Measuring natural selection

Øystein H. Opedal

9 May 2022

Introduction

Natural selection may occur whenever individuals in a population differ in fitness. If these individuals differ also in phenotypic traits, so that certain trait values are associated with variation in fitness, we say that phenotypic selection acts on those traits. If those traits are also heritable (i.e. they exhibit some additive genetic variation), natural selection may also lead to evolution (i.e. response to selection).

The simple Breeder's equation illustrates the two-step process by which natural selection may lead to evolution:

$R = h^2 S$,

where $h^2 = V_A/V_P$ is the narrow-sense heritability, and S is the selection differential $S = cov(w, z)$, where w is relative fitness defined as $w = W/\bar{W}$.

In some cases, such as in artificial-selection studies, we can directly compute the selection differential as the difference between the trait mean before selection and the trait mean after selection. To measure selection in natural populations, however, this is often not practical.

Instead, to study selection on phenotypic traits in natural populations, we can use regression techniques to relate variation in the fitness of individuals to variation in their phenotypic traits. To see why this makes sense, we can write the Breeder's equation as

$R = \frac{V_A}{V_P} cov(w, z) = V_A \frac{cov(w,z)}{V_P}$.

Recall that $\beta = cov(w, z)/V_P$ is the regression slope of w on z. This formulation shows that we can estimate the strength of selection in natural populations by regressing relative fitness on trait values, an important result first discussed by Pearson (1903), and extended and operationalized in 1983 in a landmark paper by Russ Lande and Stevan J. Arnold.

The Lande-Arnold approach to measuring selection

Lande & Arnold (1983) operationalized the use of multiple regression to measure phenotypic selection in natural populations. Linear selection gradients, conventionally denoted as β, describe the local slope of the fitness surface describing the relationship between traits and fitness. We can estimate β as the partial regression coefficients of relative fitness (w) on a trait vector z. We can write this linear model as

$y_i = \alpha + \sum_j x_i \beta_j + \epsilon_i$

where y_i is the relative fitness of individual i, α is an intercept, and the β's are partial regression slopes.

To make selection gradients comparable across traits and species, they are normally standardized either by the phenotypic standard deviation $\beta_\sigma = \beta\, \sigma(x)$, or by the trait mean $\beta_\mu = \beta\, \mu(x)$.

The variance-standardized selection gradient is also referred to as the selection intensity $i = \frac{S}{\sigma(x)}$, and measures the proportional change in fitness per standard deviation change in the trait.

The mean-standardized selection gradient measure the proportional change in fitness per proportional change in the trait, and is technically an elasticity. Note that if the trait z is relative fitness, then the mean-standardized selection gradient is 1 (because we regress relative fitness on itself). This provides a useful benchmark for judging the strength of selection.

Exercise 1: Selection on *Ipomopsis aggragata* floral traits

To demonstrate the Lande-Arnold regression approach, we will analyse data collected by Campbell & Powers (2015) on floral traits and seed set of the plant *Ipomopsis aggregata*.

Read datafile

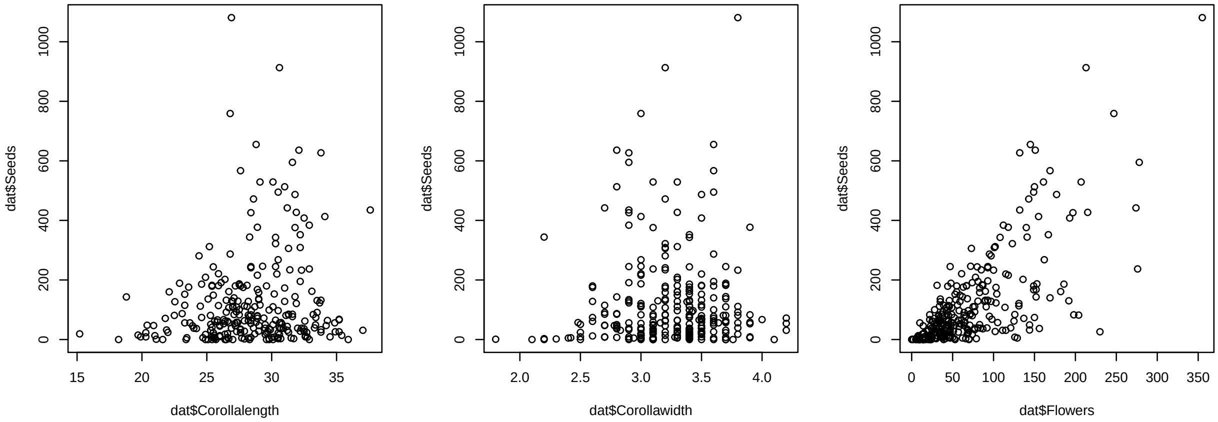
```
dat = read.table("campbellpowers.txt", header=T)
names(dat)
```

```
## [1] "Year"           "Seeds"          "Corollalength" "Corollawidth"
## [5] "Flowers"
```

Explore data

Before enbarking on any statistical analysis, it is advisable to explore the data graphically.

```
par(mfrow=c(1,3))
plot(dat$Corollalength, dat$Seeds)
plot(dat$Corollawidth, dat$Seeds)
plot(dat$Flowers, dat$Seeds)
```



Estimate variance-standardized univariate selection gradients

Univariate selection gradients measure net selection on the focal trait, and can be estimated as a simple regression of relative fitness on the trait. To obtain variance-standardized selection gradients, we scale the trait to zero mean and unit variance using the `scale` function in R.

```
# Define relative fitness
dat$relfit = dat$Seeds/mean(dat$Seeds, na.rm=T)

summary(lm(relfit ~ scale(Flowers), na=na.exclude, data=dat))
```

```
##
## Call:
## lm(formula = relfit ~ scale(Flowers), data = dat, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4156 -0.3683 -0.0174  0.3288  4.3409
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1.00000    0.05513   18.14   <2e-16 ***
## scale(Flowers)  0.95587    0.05523   17.31   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.9209 on 277 degrees of freedom
## Multiple R-squared:  0.5195, Adjusted R-squared:  0.5178
## F-statistic: 299.5 on 1 and 277 DF,  p-value: < 2.2e-16
```

The univariate directonal selection gradient on flower number is 0.96, which means that fitness nearly doubles (or changes by approximately 96%) per standard deviation increase in flower number.

Similarly, we can estimate the univariate selection gradients for corolla length and width:

```
summary(lm(relfit ~ scale(Corollalength), na=na.exclude, data=dat))$coef
```

```
##                       Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)          1.0401304 0.08193000 12.695355 4.198571e-29
## scale(Corollalength) 0.2193198 0.08208561  2.671843 8.016335e-03
```

```
summary(lm(relfit ~ scale(Corollawidth), na=na.exclude, data=dat))$coef
```

```
##                       Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)         1.04013040 0.08303049 12.5270902 1.585260e-28
## scale(Corollawidth) 0.01891072 0.08318819  0.2273245 8.203487e-01
```

The opportunity for selection

A requirement for natural selection is that there is variation in fitness among individuals. We can measure the *opportunity for selection* as the variance in relative fitness, $I = var(\frac{W}{\bar{W}})$. Conveniently, the maximum value of the variance-standardized selection gradient is the square root of I, $i = \sqrt{I}$.

```
I = var(dat$relfit)
signif(sqrt(I), 3)
```

```
## [1] 1.33
```

This suggests that selection on flower number is strong, as the variance-standardized selection gradient is $0.96/\sqrt(I) = 72\%$ of its maximum value.

Estimate mean-standardized univariate selection gradients

As noted above, variance-standardization is not the only way to make selection gradients comparable across traits and studies. Let us estimate the mean-standardized selection gradients. We can do this either by dividing the trait values by their mean prior to fitting the model, or by multiplying the raw selection gradient by the trait mean as we do below.

```
summary(lm(relfit ~ scale(Flowers, scale=F), na=na.exclude, data=dat))$coef
```

```
##                             Estimate   Std. Error  t value     Pr(>|t|)
## (Intercept)               1.00000000 0.0551319028 18.13832 5.281723e-49
## scale(Flowers, scale = F) 0.01674632 0.0009676138 17.30682 5.383857e-46
```

```
0.0167463*mean(dat$Flowers, na.rm=T)
```

```
## [1] 1.217018
```

```
summary(lm(relfit ~ scale(Corollalength, scale=F), na=na.exclude, data=dat))$coef
```

```
##                                   Estimate Std. Error   t value     Pr(>|t|)
## (Intercept)                     1.04013040 0.08193000 12.695355 4.198571e-29
## scale(Corollalength, scale = F) 0.06004131 0.02247187  2.671843 8.016335e-03
```

```
0.06004*mean(dat$Corollalength, na.rm=T)
```

```
## [1] 1.703555
```

```
summary(lm(relfit ~ scale(Corollawidth, scale=F), na=na.exclude, data=dat))$coef
```

```
##                                  Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)                    1.04013040 0.08303049 12.5270902 1.585260e-28
## scale(Corollawidth, scale = F) 0.04809602 0.21157424  0.2273245 8.203487e-01
```

```
0.04810*mean(dat$Corollawidth, na.rm=T)
```

```
## [1] 0.1560553
```

Selection on flower number and corolla length is stronger than selection on fitness as a trait($\beta_\mu = 1$), which is very strong.

Estimate variance-standardized multivariate selection gradients

Univariate selection gradients measure the net strength of selection on a trait, including direct selection on the trait, and indirect selection due to selection on trait that are phenotypically correlated with the focal trait. In a multiple-regression model, the coefficient for each independent variable is estimated while holding the other independent variables constant at their mean. The key advantage of the multiple-regression approach of Lande and Arnold is therefore that selection gradients estimated as partial regression coefficients accounts for indirect selection.

```
m = lm(relfit~scale(Flowers) + scale(Corollalength) + scale(Corollawidth), na=na.exclude, data=dat)
summary(m)$coef
```

```
##                        Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)          1.00121187 0.05812729 17.2244723 7.027638e-45
## scale(Flowers)       0.96440781 0.05964956 16.1678956 3.602006e-41
## scale(Corollalength) 0.01459245 0.05980547  0.2439986 8.074243e-01
## scale(Corollawidth)  0.08357638 0.05868517  1.4241483 1.556026e-01
```

We notice that selection on flower number is still strong, while the selection gradients on corolla length and width have changed. To see how the multiple-regression approach separates direct from indirect selection, we can compute the univariate selection gradient (referred to here as s for simplicity) from the coefficients above. For corolla length (trait 2), for example,this is given by

$s = \beta_1 r_{12} + \beta_2 + \beta_3 r_{23}$, where r_{12} is the phenotypic correlation between trait 1 and trait 2.

```
beta_flowers = summary(m)$coef[2,1]
beta_length = summary(m)$coef[3,1]
beta_width = summary(m)$coef[4,1]

beta_length +
  beta_flowers*cor(dat$Corollalength, dat$Flowers, use="pairwise") +
  beta_width*cor(dat$Corollalength, dat$Corollawidth, use="pairwise")
```

```
## [1] 0.2192202
```

This demonstrates that the net selection gradient on a trait includes the indirect selection due to phenotypic correlations with another trait under selection (e.g. flower number in the case of corolla length). For all the traits, we can do this with matrix algebra, computing

$s^T = \beta \mathbf{P}$, where $\mathbf{P}$ is the phenotypic correlation matrix. Matrix multiplication in R is done using the `%*%` operator.

```
beta = summary(m)$coef[2:4,1]
signif(t(beta), 2)
```

```
##      scale(Flowers) scale(Corollalength) scale(Corollawidth)
## [1,]           0.96                0.015               0.084
```

```
cormat = cor(dat[,3:5], use="pairwise")[c(3,1,2),c(3,1,2)]
signif(cormat, 2)
```

```
##                Flowers Corollalength Corollawidth
## Flowers          1.000         0.200       -0.068
## Corollalength    0.200         1.000        0.094
## Corollawidth    -0.068         0.094        1.000
```

```
beta%*%cormat
```

```
##        Flowers Corollalength Corollawidth
## [1,] 0.9616648     0.2192202   0.01894413
```

Exercise 2: Selection on *Dalechampia scandens* blossom traits

Read datafile

```
dal = read.csv("Dalechampia.csv")
head(dal)
```

```
##   patch    UBA    GA  GAD  GSD  ASD pollenfem pollenmale total_seeds seeds_pred
## 1     0 526.21 41.27 5.26 4.41 6.23        22          0          NA         NA
## 2     0 690.84 47.82 7.50 6.23 6.72        60          0           0          0
## 3     0 404.43 42.34 5.55 4.80 4.37       120         30           0          0
## 4     0 456.99 33.11 3.72 5.00 5.47        28          2          NA         NA
## 5     0 342.99 33.29 4.64 5.10 5.42        30          0          NA         NA
## 6     0 464.24 35.96 4.62 4.15 6.12        22          0           0          0
```

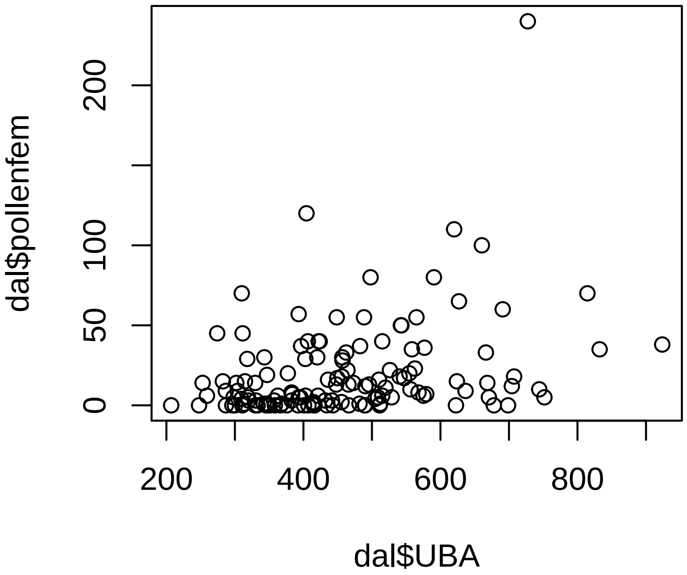
Explore data

```
dal$relfit = dal$total_seeds/mean(dal$total_seeds, na.rm=T)
dal$pollentot = dal$pollenfem + dal$pollenmale

signif(cor(dal[,c(2,3,5,6)], use="pairwise"), 2)
```

```
##      UBA   GA   GSD   ASD
## UBA 1.00 0.69 0.440 0.220
## GA  0.69 1.00 0.270 0.290
## GSD 0.44 0.27 1.000 0.043
## ASD 0.22 0.29 0.043 1.000
```

```
plot(dal$UBA, dal$pollenfem)
```



Univariate selection gradients for Upper Bract Area

```
m = lm(relfit ~ scale(UBA, scale=F), na=na.exclude, data=dal)
summary(m)
```

```
##
## Call:
## lm(formula = relfit ~ scale(UBA, scale = F), data = dal, na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0474 -0.9007 -0.5954  0.9562  3.0734
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.994678   0.129800   7.663 7.66e-12 ***
## scale(UBA, scale = F) 0.002831   0.000931   3.041  0.00294 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.374 on 110 degrees of freedom
##   (20 observations deleted due to missingness)
## Multiple R-squared:  0.07757,    Adjusted R-squared:  0.06918
## F-statistic:  9.25 on 1 and 110 DF,  p-value: 0.002945
```

```
summary(m)$coef[2,1]*sd(dal$UBA, na.rm=T)
```

```
## [1] 0.3849789
```

```
summary(m)$coef[2,1]*mean(dal$UBA, na.rm=T)
```

```
## [1] 1.303954
```

Multivariate selection gradients

```
m = lm(relfit ~ scale(UBA) + scale(GA) + scale(GSD) + scale(ASD), na=na.exclude, data=dal)
summary(m)$coef
```

```
##                Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)  1.05818407  0.1332267  7.9427351 3.360314e-12
## scale(UBA)   0.76167461  0.1872266  4.0681966 9.590341e-05
## scale(GA)   -0.50923897  0.1828435 -2.7851082 6.423837e-03
## scale(GSD)  -0.01957882  0.1579361 -0.1239667 9.015955e-01
## scale(ASD)  -0.20552292  0.1358425 -1.5129499 1.335107e-01
```

Selection through pollen arrival

To obtain the most accurate measure of selection, we would like our fitness measure to be as 'close' as possible to fitness. In some cases, such as when working with annual plants, the number of seeds produced may be a good proxy of fitness, but in most cases we are stuck with some fitness component such as the seed set in a given year in a perennial plant.

In the conetext of pollinator-mediated selection on floral traits, we are interested in how the interactions between flowers and pollinators affect fitness, thus learning about the functional components of selection, that we expect over time to lead to selection and adaptation.

The number of seeds produced may depend on many factors, including the number of pollen grains arriving onto stigmas, and the environment of the mother plant. Selection gradients estimated using seed set as a fitness component may therefore not represent purely pollinator-mediated selection. In these cases, much can be learned by considering lower-level fitness components such as the number of pollen grains deposited onto stigmas by pollinators.

Dalechampia blossoms are functionally protogynous, which means that during the development of the blossom there is a female phase of several days before the first male flower opens. All pollen deposited during this phase is necessarily deposited by pollinators.

Because the number of pollen grains is a count variable, we fit a generalized linear model with Poisson errors. Conveniently, because of the log link function, regression coefficients from a Poisson GLM are roughly identical to selection gradients estimated from a linear model with relative fitness as the response. This is because the variance of a natural log-transformed variable is almost identical to that of a mean-scaled variable.

```
m = glm(pollenfem ~ scale(UBA, scale=F), family="poisson", na=na.exclude, data=dal)
summary(m)$coef
```

```
##                          Estimate   Std. Error   z value      Pr(>|z|)
## (Intercept)           2.906816742 0.0211406974 137.49862  0.000000e+00
## scale(UBA, scale = F) 0.003135779 0.0001229824  25.49779 2.085563e-143
```

```
summary(m)$coef[2,1]*sd(dal$UBA, na.rm=T)
```

```
## [1] 0.4263522
```

```
summary(m)$coef[2,1]*mean(dal$UBA, na.rm=T)
```

```
## [1] 1.444089
```