Effect plots for hurdle and zero-inflated models

Michael Friendly

Fri Sep 19 15:49:39 2014

```
## load data
library(countreg)
```

```
## Loading required package: MASS
```

```
data("CodParasites", package = "countreg")
## omit NAs in response
CodParasites <- subset(CodParasites, !is.na(intensity))
```

fit some count data models: 2 (Poisson, negbin) x 3 (GLM, hurdle, zeroinfl)

```
cp_p   <-    glm(intensity ~ length + area * year, data = CodParasites, family = poisson)
cp_nb  <- glm.nb(intensity ~ length + area * year, data = CodParasites)
cp_hp  <- hurdle(intensity ~ length + area * year, data = CodParasites, dist = "poisson")
cp_hnb <- hurdle(intensity ~ length + area * year, data = CodParasites, dist = "negbin")
cp_zip <- zeroinfl(intensity ~ length + area * year, data = CodParasites, dist = "poisson")
cp_znb <- zeroinfl(intensity ~ length + area * year, data = CodParasites, dist = "negbin")
```

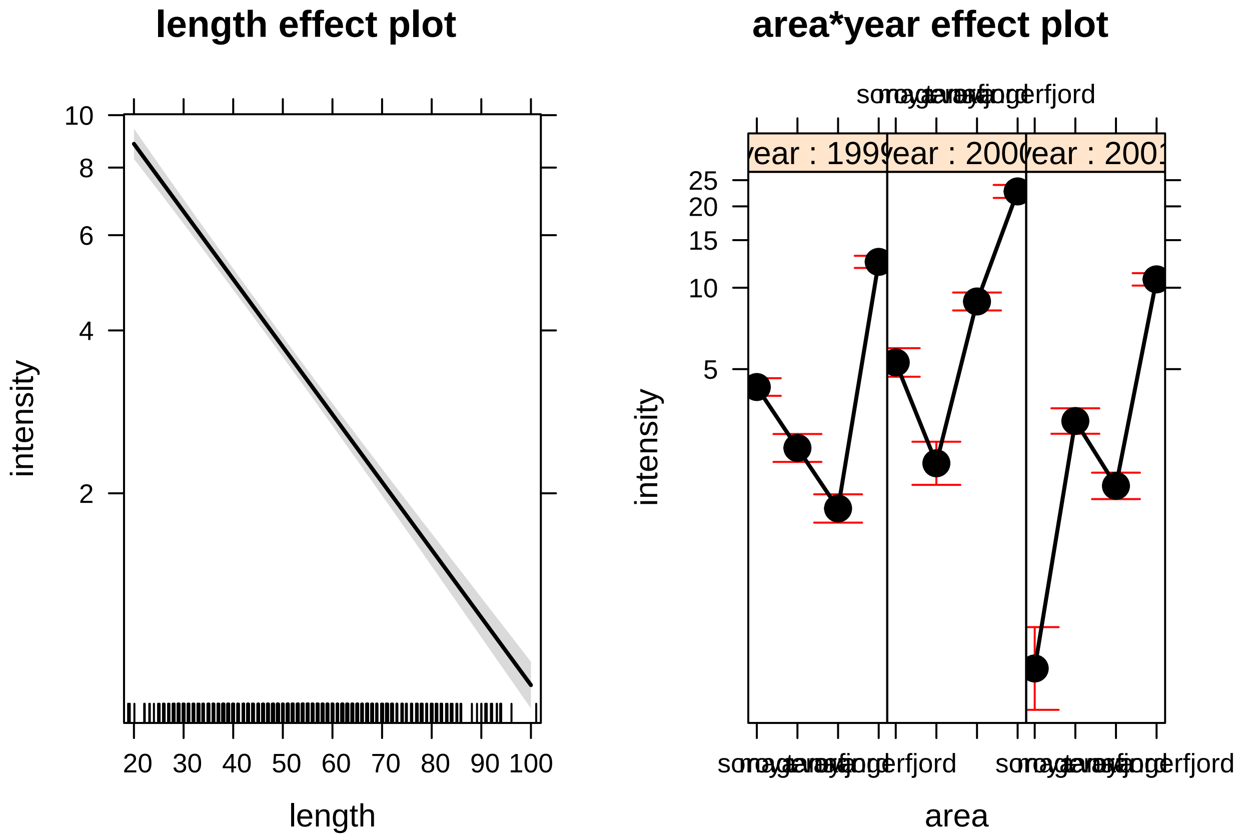
effect displays

These work for standard GLM distributions

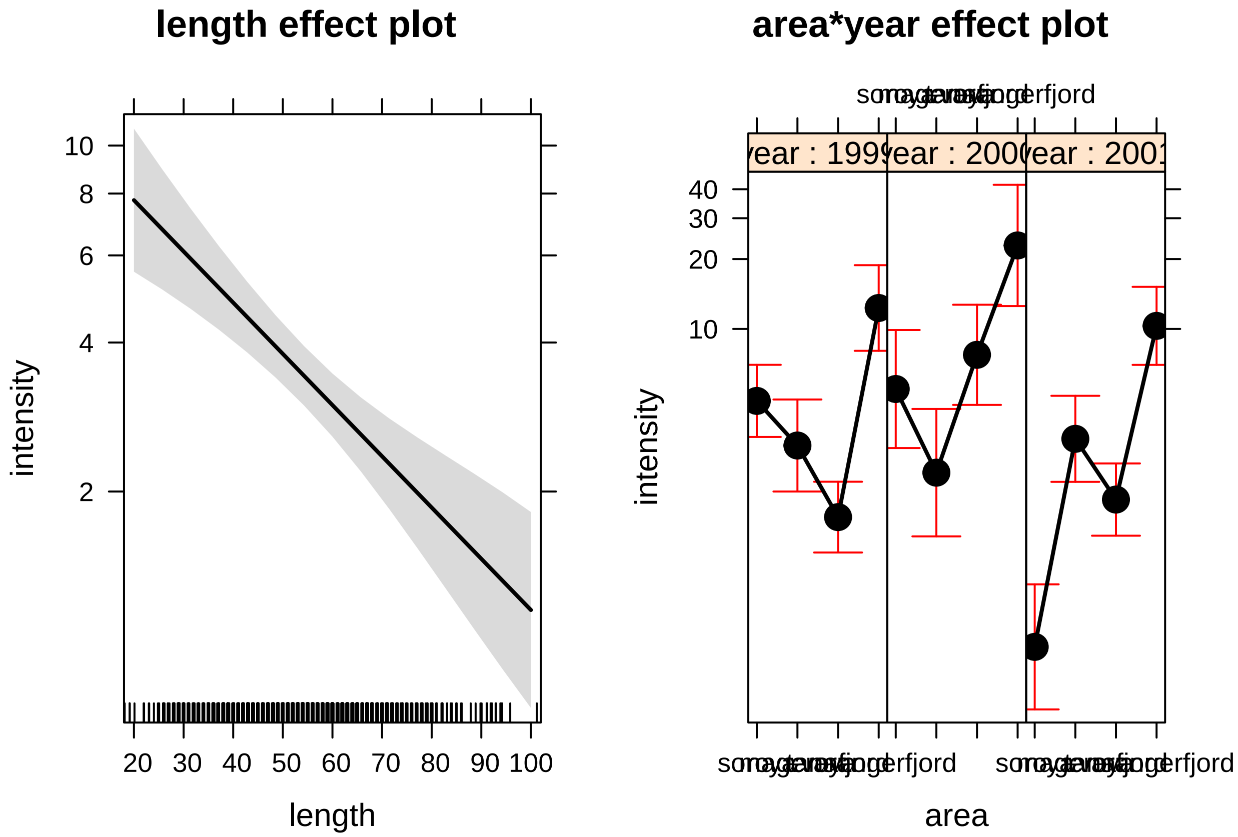
```
library(effects)
```

```
## Loading required package: lattice
## Loading required package: grid
## Loading required package: colorspace
```

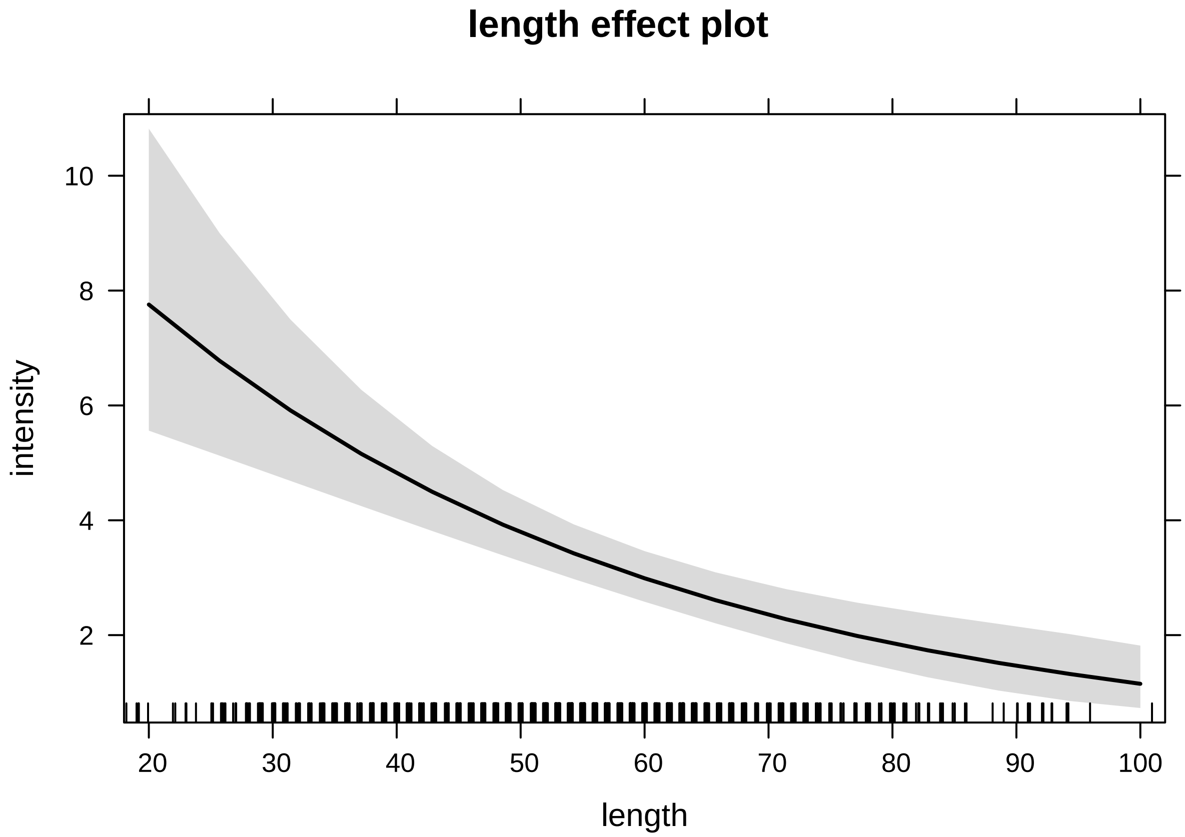
```
plot(allEffects(cp_p))
```



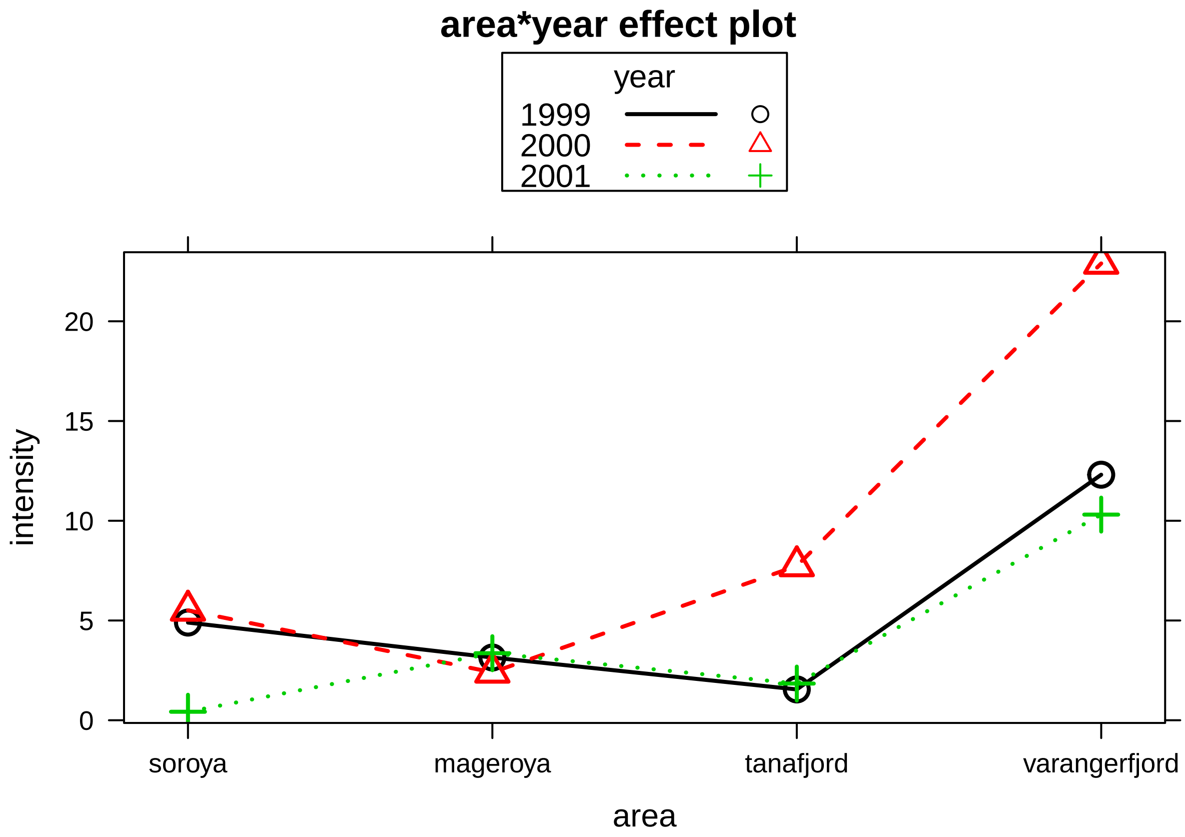
```
plot(allEffects(cp_nb))
```



```
# plot on intensity scale
eff.nb <- allEffects(cp_nb)
plot(eff.nb[[1]], rescale=FALSE)
```



```r
plot(eff.nb[[2]], rescale=FALSE, multiline=TRUE)
```



```
# doesn't work, of course for hurdle or zeroinfl models
#eff_hp <- allEffects(cp_hp)
```

Hurdle models

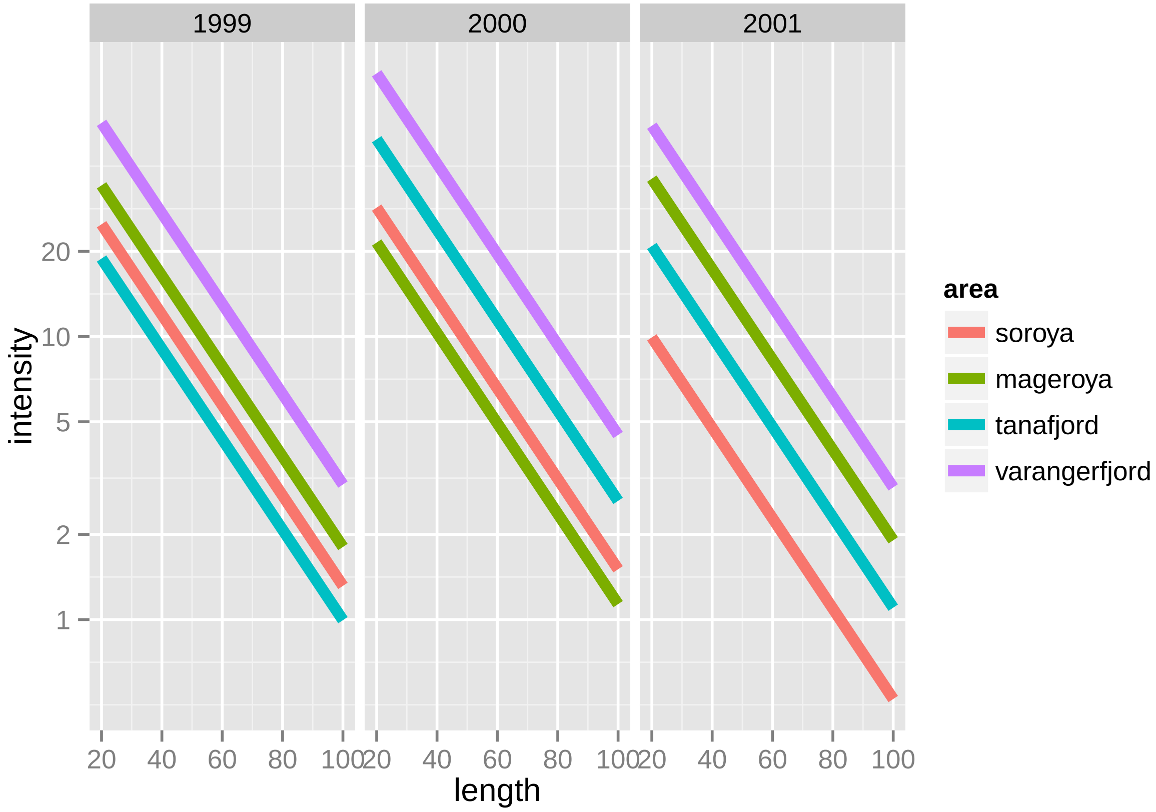
Get the predicted values for count and zero components. Plot these either as full-model plots or as effect plots.

Unfortunately, the `predict` methods don't provide standard errors. Could get these by bootstrapping…

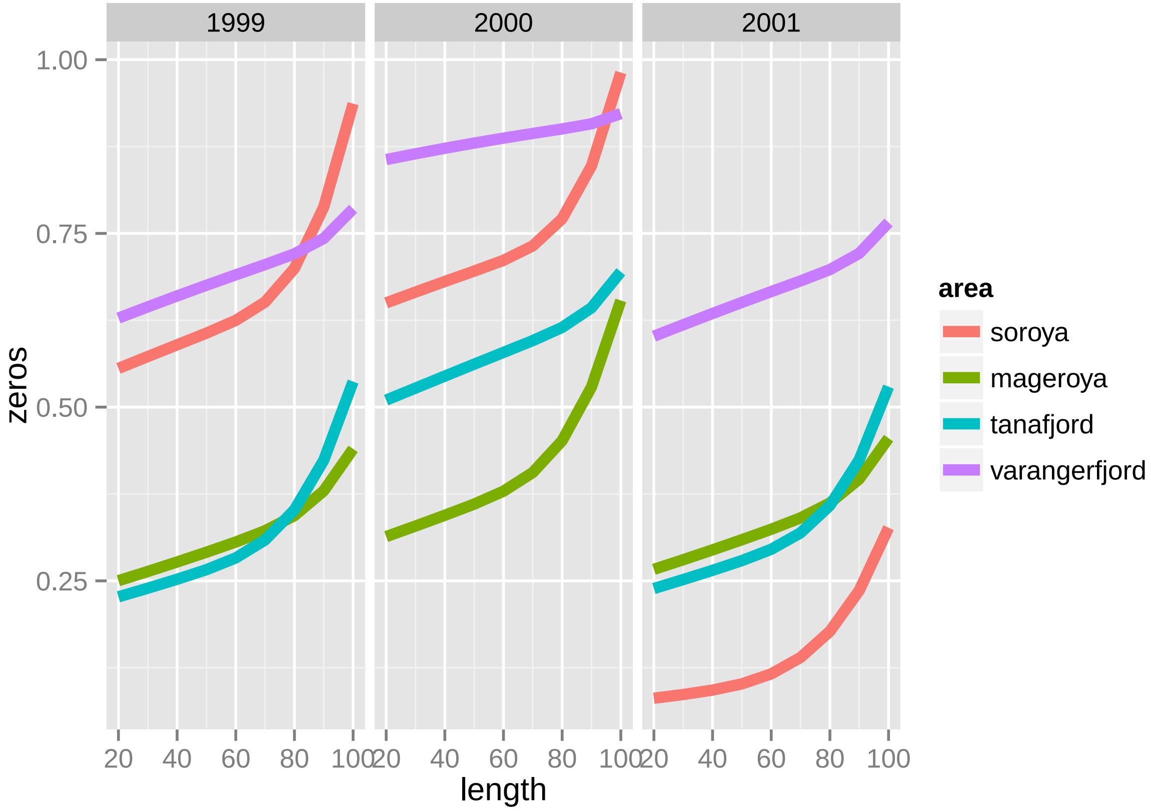
```
# home-brew, full model plot: plot the entire response surface
newdata <- expand.grid( length=seq(20, 100, 10),
                        area=levels(CodParasites$area),
                        year=levels(CodParasites$year))

pred_func <- function(model, newdata) {
    data.frame(
        newdata,
        intensity = predict(model, newdata=newdata, type="count"),
        zeros = predict(model, newdata=newdata, type="zero"))
}

pred <- pred_func(cp_hp, newdata)
```

Plotting with ggplot2

```
library(ggplot2)

# plot count and zeros separately
ggplot(pred, aes(x=length, y=intensity, color=area)) + geom_line(size=2) +
  facet_wrap(~year) + scale_y_log10(breaks=c(1,2,5,10,20))
```

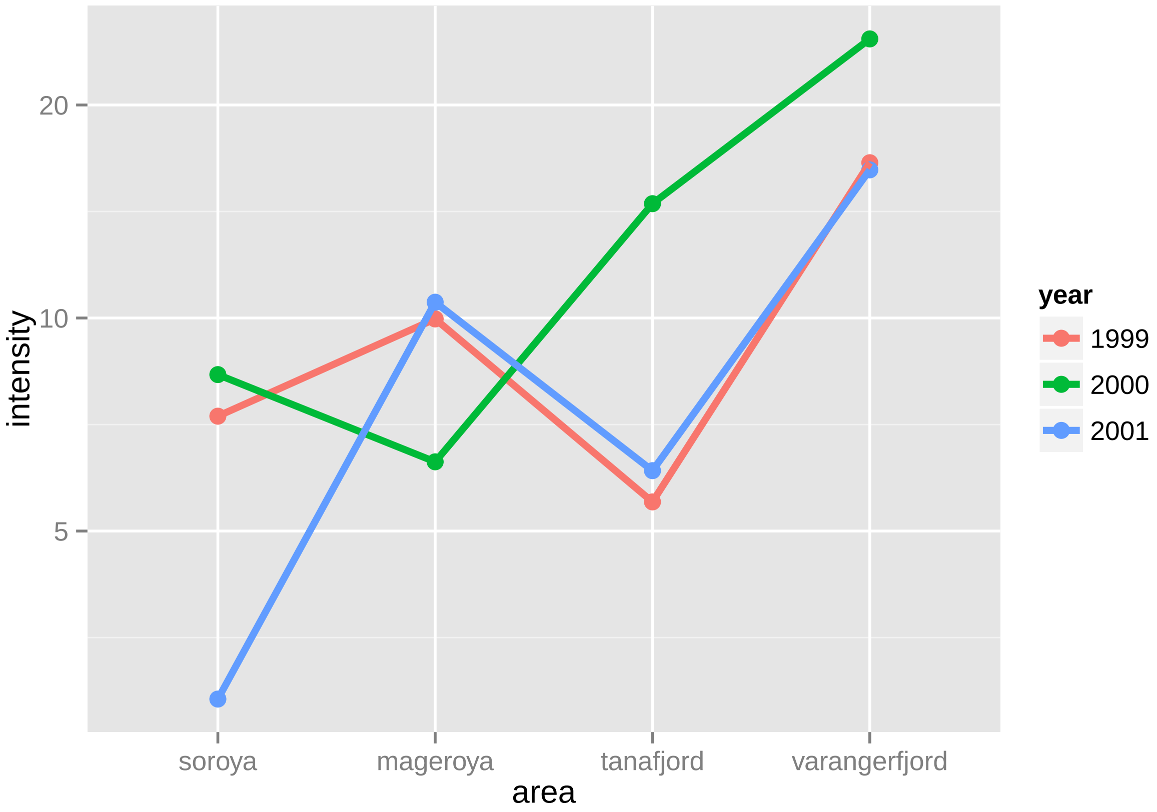


```
ggplot(pred, aes(x=length, y=zeros, color=area)) + geom_line(size=2) +
  facet_wrap(~year)
```

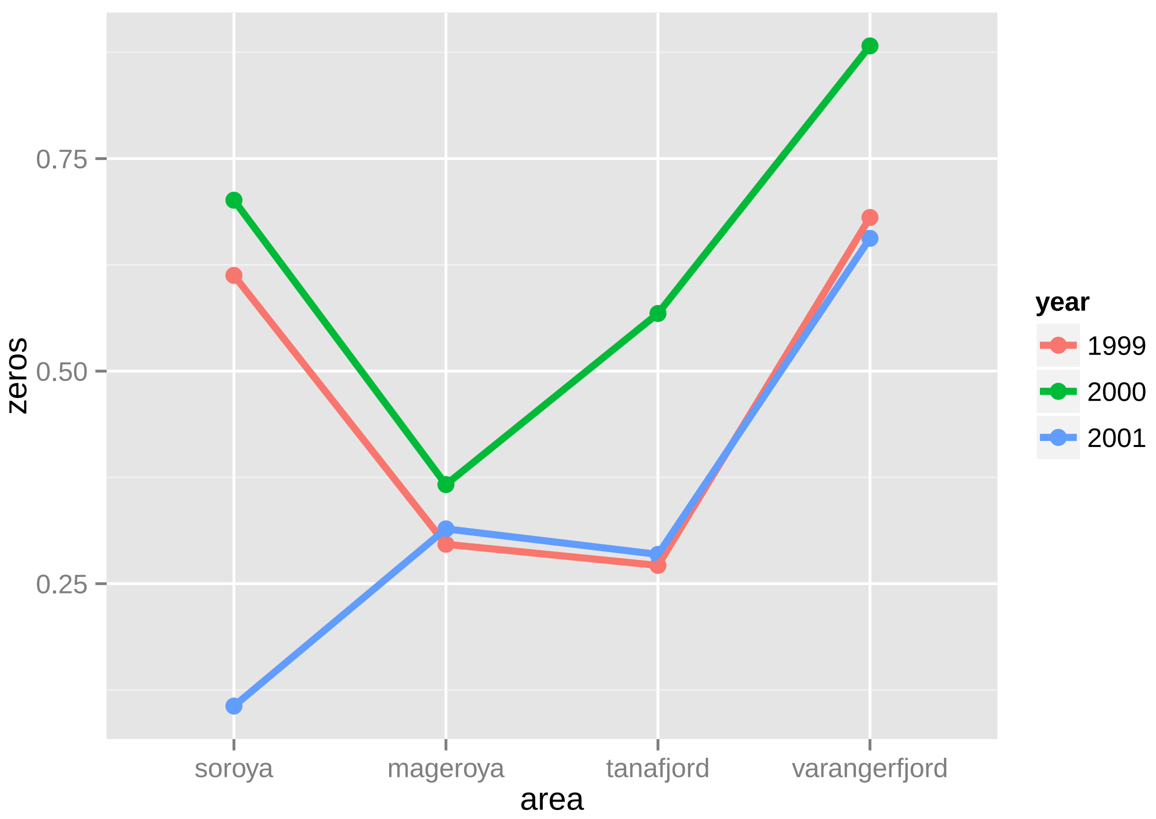


Effect plot: area * year effect, at mean length

```
newdata <- expand.grid( length=mean(CodParasites$length, na.rm=TRUE),
                        area=levels(CodParasites$area),
                        year=levels(CodParasites$year))
pred <- pred_func(cp_hp, newdata)

# perhaps should plot on log scale
ggplot(pred, aes(x=area, y=intensity, group=year, color=year)) +
  geom_point(size=3) + geom_line(size=1.25) +
  scale_y_log10(breaks=c(1,2,5,10,20))
```



```
ggplot(pred, aes(x=area, y=zeros, group=year, color=year)) +
  geom_point(size=3) +
  geom_line(size=1.25)
```


0.75
0.50
0.25
zeros
soroya
mageroya
tanafjord
varangerfjord
area
year
1999
2000
2001