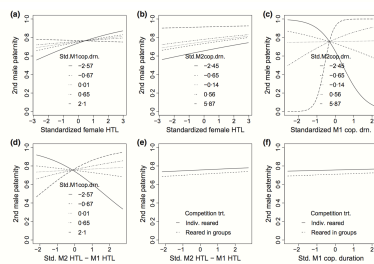


Advancing in R Module 7: predict



Outline

- Interpreting tables of coefficients
- What does R need to make predictions?
- Review use of predict()
- Strategies for visualizing interactions
- {broom} and {visreg}

Challenge

- Produce a drawing representing data using only coefficients from a linear model:

Call:

```
lm(formula = mass ~ pop * length, data = mydat)
```

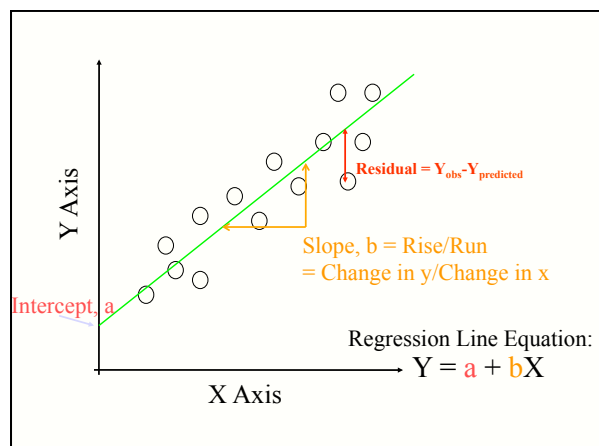
Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-0.2737	23.8061	-0.011	0.991
popMassif Central	17.5304	31.8089	0.551	0.587
popJura	6.7285	38.6962	0.174	0.863
length	6.1459	0.4220	14.564	2.06e-13 ***
popMassif Central:length	11.7764	0.5746	20.495	< 2e-16 ***
popJura:length	24.1139	0.6954	34.679	< 2e-16 ***

Residual standard error: 9.359 on 24 degrees of freedom

Multiple R-squared: 0.9998, Adjusted R-squared: 0.9997

F-statistic: 2.113e+04 on 5 and 24 DF, p-value: < 2.2e-16



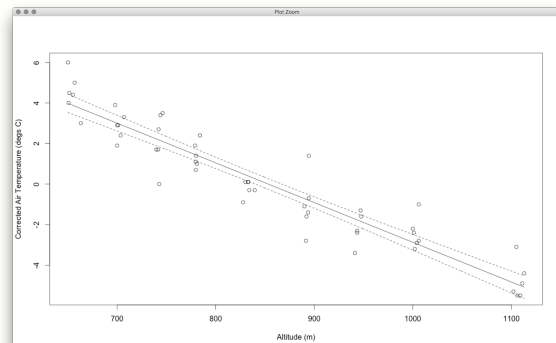
What do you need to generate a fitted line?

- An equation
 - the model coefficients
- Some new data
 - incl. a value for every predictor in each row

```

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 16.5723075  0.8498285   19.50  <2e-16 ***
ELEVATION   -0.0193620  0.0009853  -19.65  <2e-16 ***

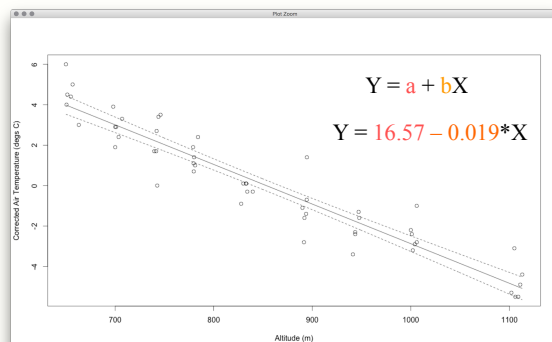
```



```

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 16.5723075  0.8498285   19.50  <2e-16 ***
ELEVATION   -0.0193620  0.0009853  -19.65  <2e-16 ***

```

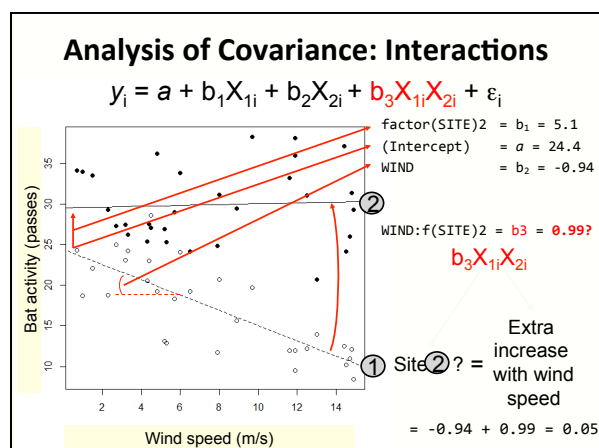
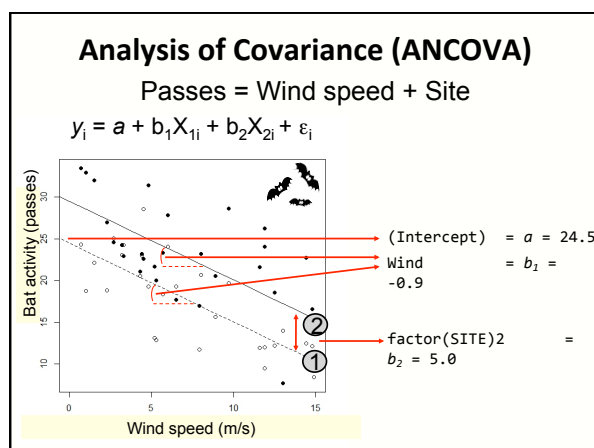
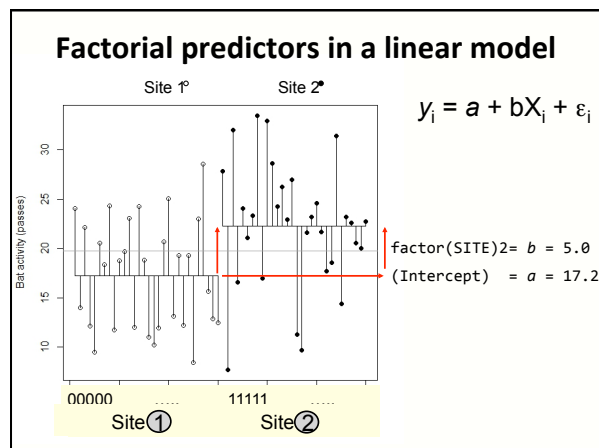
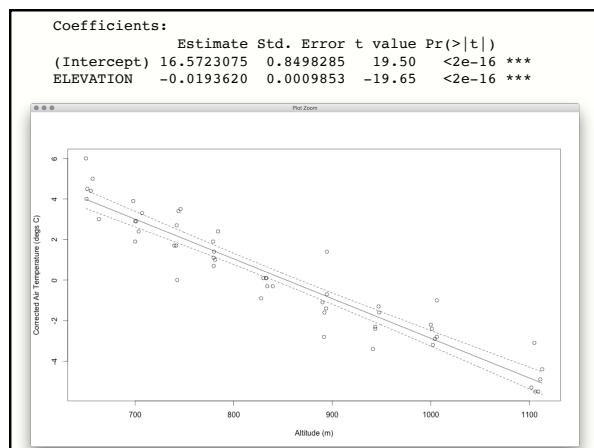


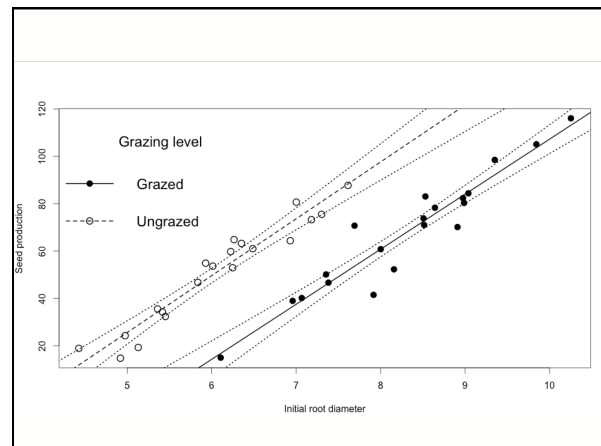
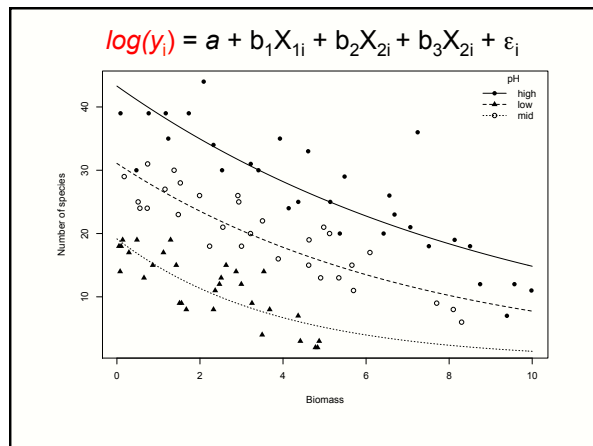
```
NEWXVARS.AIR<-seq(650,1113,1)
```

```
# generate predictions of model
NEWYVARS.AIR<-predict(AIR.MOD,
newdata=list(ELEVATION=
NEWXVARS.AIR), int="c")
```

```
# plot data
plot(CG$AIR.TEMP.CORR~CG$ELEVATION)
```

```
# plot fit
matlines(NEWXVARS.AIR,NEWYVARS.AIR,
lty=c(1,2,2),col="black")
```

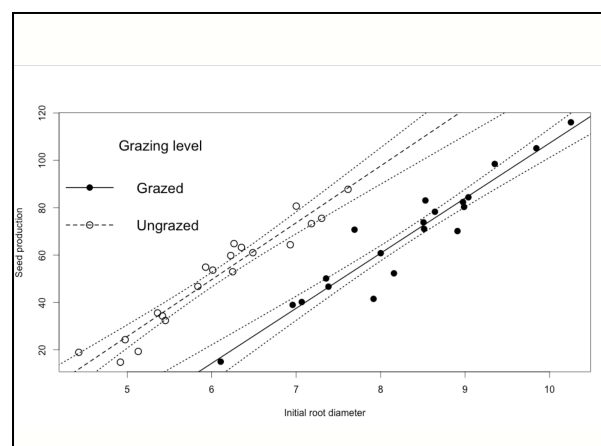




```
plot(FRUIT~ROOT,data=COMP,type="n",ylab="Seed
production",xlab="Initial root diameter",lwd=2)
points(SR[[1]],SF[[1]],pch=16,lwd=2,cex=1.5)
points(SR[[2]],SF[[2]],cex=1.5,lwd=2)

NEWGRAZED<-expand.grid(GRAZING="Grazed",
ROOT=seq(4,11,0.1))
NEWUNGRAZED<-expand.grid(GRAZING="Ungrazed",
ROOT=seq(4,11,0.1))
YVGRAZED<-predict(MOD.G1,list(GRAZING= NEWGRAZED
$GRAZING, ROOT=NEWGRAZED$ROOT),int="c")
YVUNGRAZED<-predict(MOD.G1,list(GRAZING=NEWUNGRAZED
$GRAZING,ROOT=NEWUNGRAZED$ROOT),int="c")

matlines(NEWGRAZED
$ROOT,YVGRAZED,lty=c(1,3,3),col="black",lwd=2)
matlines(NEWUNGRAZED
$ROOT,YVUNGRAZED,lty=c(2,3,3),col="black",lwd=2)
```



Q: How can we plot interactions between continuous variables?

A: Need more dimensions

- Colour or shape?
- Small multiples?
- 3D (e.g., surface or contours plots)
- Creative use of quantiles

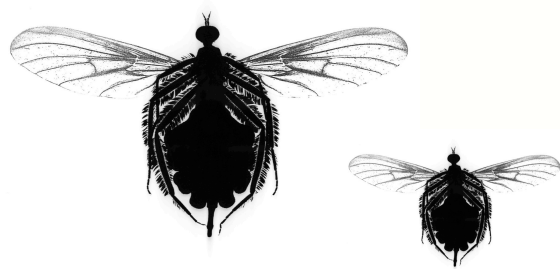
Photos: Gil Wizen



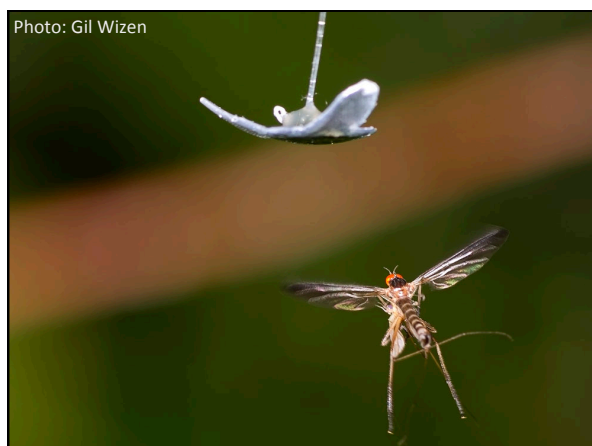
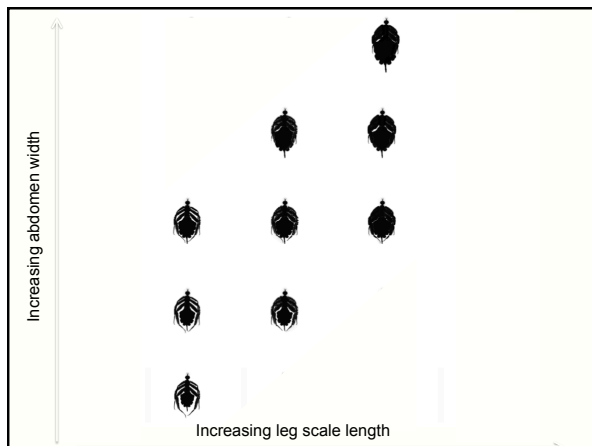
Photo: Gil Wizen



Manipulation of female size



Funk and Tallamy 2000



Call:
`glm(formula = ATTR ~ LEG * ABDOMEN, family = quasipoisson, data = DF)`

Coefficients:

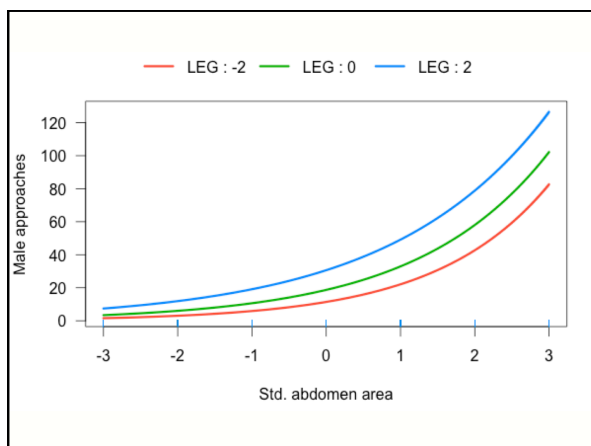
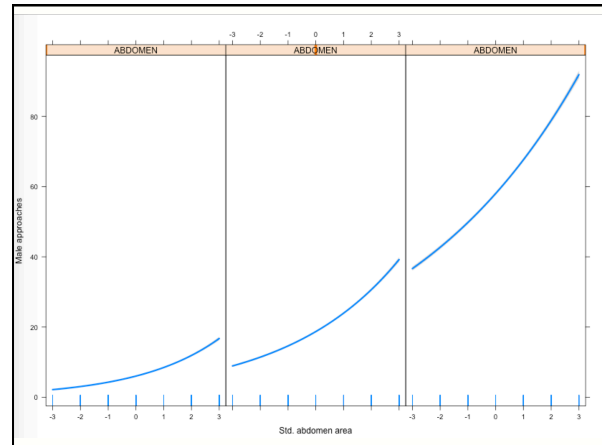
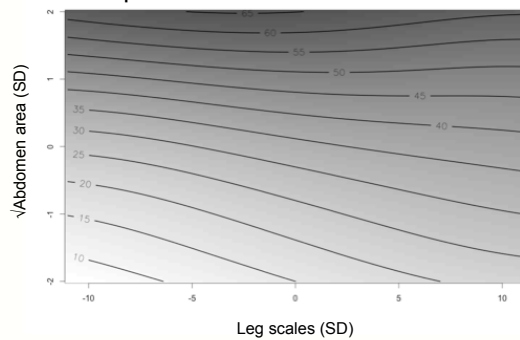
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.929183	0.006262	467.76	<2e-16 ***
LEG	0.246664	0.002964	83.22	<2e-16 ***
ABDOMEN	0.565923	0.002664	212.46	<2e-16 ***
LEG:ABDOMEN	-0.046726	0.001268	-36.86	<2e-16 ***

(Dispersion parameter for quasipoisson family taken to be 0.1501771)

Null deviance: 11550.880 on 342 df
 Residual deviance: 50.771 on 339 df
 AIC: NA

Number of Fisher Scoring iterations: 3

Both abdomen area and leg scale length improve attractiveness to males



Things you wish we told you earlier

- Package {broom}
 - Converts model outputs to tidy data frames
 - Useful for manipulations and exporting
- Package {visreg}
 - Visualizes fitted lines from model objects
 - Can condition by covariate values
 - Can work in link or response space

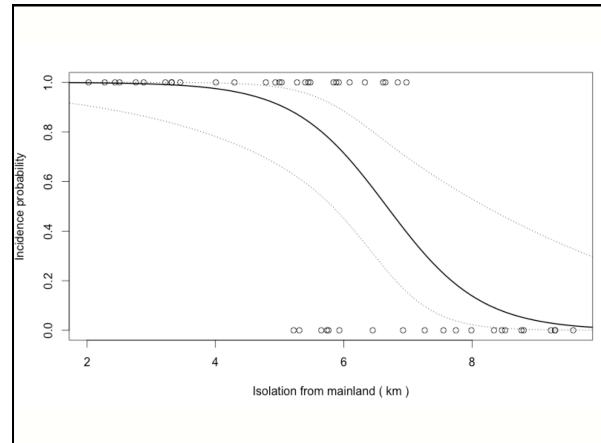
```
> summary(tanfit)

Call:
lm(formula = GROWTH ~ TANNIN, data = TANDAT)
Residuals:
    Min       1Q   Median       3Q      Max
-2.4556 -0.8889 -0.2389  0.9778  2.8944

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  11.7556    1.0408   11.295 9.54e-06 ***
TANNIN       -1.2167    0.2186   -5.565 0.000846 ***
---
Sig. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.693 on 7 degrees of freedom
Multiple R-squared:  0.8157,    Adjusted R-squared:  0.7893
F-statistic: 30.97 on 1 and 7 DF,  p-value: 0.0008461

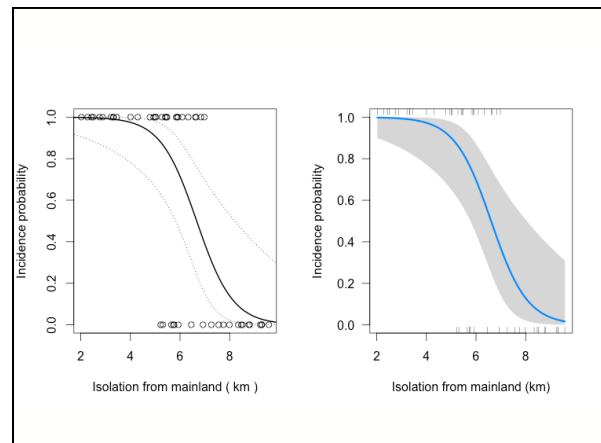
> ## Get a 'tidy' version of the model coefficients
> tidy(tanfit)
      term estimate std.error statistic    p.value
1 (Intercept) 11.75556  1.0407991 11.294740 9.537315e-06
2 TANNIN      -1.21667  0.2186115 -5.565427 8.460738e-04
```



```
XVISOL<-seq(0,10,length=100)
MNAREA<-rep(mean(ISLAND$AREA),100)
YVISOL<-predict(MOD,
2,list(ISOLATION=XVISOL,AREA=MNAREA),type="link",
se=TRUE)
plot(ISLAND$ISOLATION,ISLAND$INCIDENCE,
xlab="Isolation from mainland ( km )",
ylab="Incidence probability")
lines(XVISOL,ilogit(YVISOL$fit),lwd=2)
lines(XVISOL,
ilogit(YVISOL$fit+1.96*YVISOL$se.fit),lty=3)
lines(XVISOL,
ilogit(YVISOL$fit-1.96*YVISOL$se.fit),lty=3)

# OR

visreg(MOD.2, xvar = "AREA", scale="response",
rug=2, ylab="Incidence probability", xlab="Island
area (km^2)")
```



Suggested reading:

Homepage for {broom}
<https://github.com/dgrtwo/broom>

Breheny & Burchett 2013:
documentation for {visreg}
<http://myweb.uiowa.edu/pbreheny/publications/visreg.pdf>
(also in publications folder of course materials)