Thames estuary tables and figures

Duncan Golicher

12/1/2018

## Data wrangling

Neadless to say the data come in a fairly horrendous spreadsheet consisting of a sheet per site, with the species abundances placed as rows with column headers for month and tide state split over two rows.

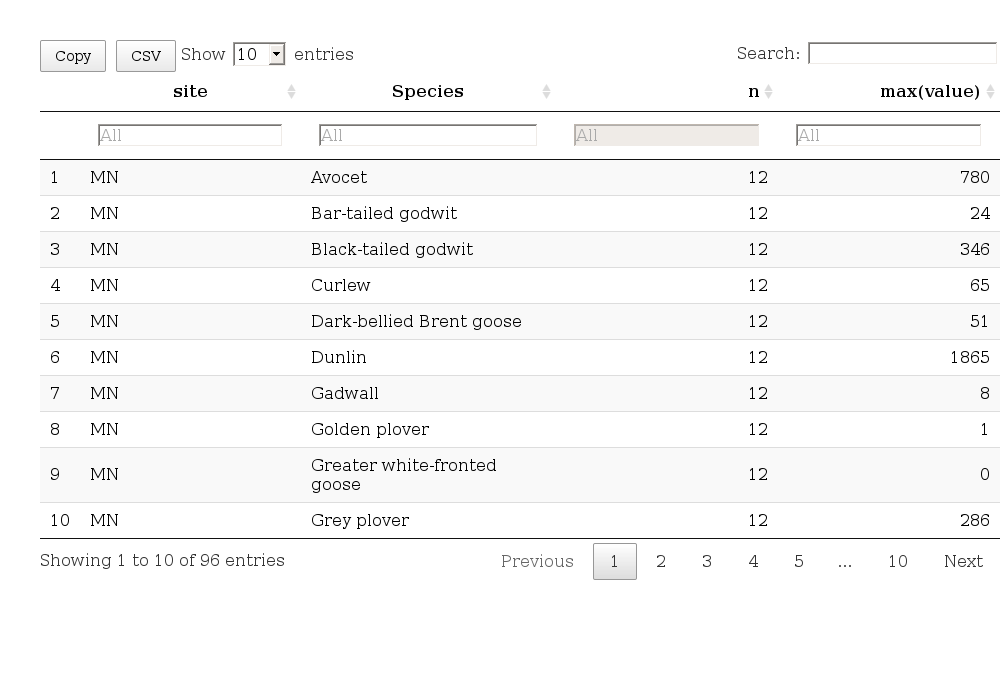
library(readxl)  
library(reshape2)  
library(DT)  
library(dplyr)  
library(ggplot2)  
library(ggthemes)  
  
library(DT)  
  
dt<-function(x) DT::datatable(x,   
 filter = "top",   
 extensions = c('Buttons'), options = list(  
 dom = 'Blfrtip',  
 buttons = c('copy', 'csv', 'excel')))  
  
mns<-month.name[c(10:12,1:3)] ## Make a vector of names for the columns  
mns<-rep(mns,each=2) ## The months are repeated twice  
tides<-rep(c("Hi","Lw"),times=6) ## There are six months each with a hi and low record  
nms<-paste(tides,mns,sep="\_") ## Paste together to get the names  
nms<-c("Species",nms) ## First colummn has the species  
  
  
MS<- read\_excel("6541.1 DPW Winter Bird Summary MuckingsStanford WharfSiteX 2017-18 V1.xlsx",   
 sheet = 1, col\_names = TRUE,   
 skip = 3)[,1:13]  
  
names(MS)<-nms  
  
MN<- read\_excel("6541.1 DPW Winter Bird Summary MuckingsStanford WharfSiteX 2017-18 V1.xlsx",   
 sheet = 2, col\_names = TRUE,   
 skip = 3)[,1:13]  
  
names(MN)<-nms  
  
SF<- read\_excel("6541.1 DPW Winter Bird Summary MuckingsStanford WharfSiteX 2017-18 V1.xlsx",   
 sheet = 3, col\_names = TRUE,   
 skip = 3)[,1:13]  
  
names(SF)<-nms  
  
SW<- read\_excel("6541.1 DPW Winter Bird Summary MuckingsStanford WharfSiteX 2017-18 V1.xlsx",   
 sheet = 4, col\_names = TRUE,   
 skip = 3)[,1:13]  
  
names(SW)<-nms

Now melt each data frame with the consistent column names and combine them into a single table. Split out the month and tide state to form two columns.

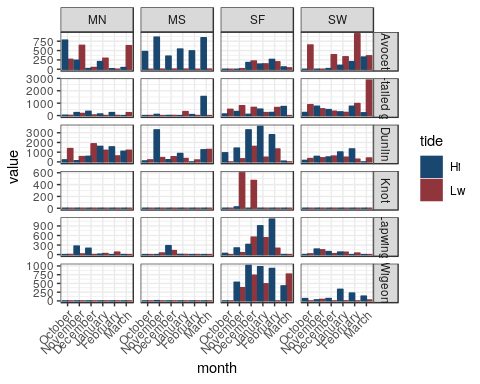
MN<-data.frame(site="MN",melt(MN,"Species"))  
MS<-data.frame(site="MS",melt(MS,"Species"))  
SF<-data.frame(site="SF",melt(SF,"Species"))  
SW<-data.frame(site="SW",melt(SW,"Species"))  
  
d<-rbind(MN,MS,SF,SW)  
  
## Split up the variable names into tide and month.  
d$tide<-substr(d$variable,1,2)  
d$month<-substr(d$variable,4,20)  
d$month<-factor(d$month,levels=month.name[c(10:12,1:3)])  
d$value<-as.numeric(d$value)  
d$value[is.na(d$value)]<-0

## Maximum counts of species per site

d %>% group\_by(site,Species) %>% summarise(n=n(),max(value)) %>% dt()



theme\_set(theme\_bw())  
dd<-droplevels(subset(d,d$value>500))  
tops<-unique(dd$Species)  
dd<-droplevels(subset(d,d$Specie%in%tops))  
dd$Species<-as.factor(dd$Species)  
  
ddSpecies<-factor(dd$Species,levels=levels(dd$Species)[c(3,2,6,1,4,5)])  
library(ggplot2)  
  
g1<-ggplot(dd,aes(x=month,y=value,col=tide,fill=tide))  
g1 + geom\_bar(position="dodge",stat= "identity") +facet\_grid(Species~site,scales = "free") + theme(axis.text.x = element\_text(angle = 45, hjust = 1)) + scale\_fill\_stata(scheme = "s2color") +  
scale\_color\_stata(scheme = "s2color")



## Maximum abundance of each species summed over the sites

d %>% group\_by(site,Species,month) %>% summarise(max=max(value)) %>% group\_by(Species,month) %>% summarise(total=sum(max)) %>% group\_by(Species) %>% summarise(Peak=max(total)) %>% dt()



## Just taking the low water

d %>% filter(tide=="Lw") %>% group\_by(site,Species,month) %>% summarise(max=max(value)) %>% group\_by(Species,month) %>% summarise(total=sum(max)) %>% group\_by(Species) %>% summarise(Peak=max(total)) -> peaks  
dt(peaks)



## Historical data

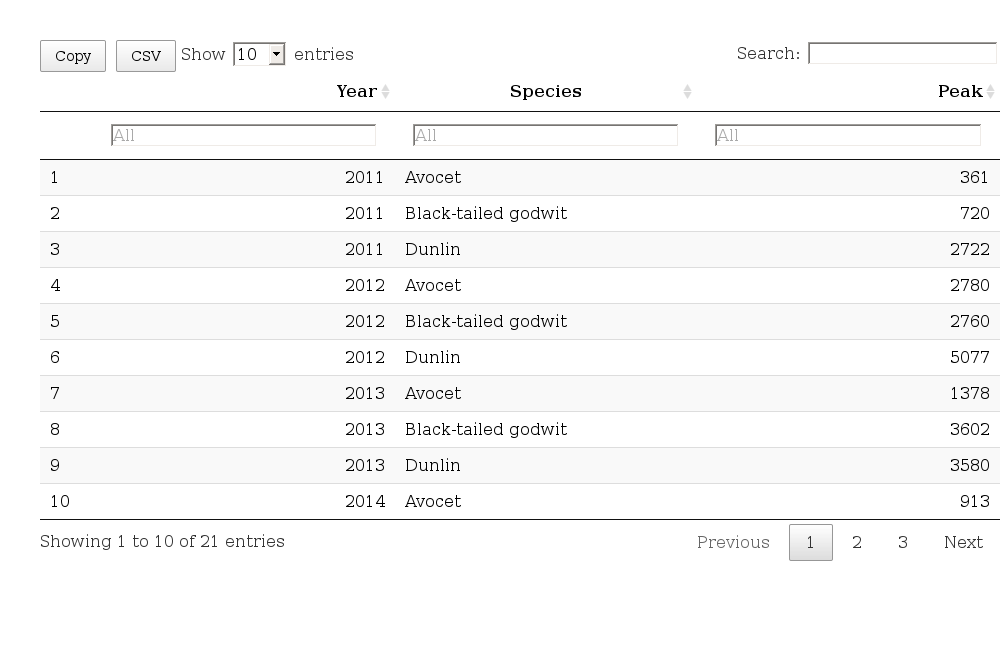
I think this is the right file. It has totals per site, and columns for month and year. There is only one value per month per site. So gruping by year, month and species and taking the sum should be more or less comparable with the procedure above to get the monthly total. The taking the maximum should also produce the peak abundances. Rbind the data frames together to get one.

d2<-read.csv("LondonGatewayDataFINAL\_2017.CSV")  
d2$Date<-as.Date(d2$Date, "%d/%m/%Y")  
names(d2)

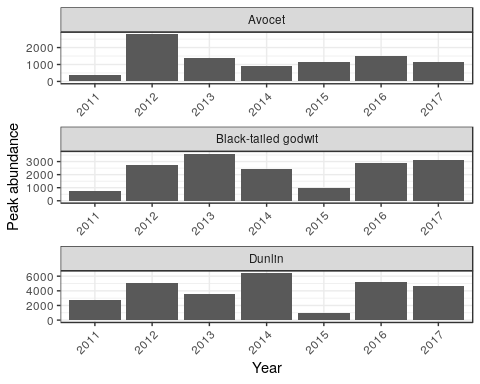
## [1] "Site" "Date" "Month" "Winter" "Period" "Number" "Spp"   
## [8] "Species" "Source"

d2 %>% group\_by(Winter,Month,Species) %>% summarise(msum=sum(Number)) %>% group\_by(Year=Winter,Species) %>% summarise(Peak=max(msum)) -> ypeaks  
  
peaks<-data.frame(Year=2017,peaks)  
ypeaks<-as.data.frame(ypeaks)  
ypeaks<-rbind(ypeaks,peaks)   
ypeaks$Year<-as.numeric(ypeaks$Year)

assemblage\_species<-c("Avocet","Black-tailed godwit", "Dunlin")  
  
ypeaks %>% filter(Species %in% assemblage\_species) %>% filter(Year>2010)-> as\_sp  
  
dt(as\_sp)

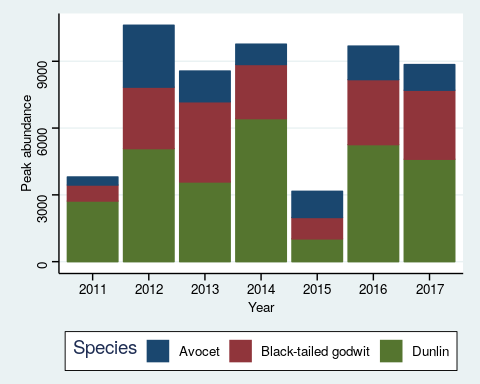


g1<-ggplot(as\_sp,aes(x=as.character(Year),y=Peak))  
g1 + geom\_bar(stat= "identity") +facet\_wrap(~Species,scales = "free",ncol=1) +theme(axis.text.x = element\_text(angle = 45, hjust = 1)) +xlab("Year") + ylab("Peak abundance")



Need to change the horrible colours!

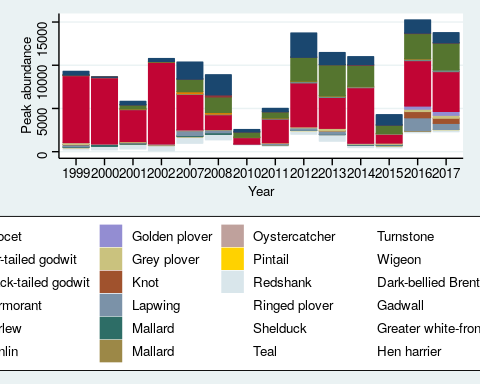
g1<-ggplot(as\_sp,aes(x=as.character(Year),y=Peak, col=Species, fill=Species))  
g1 + geom\_bar(stat= "identity") +theme(axis.text.x = element\_text(angle = 45, hjust = 1)) +xlab("Year") + ylab("Peak abundance") + theme\_stata() + scale\_fill\_stata(scheme = "s2color") +  
scale\_color\_stata(scheme = "s2color")



## Using all species, even rare ones

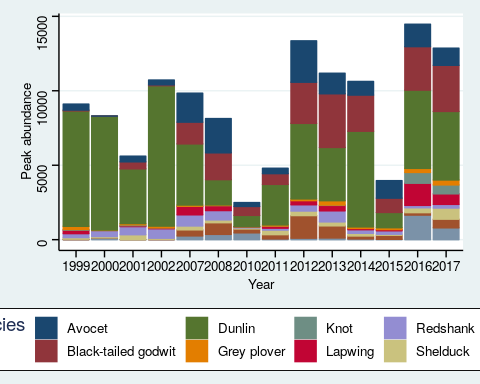
Figure looks a bit odd and has too large a legend, but similar pattern. Can be refined if useful.

g1<-ggplot(ypeaks,aes(x=as.character(Year),y=Peak, col=Species, fill=Species))  
g1 + geom\_bar(stat= "identity") +theme(axis.text.x = element\_text(angle = 45, hjust = 1)) +xlab("Year") + ylab("Peak abundance") + theme\_stata() + scale\_fill\_stata(scheme = "s2color") +  
scale\_color\_stata(scheme = "s2color")



## Take top 10 instead

ypeaks %>% group\_by(Species) %>% summarise(sum=sum(Peak)) %>% arrange(-sum) ->psp  
  
  
g1<-ggplot(filter(ypeaks, Species%in% psp$Species[1:10]),aes(x=as.character(Year),y=Peak, col=Species, fill=Species))  
g1 + geom\_bar(stat= "identity") +theme(axis.text.x = element\_text(angle = 45, hjust = 1)) +xlab("Year") + ylab("Peak abundance") + theme\_stata() + scale\_fill\_stata(scheme = "s2color") +  
scale\_color\_stata(scheme = "s2color")



## Raw data

## Full data table for export and download

dt(d)

