checking BCI species selection estimates against frequencies

preamble

This checks frequency independent selection estimates against changes in frequency estimated by linear regression. Ben's selection estimates are normalized for generation time. So we don't want to use those to predict changes in frequency estimated across census years. Instead, we will start with the *betas* which are estimated in terms of census years and then calculate selection coeffiecients directly from them — but now couched in terms of census years. They should predict the slopes of our linear regressions of frequency ~ census year.

libraries

data

these are the beta estimates. They are not normalized for generation, but are couched in terms of census years

```
beta<-readRDS("beta_freqinde.rds")
beta<-as.data.frame(beta)
dim(beta)
```

```
## [1] 268   3
```

these are the selection estimates. They **are** normalized for generation.

```
sel<-readRDS("estimates_selection.rds")
sel<-as.data.frame(sel)
dim(sel)
```

```
## [1] 269   7
```

The immigration rates: estimated in terms of census years

```
d<-readRDS("immigration.rds")
```

get species names for the beta estimates. The last variant has to have beta=0 to allow the model to be identifiable. We will assume that the selection list, which is alphabetical, is the right order

```
beta$species<-head(sel$species,-1)
```

Add the last species in. By definition its betas are zero

```
species<-tail(sel$species, 1)
last<-as.data.frame(species)
last$beta_freqinde_2.5<-0
last$beta_freqinde_50<-0
last$beta_freqinde_97.5<-0
last<-last%>%
  select(beta_freqinde_2.5,beta_freqinde_50,beta_freqinde_97.5, species)
beta1<-rbind(beta,last)
dim(beta1)
```

```
## [1] 269   4
```

Estimate absolute fitness, W using just the betas. We need an estimate of delta (or mu).We use the median estimate of the frequency independent rate.

```
delta<-d[3,1]
delta
```

```
## [1] 0.0002521865
```

```
beta1$W=(1-delta)*(1+beta1$beta_freqinde_50)
beta1$w=beta1$W/median(beta1$W)
beta1$s=beta1$w-1
```

these selection coefficients should correlate with estimated changes in frequency.

Let's get the frequency data from "reproductives_counts.rds"

```
f<-readRDS("reproductives_counts.rds")
```

get linear estimates of frequencies of Ben's data and compare them to the selection coefficients. First check how many species and census years we have

```
length(unique(f$species))
```

```
## [1] 269
```

```
length(unique(f$censusyear))
```

```
## [1] 8
```

looks like we have all the data in there. Ensure that census year and frequency are numeric

```
f$censusyear<-as.numeric(as.character(f$censusyear))
f$freq<-as.numeric(as.character(f$freq))
```

The selection coefficients, s predict the **relative** or % change in frequency. To estimate that, then, we apply a log10 transform to the frequencies. Since some species have zero counts in some years this means that we will lose some data.

```
Nobs<-f%>%
group_by(species)%>%
summarize(N_obs=length(species))%>%
  arrange(N_obs)
dim(Nobs)
```

```
## [1] 269   2
```

Just as we expect we have 8 observations for each species. Now we remove zero counts so that we can do linear regressions on logged data.

```
f1<-f%>%
filter(!count==0)
Nobs<-f1%>%
group_by(species)%>%
summarize(N_obs=length(species))%>%
  arrange(N_obs)%>%
  filter(N_obs<8)
dim(Nobs)
```

```
## [1] 38  2
```

So now there are 38 species with fewer than 8 observations. These are going to be mostly low-frequency species that go extinct or go extinct and re-immigrate. We'll remove them.

```
remove<-Nobs$species
f1<-f1%>%
filter(!species %in% remove)
```

check how many species remain, and how many obs in each

```
Nobs<-f1%>%
group_by(species)%>%
summarize(N_obs=length(species))%>%
  arrange(N_obs)
dim(Nobs)
```

```
## [1] 231   2
```

```
head(Nobs)
```

```
## # A tibble: 6 x 2
##   species                 N_obs
##   <chr>                   <int>
## 1 Abarema macradenia          8
## 2 Acalypha diversifolia       8
## 3 Acalypha macrostachya       8
## 4 Adelia triloba              8
## 5 Aegiphila panamensis        8
## 6 Alchornea costaricensis     8
```

I have 231/269 species left. log their frequencies.

```
f1$log10_freq<-log10(f1$freq)
```

run the linear model for all species and extract the linear coefficients

```
m<-f1%>%
group_by(species) %>%
  do(model = lm(log10_freq ~ censusyear, data = .))
m1<-as.data.frame(tidy(m, model))
m2<-m1%>%filter(term=="censusyear")
dim(m2)
```

```
## [1] 231   6
```

predict models

```
pred<-augment(m, model)%>%
select(species, censusyear, .fitted, log10_freq)
```

order the species from postive to negative coefficients for nice plotting

```
ord<-m2%>%
select(species, estimate)%>%
  arrange(desc(estimate))
ord$order<-1:nrow(ord)
ord$estimate<-NULL
pred<-merge(pred, ord, by="species")
pred$species<- reorder(pred$species,pred$order)
#levels(pred$species)
```

check that the fits are reasonable

```
#ggplot()+
#geom_point(data=pred, aes(x=censusyear, y=log10_freq))+
#geom_line(data=pred, aes(x=censusyear, y=.fitted))+
#guides(colour=FALSE)+
#theme_classic()+
#theme(aspect.ratio = 1)+
#  facet_wrap(~species, ncol=20, scales="free")+
#ggsave("BCI plot of frequencies with fits.pdf",device = "pdf", scale = 1, width = 100, height = 100, u
```

That looks good. The linear regressions are doing their thing

Merge selection estimates these with the estimates of the linear model coefficients on log10(frequency).

```
m2<-m2%>%
select(species, estimate, p.value)
beta2<-merge(beta1, m2, by="species")
dim(beta2)
```

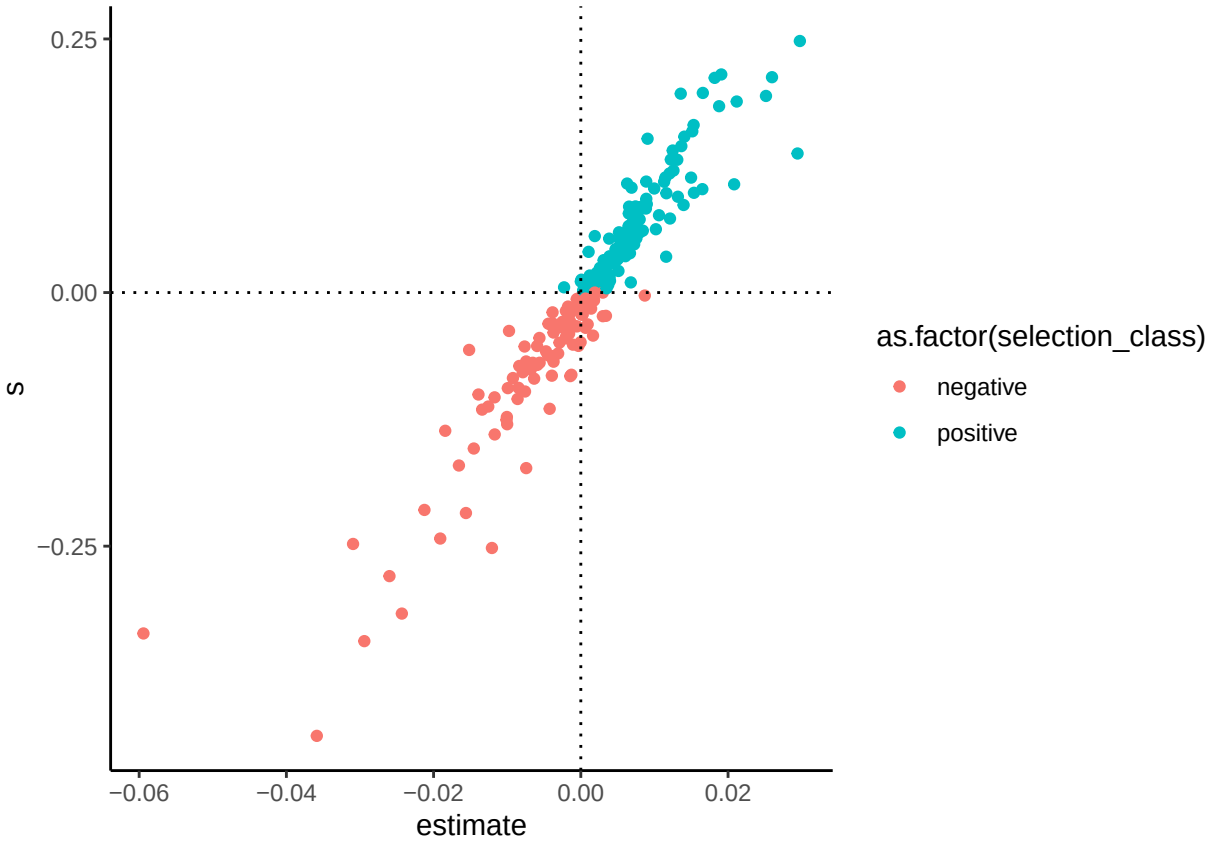
```
## [1] 231   9
```

code according to whether they have positive or negative freqency independent selection coefficients

```
beta2$selection_class<-ifelse(beta2$s>0, "positive", "negative")
```

plot estimates of regression coefficients v. selection coefficients

```
ggplot(data=beta2, aes(x=estimate, y=s, colour=as.factor(selection_class)))+
geom_point()+
theme(aspect.ratio = 1)+
geom_vline(xintercept=0, linetype="dotted")+
geom_hline(yintercept=0, linetype="dotted")+
theme_classic()
```

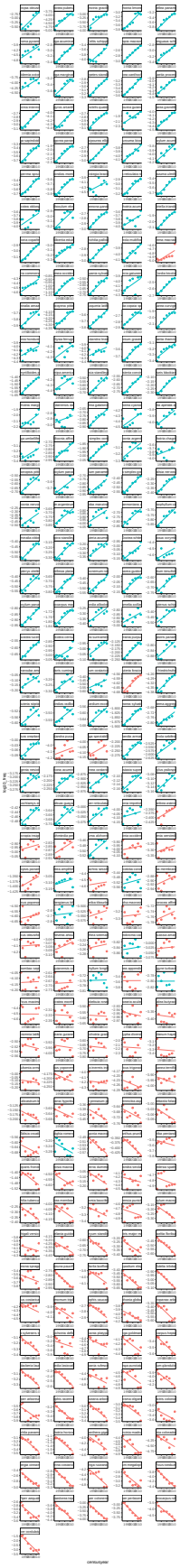


So, linear regression on the **relative** change in frequency – or log10(frequency) – strongly predicts the s coefficents scaled by census year. That's just as it should be.

Look at the individual trajectories now colour coded according to whether they have positive or negative selection coefficients. Merge with frequency data and linear model predictions

```
pred1<-merge(pred, beta2, by="species")
```

```
ggplot()+
geom_point(data=pred1, aes(x=censusyear, y=log10_freq, colour=as.factor(selection_class)))+
geom_line(data=pred1, aes(x=censusyear, y=.fitted, colour=as.factor(selection_class)))+
guides(colour=FALSE)+
theme_classic()+
theme(aspect.ratio = 1)+
  facet_wrap(~species, ncol=5, scales="free")
```

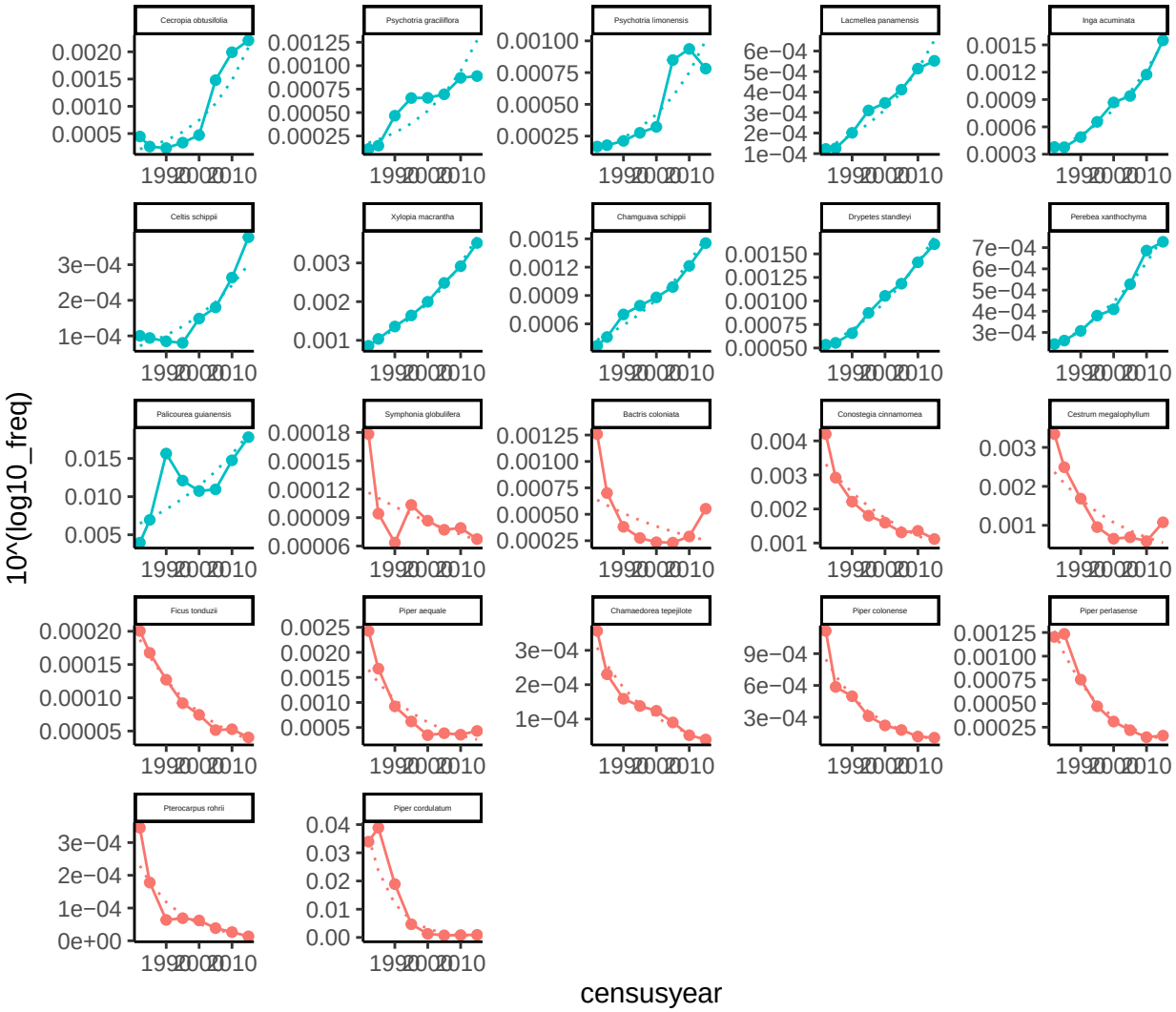


```
#+ggsave("BCI plot of frequencies with fits and selection.pdf",device = "pdf", scale = 1, width = 50, h
```

There are a few mismatches — but these are species which have a small relative change in frequency

Let's look at the species that are increasing and decreasing fastest at BCI. We'll plot the original frequencies rather than the log-transformed ones.

```
pred2<-pred1%>%
filter(s>quantile(s,0.95) | s<quantile(s,0.05))
ggplot()+
geom_point(data=pred2, aes(x=censusyear, y=10^(log10_freq), colour=as.factor(selection_class)))+
geom_line(data=pred2, aes(x=censusyear, y=10^(log10_freq), colour=as.factor(selection_class)))+
geom_line(data=pred2, aes(x=censusyear, y=10^(.fitted), colour=as.factor(selection_class)),linetype="do
guides(colour=FALSE)+
theme_classic()+
theme(aspect.ratio = 1)+
facet_wrap(~species, ncol=5, scales="free")+
  theme(strip.text = element_text(size=3))
```




```
#+ggsave("BCI plot of frequencies with fits and selection.pdf",device = "pdf", scale = 1, width = 50, h
```