 p  
}  
  
# Subset data  
Fig2 = Clean[,c("Epoch", "Globosity\_WoverL", "Length\_mm", "Width\_mm")]   
# Rename columns  
names(Fig2) = c("Epoch", "Globosity", "Length", "Width")   
# Create figure (automatically excludes NAs)  
Fig2\_chart = ggpairs(Fig2, aes(color = Epoch), lower = list(continuous = my\_fn), upper = list(continuous = wrap("cor", size = 4.25)))  
# Change colors manually with a loop through each plot  
for(i in 1:Fig2\_chart$nrow) {  
 for(j in 1:Fig2\_chart$ncol){  
 Fig2\_chart[i,j] <- Fig2\_chart[i,j] +   
 scale\_fill\_manual(values=c("#FC4E07", "#E7B800", "#00AFBB")) +  
 scale\_color\_manual(values=c("#FC4E07", "#E7B800", "#00AFBB"))   
 }  
}

## Warning in (function (data, mapping, alignPercent = 0.6, method =  
## "pearson", : Removed 99 rows containing missing values

## Warning in (function (data, mapping, alignPercent = 0.6, method =  
## "pearson", : Removed 99 rows containing missing values

## Warning in (function (data, mapping, alignPercent = 0.6, method =  
## "pearson", : Removed 99 rows containing missing values

Fig2\_chart +   
 theme\_classic() # theme for figure

## Warning: Removed 99 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 99 rows containing non-finite values (stat\_boxplot).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_bin).

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_density).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_bin).

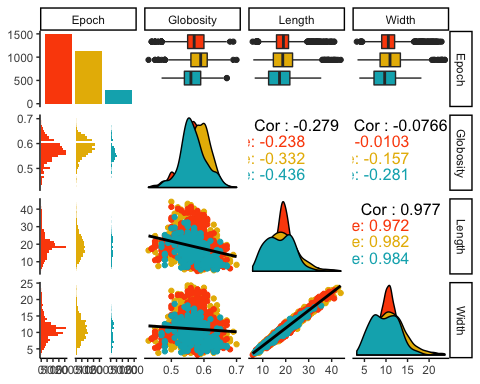
## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_density).



ggsave("Figure2b\_Data-Clean\_Shell-Allometry.png", dpi = 800)

## Saving 5 x 4 in image

## Warning: Removed 99 rows containing non-finite values (stat\_boxplot).

## Warning: Removed 99 rows containing non-finite values (stat\_boxplot).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_bin).

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_density).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_bin).

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

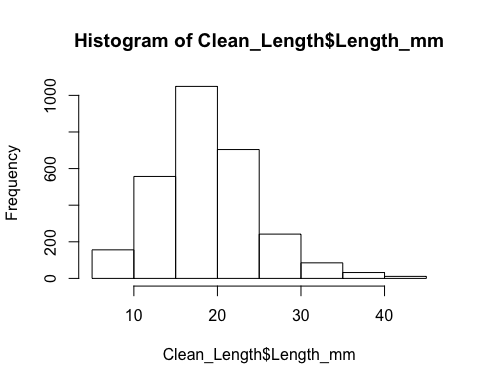
## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_density).

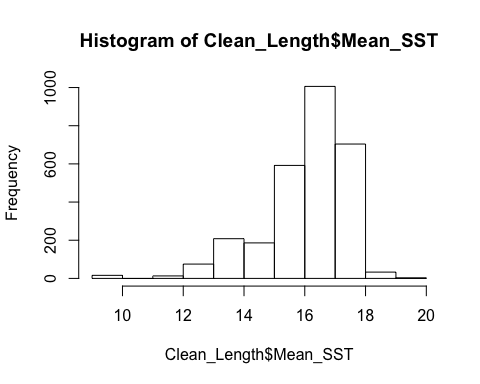
# Length = Width, globosity independent of length and width  
# size = length=width, aperture, thickness (see categories)  
# shape = globosity

Figure 3: Relationship between shell morphology and temperature (Clean)

Clean\_Length = Clean[!is.na(Clean$Length\_mm), ] # remove rows with NAs [iNat]  
  
################################################  
# STATISTICS  
### Distribution of data  
hist(Clean\_Length$Length\_mm) # fairly normal



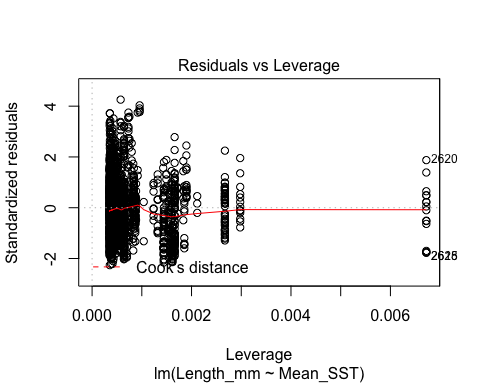
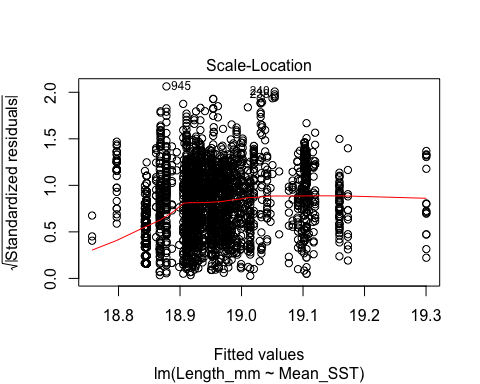
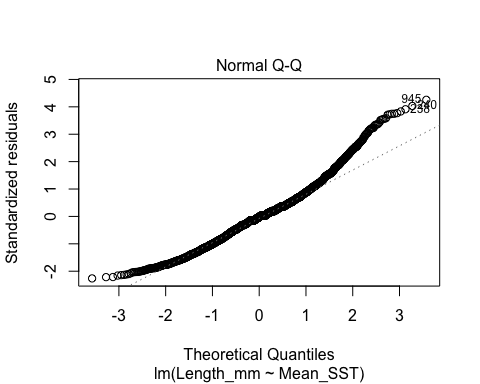
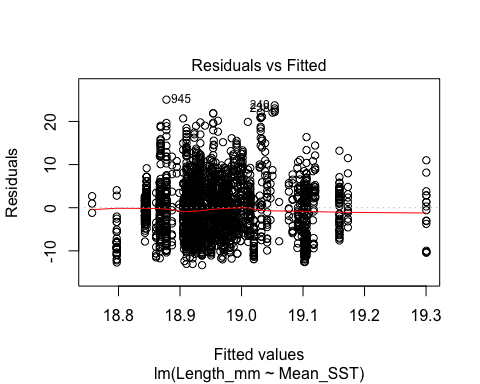
hist(Clean\_Length$Mean\_SST) # slightly skewed



### Parametric (assumes normality)  
# linear model (1st order polynomial)? No  
x = lm(Length\_mm~Mean\_SST, Clean\_Length)  
shapiro.test(residuals(x)) # W = 0.97431, p-value < 2.2e-16

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(x)  
## W = 0.97431, p-value < 2.2e-16

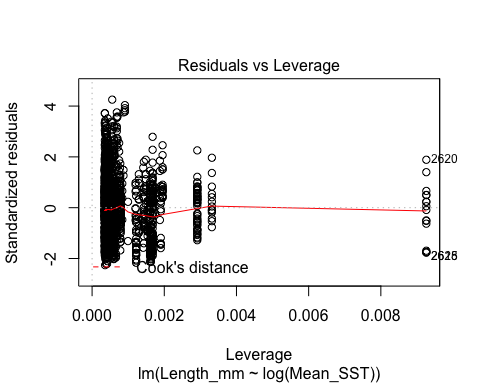
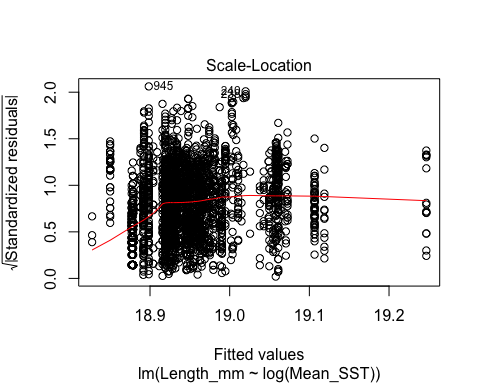
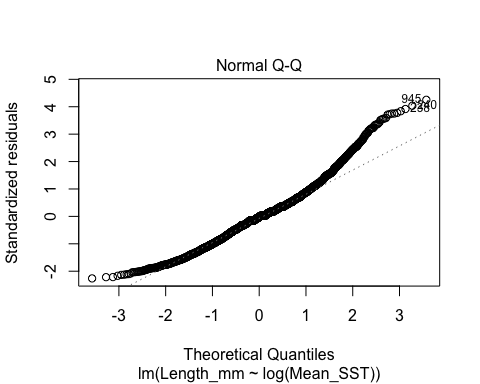
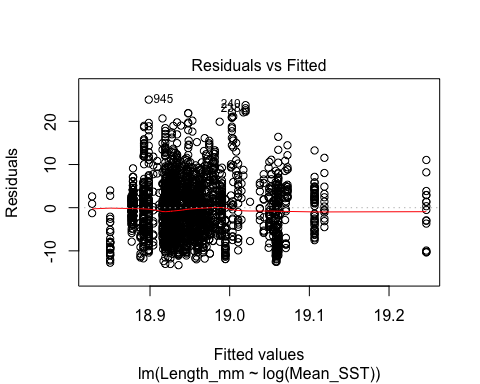
# p needs to be over 0.05  
plot(x) # shows residuals, qqplot not great



# nonlinear models  
# logarithmic model? No (about the same as before)  
ya = lm(Length\_mm~log(Mean\_SST), Clean\_Length) # log transform   
shapiro.test(residuals(ya)) # W = 0.9742, p-value < 2.2e-16

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(ya)  
## W = 0.9742, p-value < 2.2e-16

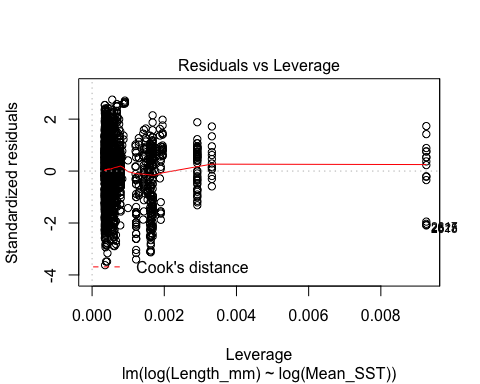
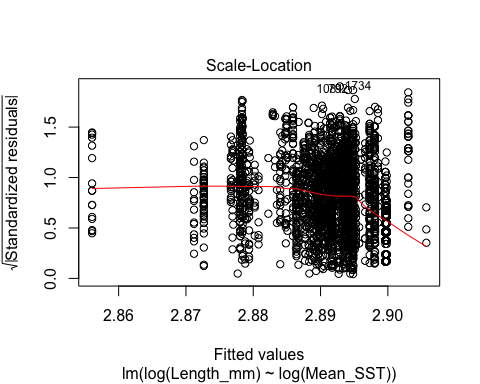
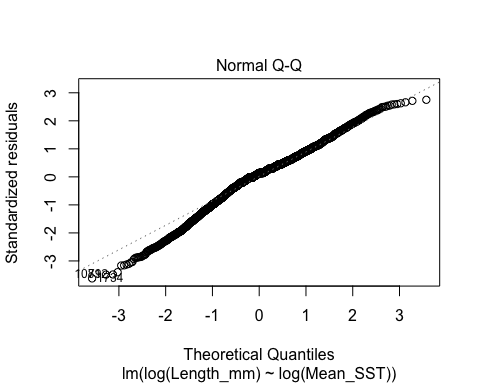
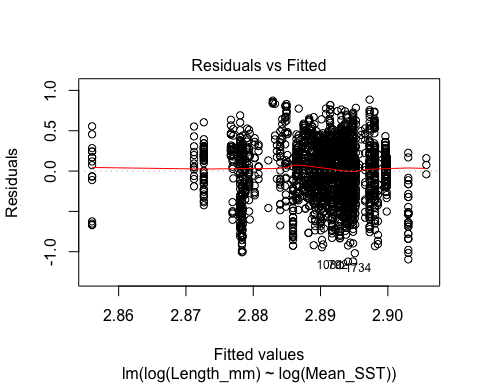
plot(ya)



# log-log model? No (looks decent)  
yb = lm(log(Length\_mm)~log(Mean\_SST), Clean\_Length)   
shapiro.test(residuals(yb)) # W = 0.98499, p-value < 2.2e-16

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(yb)  
## W = 0.98499, p-value < 2.2e-16

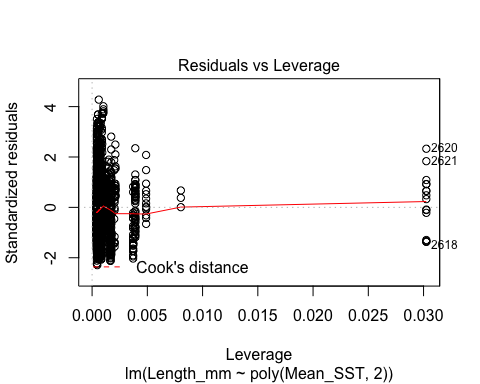
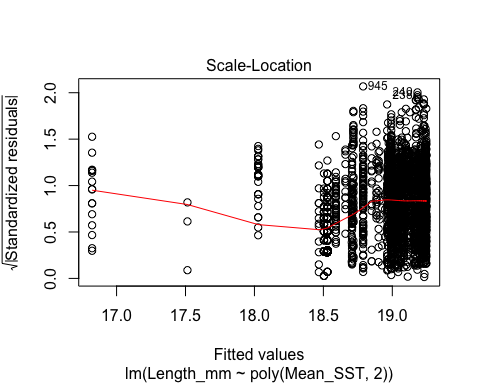
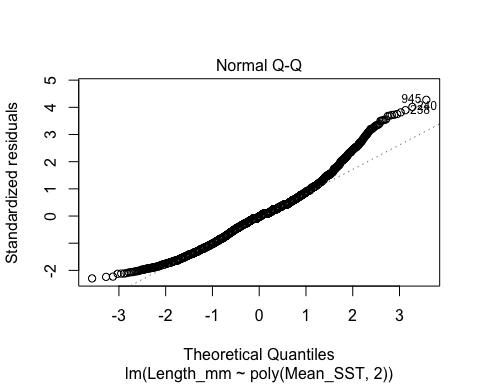
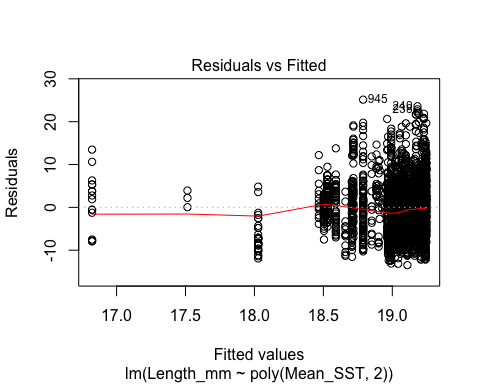
plot(yb)



# 2nd order polynomial model? No  
z = lm(Length\_mm~poly(Mean\_SST,2), Clean\_Length)  
shapiro.test(residuals(z)) # W = 0.97539, p-value < 2.2e-16

##   
## Shapiro-Wilk normality test  
##   
## data: residuals(z)  
## W = 0.97539, p-value < 2.2e-16

plot(z)



summary(x) # r^2 = -0.0001501

##   
## Call:  
## lm(formula = Length\_mm ~ Mean\_SST, data = Clean\_Length)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -13.3159 -3.9427 -0.1087 3.0693 25.0521   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19.83457 1.17968 16.814 <2e-16 \*\*\*  
## Mean\_SST -0.05518 0.07280 -0.758 0.449   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.887 on 2834 degrees of freedom  
## Multiple R-squared: 0.0002027, Adjusted R-squared: -0.0001501   
## F-statistic: 0.5746 on 1 and 2834 DF, p-value: 0.4485

summary(ya) # r^2 = -0.0002482

##   
## Call:  
## lm(formula = Length\_mm ~ log(Mean\_SST), data = Clean\_Length)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -13.3158 -3.9325 -0.1142 3.0661 25.0316   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 20.6038 3.0501 6.755 1.73e-11 \*\*\*  
## log(Mean\_SST) -0.5978 1.0980 -0.544 0.586   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.887 on 2834 degrees of freedom  
## Multiple R-squared: 0.0001046, Adjusted R-squared: -0.0002482   
## F-statistic: 0.2964 on 1 and 2834 DF, p-value: 0.5862

summary(yb) # r^2 = 0.0001369

##   
## Call:  
## lm(formula = log(Length\_mm) ~ log(Mean\_SST), data = Clean\_Length)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.1665 -0.1836 0.0433 0.1956 0.8853   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.69542 0.16684 16.156 <2e-16 \*\*\*  
## log(Mean\_SST) 0.07076 0.06006 1.178 0.239   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.322 on 2834 degrees of freedom  
## Multiple R-squared: 0.0004896, Adjusted R-squared: 0.0001369   
## F-statistic: 1.388 on 1 and 2834 DF, p-value: 0.2388

summary(z) # r^2 = 0.002156 (best, but not by a lot...)

##   
## Call:  
## lm(formula = Length\_mm ~ poly(Mean\_SST, 2), data = Clean\_Length)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -13.4895 -4.0518 -0.1517 3.0831 25.1416   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 18.9443 0.1104 171.571 < 2e-16 \*\*\*  
## poly(Mean\_SST, 2)1 -4.4623 5.8802 -0.759 0.44799   
## poly(Mean\_SST, 2)2 -16.1555 5.8802 -2.747 0.00604 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.88 on 2833 degrees of freedom  
## Multiple R-squared: 0.00286, Adjusted R-squared: 0.002156   
## F-statistic: 4.062 on 2 and 2833 DF, p-value: 0.01731

# 2nd order (z) didn't pass Shapiro's test (log-log), qqplots look bad  
  
### Non-parametric (assumes non-normal, but weaker statistic)  
cor.test(Clean\_Length$Length\_mm, Clean\_Length$Mean\_SST, method = "spearman")

## Warning in cor.test.default(Clean\_Length$Length\_mm,  
## Clean\_Length$Mean\_SST, : Cannot compute exact p-value with ties

##   
## Spearman's rank correlation rho  
##   
## data: Clean\_Length$Length\_mm and Clean\_Length$Mean\_SST  
## S = 3885200000, p-value = 0.2418  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho   
## -0.02198427

# S = 3885200000, p-value = 0.2418, rho = -0.02198427 ('r^2')  
# p value needs to be less than 0.05 to be significant  
# data is linear  
  
################################################  
  
# Subset data  
Fig3 = Clean[,c("Epoch", "Globosity\_WoverL", "Length\_mm", "Mean\_SST")]   
# Rename columns  
names(Fig3) = c("Epoch", "Globosity", "Length", "Mean SST")   
# Create figure (automatically excludes NAs)  
Fig3\_chart = ggpairs(Fig3, aes(color = Epoch), lower = list(continuous = my\_fn), upper = list(continuous = wrap("cor", size = 4.25)))  
# Change colors manually with a loop through each plot  
for(i in 1:Fig3\_chart$nrow) {  
 for(j in 1:Fig3\_chart$ncol){  
 Fig3\_chart[i,j] <- Fig3\_chart[i,j] +   
 scale\_fill\_manual(values=c("#FC4E07", "#E7B800", "#00AFBB")) +  
 scale\_color\_manual(values=c("#FC4E07", "#E7B800", "#00AFBB"))   
 }  
}

## Warning in (function (data, mapping, alignPercent = 0.6, method =  
## "pearson", : Removed 99 rows containing missing values

## Warning in (function (data, mapping, alignPercent = 0.6, method =  
## "pearson", : Removed 99 rows containing missing values

Fig3\_chart +   
 theme\_classic() # theme for figure

## Warning: Removed 99 rows containing non-finite values (stat\_boxplot).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_bin).

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

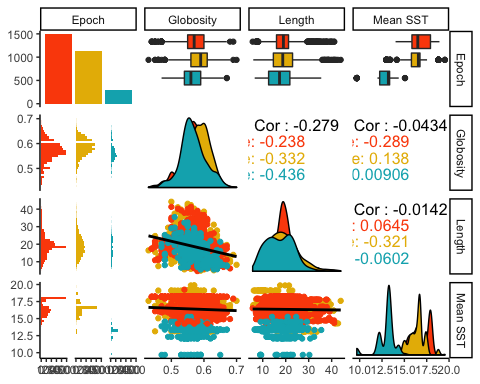
## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_density).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).



ggsave("Figure3\_Data-Clean\_Bergmann.png", dpi = 800)

## Saving 5 x 4 in image

## Warning: Removed 99 rows containing non-finite values (stat\_boxplot).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_bin).

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).

## Warning: Removed 99 rows containing non-finite values (stat\_density).

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 99 rows containing non-finite values (stat\_smooth).

## Warning: Removed 99 rows containing missing values (geom\_point).

Figure 4: Relationships morphology, humans, between tempearture and shell morphology (SoCal)

temp\_socal = read.csv("SupplementalTable4\_Data-SoCal.csv", header=T, na.strings=c("","NA"))  
  
pop\_socal = read.csv("SupplementalTable3\_Data-CleanSabah.csv", header=T, na.strings=c("","NA"))  
  
chart.Correlation <- function (R, histogram = TRUE, method = c("pearson", "kendall",   
 "spearman"), ...)   
{  
 x = checkData(R, method = "matrix")  
 if (missing(method))   
 method = method[1]  
 panel.cor <- function(x, y, digits = 2, prefix = "", use = "pairwise.complete.obs",   
 method = "pearson", cex.cor, ...) {  
 usr <- par("usr")  
 on.exit(par(usr))  
 par(usr = c(0, 1, 0, 1))  
 r <- cor(x, y, use = use, method = method)  
 txt <- format(c(r, 0.123456789), digits = digits)[1]  
 txt <- paste(prefix, txt, sep = "")  
 if (missing(cex.cor))   
 cex <- 0.8/strwidth(txt)  
 test <- cor.test(as.numeric(x), as.numeric(y), method = method)  
 Signif <- symnum(test$p.value, corr = FALSE, na = FALSE,   
 cutpoints = c(0, 0.001, 0.01, 0.05, 0.1, 1), symbols = c("\*\*\*",   
 "\*\*", "\*", ".", " "))  
 text(0.5, 0.5, txt, cex = cex \* (abs(r) + 0.3)/1.3)  
 text(0.8, 0.8, Signif, cex = cex, col = 2)  
 }  
 f <- function(t) {  
 dnorm(t, mean = mean(x), sd = sd.xts(x))  
 }  
 dotargs <- list(...)  
 dotargs$method <- NULL  
 rm(method)  
 hist.panel = function(x, ... = NULL) {  
 par(new = TRUE)  
 hist(x, col = "light gray", probability = TRUE, axes = FALSE,   
 main = "", breaks = "FD")  
 lines(density(x, na.rm = TRUE), col = "red", lwd = 1)  
 rug(x)  
 }  
 if (histogram)   
 pairs(x, gap = 0, lower.panel = panel.smooth, upper.panel = panel.cor,   
 diag.panel = hist.panel, ...)  
 else pairs(x, gap = 0, lower.panel = panel.smooth, upper.panel = panel.cor, ...)  
}   
  
test = merge(temp\_socal, pop\_socal, by = "Specimen\_Number")  
Fig4 = c("Length\_mm.x", "Globosity\_WoverL.x", "Location.x", "Land\_Category.x", "Population\_Density", "Mean\_SST")  
test2 = test %>% select (Fig4)  
test3 = transform(test2,   
 Length\_mm.x = as.numeric(Length\_mm.x),  
 Globosity\_WoverL.x = as.numeric(Globosity\_WoverL.x),  
 Location.x = as.numeric(Location.x),  
 Land\_Category.x = as.numeric(Land\_Category.x),  
 Population\_Density = as.numeric(Population\_Density),  
 Mean\_SST = as.numeric(Mean\_SST))  
  
chart.Correlation(test3, histogram = T, pch = 22, col = "gray", lwd = 1.5, cex.axis=1, labels=c("Size", "Shape", "Site", "Coastal Proximity", "Human population", "Temperature"), cex.labels=1, lab.cols = 2)

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter  
  
## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "lab.cols" is not a  
## graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter  
  
## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

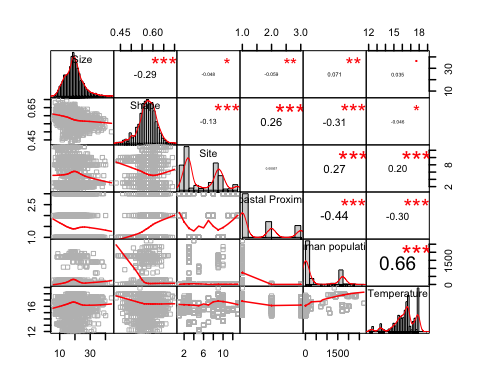
## Warning in title(...): "lab.cols" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "lab.cols" is  
## not a graphical parameter

## Warning in plot.window(...): "lab.cols" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "lab.cols" is not a graphical parameter

## Warning in title(...): "lab.cols" is not a graphical parameter



dev.copy(png, "Figure4\_SoCal.png")

## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter

## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter  
  
## Warning in dev.copy(png, "Figure4\_SoCal.png"): "lab.cols" is not a  
## graphical parameter

## quartz\_off\_screen   
## 3

dev.off()

## quartz\_off\_screen   
## 2