# ENV 710

overview

Duke | NICHOLAS SCHOOL OF THE ENVIRONMENT

**Protocol for conducting and presenting results of regression-type analyses**

1. **State appropriate questions**
2. **Visualize the experimental design**
3. **Conduct data exploration**
4. **Identify the dependency structure in the data**
5. **Present the statistical model**
6. **Fit the model**
7. **Validate the model**
8. **Interpret and present the numerical output of the model**
9. **Create a visual representation of the model**
10. **Simulate from the model**

know what question you are asking or problem you are trying to solve

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE

# Mockingbirds: is the friend of my enemy, my enemy too?

We tested the following hypotheses:

1. Flush distance would increase from days 1-4 as the female bird increased recognition of the intruder.
2. The female bird would link the associate to the intruder, and recognize the associate as a threat; therefore, the female bird would flush sooner upon the associate's approach than an unrecognized person (i.e., the intruder on trial 1).
3. The female bird would identify the bystander as non-threatening; therefore, she would flush later upon the bystander's approach (i.e., similar to the intruder on trial 1).
4. The female bird would not recognize the control as a threat; therefore she would flush later upon the control's approach (i.e., similar to the intruder on trial 1).

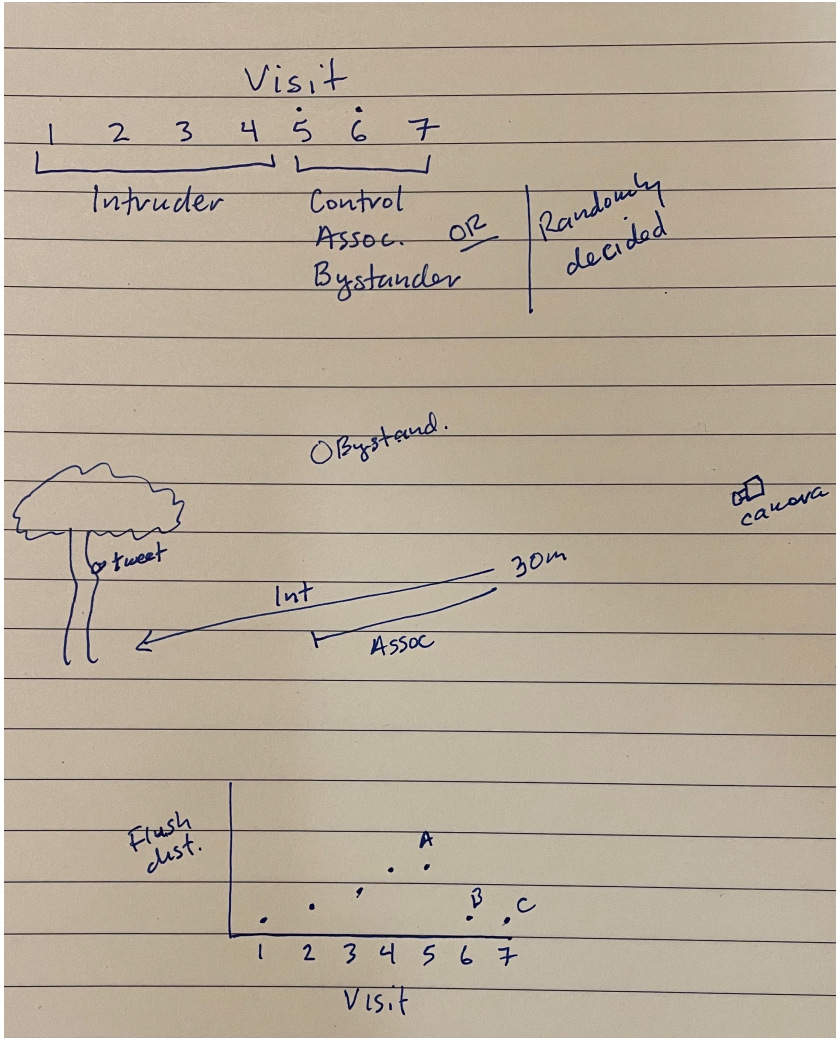
**Protocol for conducting and presenting results of regression-type analyses**

1. **State appropriate questions**
2. **Visualize the experimental design**
3. **Conduct data exploration**
4. **Identify the dependency structure in the data**
5. **Present the statistical model**
6. **Fit the model**
7. **Validate the model**
8. **Interpret and present the numerical output of the model**
9. **Create a visual representation of the model**
10. **Simulate from the model**

must fully understand the experimental design to build the correct model and answer your stated question

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE




must fully understand the experimental design to build the correct model and answer your stated question

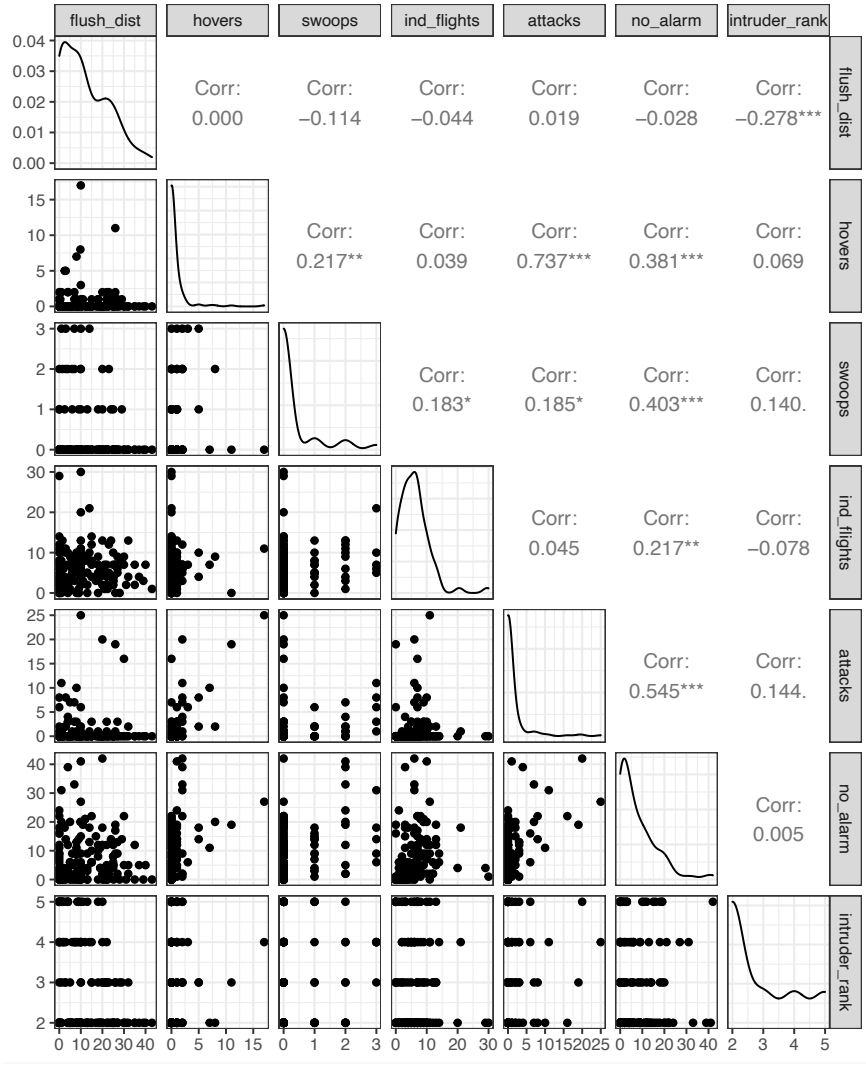
**Protocol for conducting and presenting results of regression-type analyses**

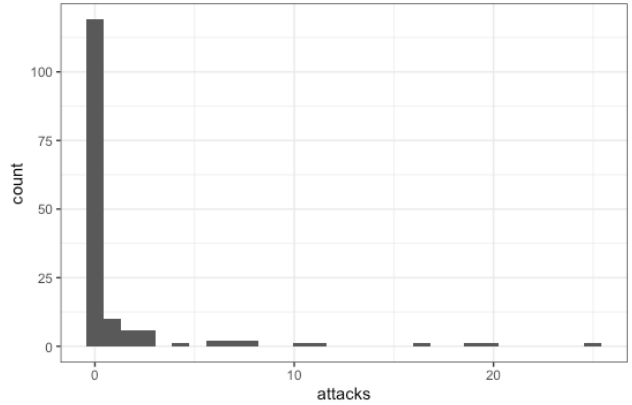
1. **State appropriate questions**
2. **Visualize the experimental design**
3. **Conduct data exploration**
4. **Identify the dependency structure in the data**
5. **Present the statistical model**
6. **Fit the model**
7. **Validate the model**
8. **Interpret and present the numerical output of the model**
9. **Create a visual representation of the model**
10. **Simulate from the model**

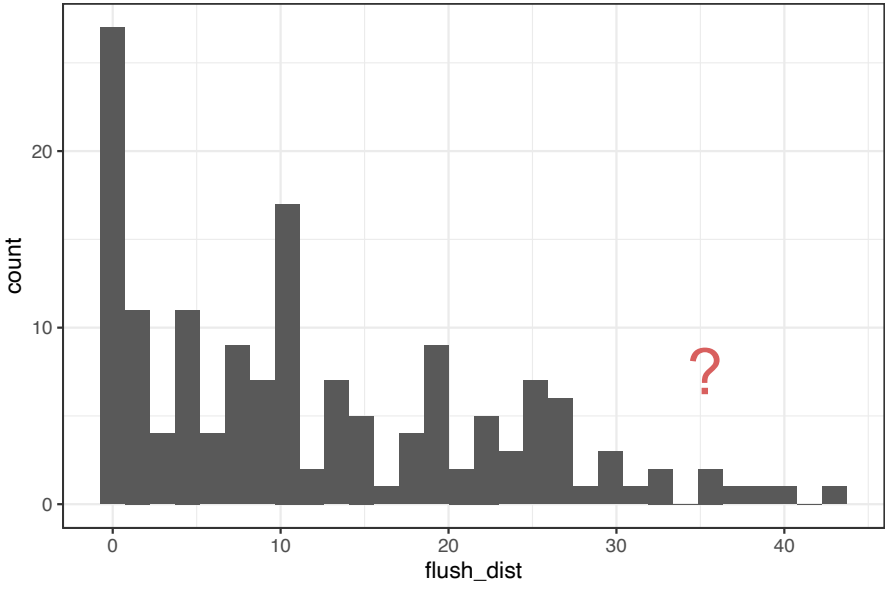
`ggpairs()` ... understand your data, graph it in different ways, look for potential extreme data points, non-linearities, obvious differences among groups, etc.

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE


flush_dist
hovers
swoops
ind_flights
attacks
no_alarm
intruder_rank
Corr: 0.000
Corr: −0.114
Corr: −0.044
Corr: 0.019
Corr: −0.028
Corr: −0.278***
Corr: 0.217**
Corr: 0.039
Corr: 0.737***
Corr: 0.381***
Corr: 0.069
Corr: 0.183*
Corr: 0.185*
Corr: 0.403***
Corr: 0.140.
Corr: 0.045
Corr: 0.217**
Corr: −0.078
Corr: 0.545***
Corr: 0.144.
Corr: 0.005


count
attacks


count
flush_dist
?

**Protocol for conducting and presenting results of regression-type analyses**

1. **State appropriate questions**
2. **Visualize the experimental design**
3. **Conduct data exploration**
4. **Identify the dependency structure in the data**
5. **Present the statistical model**
6. **Fit the model**
7. **Validate the model**
8. **Interpret and present the numerical output of the model**
9. **Create a visual representation of the model**
10. **Simulate from the model**

- this goes hand-in-hand with "visualize the experimental design"
- random effects should be used in most models

*same nest (and bird) is approached for 7 visits, correlation in bird reaction*

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE

**Protocol for conducting and presenting results of regression-type analyses**

1. **State appropriate questions**
2. **Visualize the experimental design**
3. **Conduct data exploration**
4. **Identify the dependency structure in the data**
5. **Present the statistical model**
6. **Fit the model**
7. **Validate the model**
8. **Interpret and present the numerical output of the model**
9. **Create a visual representation of the model**
10. **Simulate from the model**

yes, writing out the statistical model is good. most importantly, identify the type of data of your response variable and the model that usually works for it, i.e., normal, Poisson, binomial

$$\hat{y}_i = \alpha_{j[i]} + \beta_1 X_{iI2} + \beta_2 X_{iI3} + \beta_3 X_{iI4} + \beta_4 X_{iCntl} + \beta_5 X_{iByst} + \beta_6 X_{iAssc}$$
$$y_i \sim Norm(\hat{y}_i, \sigma^2)$$
$$\alpha_1 \sim Norm(0, \sigma_\alpha^2)$$

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE

**Protocol for conducting and presenting results of regression-type analyses**

1. State appropriate questions
2. Visualize the experimental design
3. Conduct data exploration
4. Identify the dependency structure in the data
5. Present the statistical model
6. Fit the model
7. Validate the model
8. Interpret and present the numerical output of the model
9. Create a visual representation of the model
10. Simulate from the model

```
#### model 1
fd1_lm <- lmer(flush_dist~factor(visit) + (1|location),
               data = mdat, REML = FALSE)
r.squaredGLMM(fd1_lm)

nsim <- 2000
bsim <- sim(fd1_lm, n.sim = nsim)
bvals <- apply(bsim@fixef, 2, quantile, prob = c(0.025, 0.5,
0.975))
```

play in R!

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE

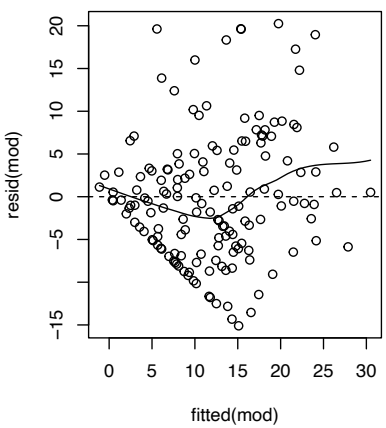
**Protocol for conducting and presenting results of regression-type analyses**

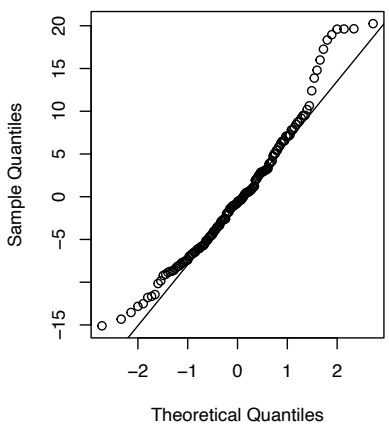
1. State appropriate questions
2. Visualize the experimental design
3. Conduct data exploration
4. Identify the dependency structure in the data
5. Present the statistical model
6. Fit the model
7. Validate the model
8. Interpret and present the numerical output of the model
9. Create a visual representation of the model
10. Simulate from the model

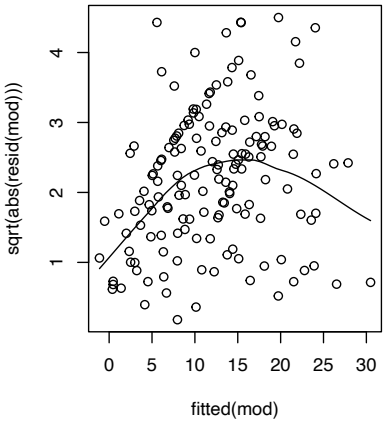
diagnostic plots, goodness of fit, check assumptions

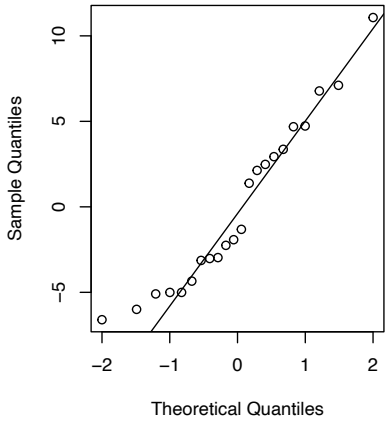
**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE


resid(mod)
fitted(mod)


normal QQ−plot, residuals
Sample Quantiles
Theoretical Quantiles


sqrt(abs(resid(mod)))
fitted(mod)


normal QQ−plot, random effects
Sample Quantiles
Theoretical Quantiles

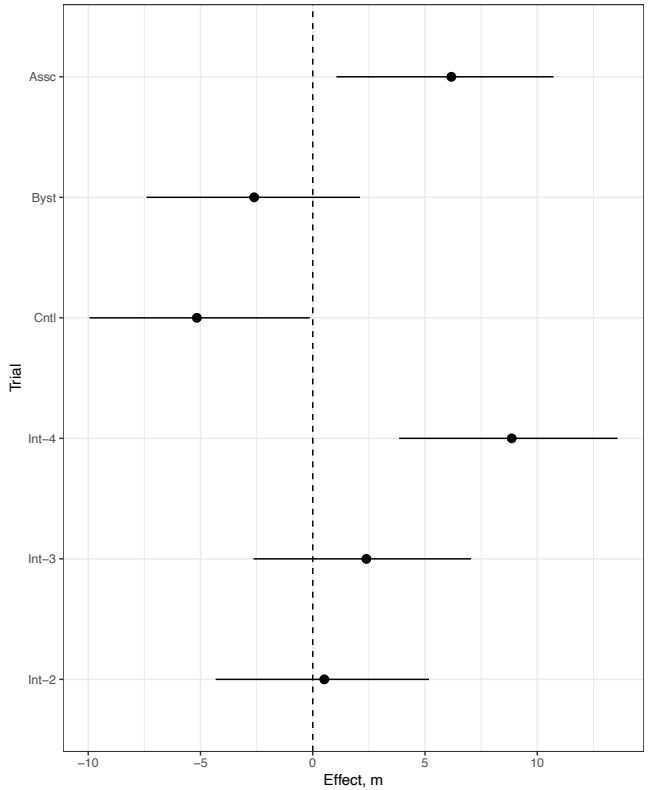
**Protocol for conducting and presenting results of regression-type analyses**

1. **State appropriate questions**
2. **Visualize the experimental design**
3. **Conduct data exploration**
4. **Identify the dependency structure in the data**
5. **Present the statistical model**
6. **Fit the model**
7. **Validate the model**
8. **Interpret and present the numerical output of the model**
9. **Create a visual representation of the model**
10. **Simulate from the model**

write your results and interpret effects

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

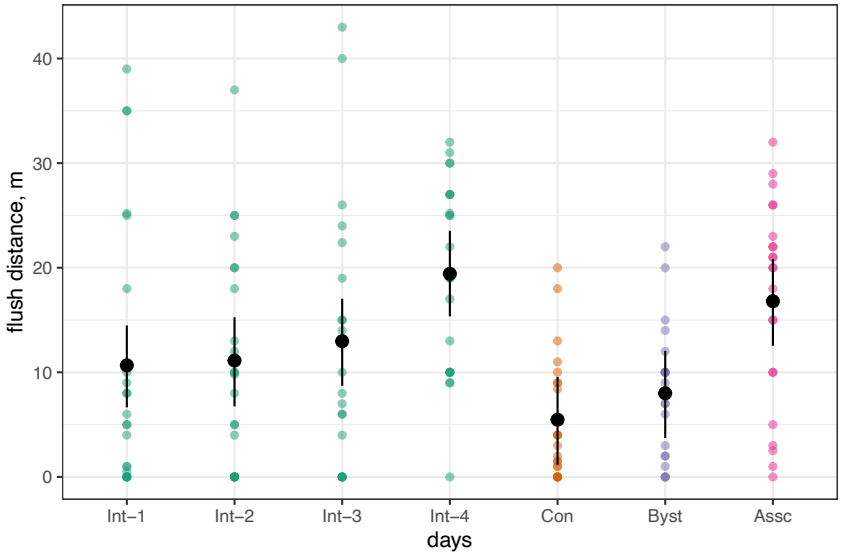
Zuur and Ieno 2016 MEE




- female birds flush when intruder 4 is on average 8.6 m farther from the nest than for intruder 1
- female birds flush when associate is on average 6 m farther from the nest than for intruder 1

## Protocol for conducting and presenting results of regression-type analyses

1. State appropriate questions
2. Visualize the experimental design
3. Conduct data exploration
4. Identify the dependency structure in the data
5. Present the statistical model
6. Fit the model
7. Validate the model
8. Interpret and present the numerical output of the model
9. Create a visual representation of the model
10. Simulate from the model




a pictures is worth a thousand words

**Fig. 1.** Protocol for statistical analysis of data and presenting results in a scientific paper.

Zuur and Ieno 2016 MEE

```
> r.squaredGLMM(fd1_lm)
           R2m       R2c
[1,] 0.1789538 0.4454147
```

- The overall model accounted for 44.5% of the variance in the data, with the main effects accounting for 17.9%.
- Bird nest (location) accounted for 32% of the variation in flush distance.
- The female bird flushed when the Associate was 6.1 m farther away compared to Intruder 1 ($t = 2.57$, df = 132, $p = 0.011$).

```
> summary(fd1_lm)
Formula: flush_dist ~ factor(visit) + (1 | location)
   Data: mdat

Random effects:
 Groups   Name        Variance Std.Dev.
 location (Intercept) 29.99    5.477
 Residual             62.42    7.901
Number of obs: 154, groups:  location, 22

Fixed effects:
                            Estimate Std. Error       df t value Pr(>|t|)
(Intercept)                  10.6682     2.0496  94.3656   5.205 1.13e-06 ***
factor(visit)Intruder Day 2   0.4591     2.3822 132.0000   0.193 0.847476
factor(visit)Intruder Day 3   2.3045     2.3822 132.0000   0.967 0.335110
factor(visit)Intruder Day 4   8.7500     2.3822 132.0000   3.673 0.000347 ***
factor(visit)Control         -5.1955     2.3822 132.0000  -2.181 0.030958 *
factor(visit)Bystander       -2.6682     2.3822 132.0000  -1.120 0.264724
factor(visit)Associate        6.1273     2.3822 132.0000   2.572 0.011214 *
```

# Linear Modeling in R

Cheat Sheet

| Model type | Probability distribution | Dependent variable | Variable types | Independent variable(s) | R |
|---|---|---|---|---|---|
| linear models | Normal | continuous | fixed effects (main effects, interactions, polynomials) | nominal, continuous | nominal IV<br>`lm(y ~ factor(x))`<br>continuous IV<br>`lm(y ~ x)`<br>interaction<br>`lm(y ~ x*z)` |
| | | | random effects | nominal | `lmer(y ~ x + (1\|τ))` |
| generalized linear models | Poisson | count | fixed effects (main effects, interactions, polynomials) | nominal, continuous | `glm(y ~ x, family=poisson)`<br>model rates with offset<br>`glm(y ~ x, offset=log(t), family=poisson)`<br>model overdispersion with negative binomial<br>`glm.nb(y ~ x)` |
| | binomial | binary | | | `glm(y ~ x, family=binomial)` |
| | | count (*y* successes, *n* trials) | | | `glm(cbind(y, n-y) ~ x, family=binomial)` |
| | | | random effects | nominal | `glmer(y ~ x + (1\|τ), family=family)` |

lmer(y~x1*x2*x3 + 1|b)

reduce to most parsimonious model
- remove NS parameters
- nested models: anova(m1, m2, test = 'F')
- non-nested models: AICc()

normality assumption is necessary to unbiasedly estimate standard errors, and hence confidence intervals and P-values. in large sample sizes (e.g., where the number of observations per variable is >15) violations of this normality assumption often do not noticeably impact results.

check diagnostic plots, plot(m1), ggpairs(m1)
- linearity b/n DV and IVs (residual plot)
- normality of residuals (qq plot)
- homoscedasticity (scale-location plot)
- outliers (leverage plot)
- check normality of random effects,
  qqnorm(ranef(m1)),
  ggplot(df, aes(sample = y)) + stat_qq() + stat_qq_line()

✓ add variables, interactions, polynomials

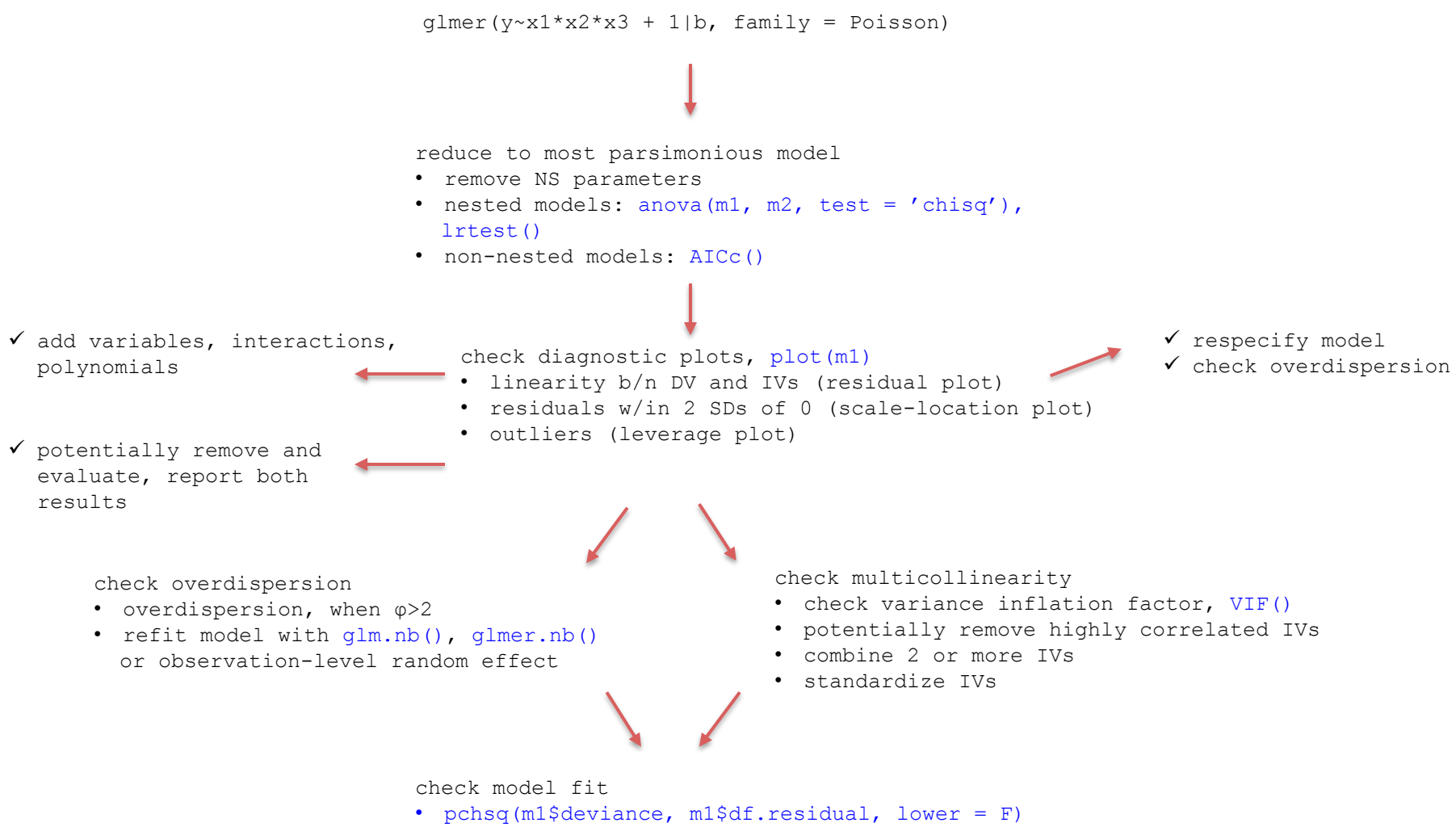
✓ potentially remove and evaluate, report both results

✓ transform DV
✓ respecify model

✓ transform DV
✓ use weighted least squares

check multicollinearity
- check variance inflation factor, VIF()
- potentially remove highly correlated IVs
- combine 2 or more IVs
- standardize IVs

```
wt <- 1 / lm(abs(mod$residuals) ~ mod$fitted.values)$fitted.values^2
wls <- lm(y ~ x, data = df, weights=wt)
```


glmer(y~x1*x2*x3 + 1|b, family = Poisson)
reduce to most parsimonious model
• remove NS parameters
• nested models: anova(m1, m2, test = 'chisq'), lrtest()
• non-nested models: AICc()
check diagnostic plots, plot(m1)
• linearity b/n DV and IVs (residual plot)
• residuals w/in 2 SDs of 0 (scale-location plot)
• outliers (leverage plot)
✓ add variables, interactions, polynomials
✓ potentially remove and evaluate, report both results
✓ respecify model
✓ check overdispersion
check overdispersion
• overdispersion, when φ>2
• refit model with glm.nb(), glmer.nb() or observation-level random effect
check multicollinearity
• check variance inflation factor, VIF()
• potentially remove highly correlated IVs
• combine 2 or more IVs
• standardize IVs
check model fit
• pchsq(m1$deviance, m1$df.residual, lower = F)

# Linear Modeling in R

Cheat Sheet

| Model type | Probability distribution | Dependent variable | Variable types | Independent variable(s) | R |
|---|---|---|---|---|---|
| linear models | Normal | continuous | fixed effects (main effects, interactions, polynomials) | nominal, continuous | nominal IV<br>`lm(y ~ factor(x))`<br>continuous IV<br>`lm(y ~ x)`<br>interaction<br>`lm(y ~ x*z)` |
| | | | random effects | nominal | `lmer(y ~ x + (1\|τ))` |
| generalized linear models | Poisson | count | fixed effects (main effects, interactions, polynomials) | nominal, continuous | `glm(y ~ x, family=poisson)`<br>model rates with offset<br>`glm(y ~ x, offset=log(t), family=poisson)`<br>model overdispersion with negative binomial<br>`glm.nb(y ~ x)` |
| | binomial | binary | | | `glm(y ~ x, family=binomial)` |
| | | count (*y* successes, *n* trials) | | | `glm(cbind(y, n-y) ~ x, family=binomial)` |
| | | | random effects | nominal | `glmer(y ~ x + (1\|τ), family=family)` |

# Parameter interpretation & model comparison

- parameter interpretation
- ✓ possible fixes to failure to meet assumptions

## LM with nominal IV's (ANOVA)

- extract ANOVA table to view sum-of-squares: `aov()`
- ✓ overall model test, *F*-test: `anova()`
- ✓ *post-hoc* tests to test for differences in factor levels: `tukeyHSD()`

## LM with nominal and continuous IV's

- partial regression plots: `avPlots()`
- ✓ compare nested models with partial F-test, e.g., `anova()`
- ✓ compare non-nested models with AIC or adjusted $R^2$, e.g., `AIC()`

## GLM with Poisson distribution (Poisson regression)

- uses log link, coefficients are outputted on log scale
- exponentiate coefficients to interpret them as ratios, e.g., `exp(coef())`
- ✓ compare nested models with likelihood ratio test: `lrtest(), anova(…, test="Chisq")`
- ✓ compare non-nested models with AIC, e.g., `AICc(), AICtab()`

## GLM binary logistic regression

- uses logit link, coefficients are outputted as log-odds
- exponentiate coefficients to interpret them as odds, e.g., `exp(coef(mod))`
- take inverse logit to predict as probability, e.g., `inv.logit()`
- ✓ same model comparison as Poisson regression

## GLM binomial count logistic regression

- same coefficient interpretation as binary logistic regression
- ✓ same model comparison as Poisson regression

Cheat Sheet created by John R. Poulsen

# G/LM assumptions and testing

- assumptions
- ✓ possible fixes to failure to meet assumptions

## DV is Independently & Identically Distributed

- each sample must have the same distribution and all samples must be mutually independent
- assess study design: are observations correlated?
- plot data in order they were collected/measured: do not want patterns

✓ include random effects to account for nested design, spatial or temporal autocorrelation

## DV & IV are linearly related

- evaluate graphically, e.g. `ggpairs()`
- residuals must be ~ normally distributed with a mean of 0; examine residuals, e.g., `plot(model)`, `qqnorm()`, `qqline()`

✓ transform DV, e.g. `log()`
✓ add interaction or polynomial to IV to account for non-linearity

## No strong multicollinearity

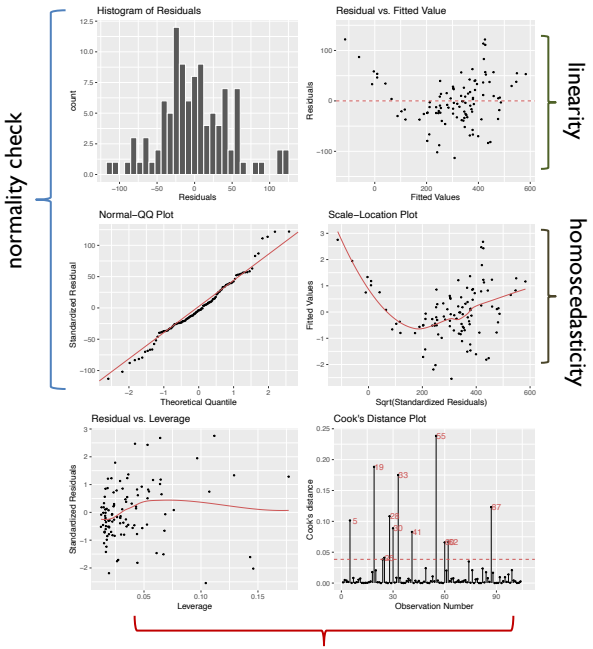
- are IV's strongly correlated ($r \geq 0.70$)?
- assess graphically or check correlations among IV's
- test variance inflation factor after running model, e.g. `vif()`; VIF should be < 5

✓ remove 1 of the highly correlated IV's – often IV with lowest correlation with DV
✓ combine correlated IV's into a single variable

## Constant variance or homoscedasticity – LM's

- evaluate graphically: does spread of DV increase over continuous IV's?
- test equality of variance of DV for nominal IV's, e.g., `var.test()`
- assess with residual plot: do standardized residuals look like a cloud of points without pattern?

✓ transform DV transform, e.g., `log()`
✓ use a different model, e.g., weighted regression




## Link function correctly specified – GLM's

- evaluate relationship between mean and variance; for Poisson and binomial count models no overdispersion, $\phi > 2$, e.g., `dispersiontest()`, `check_overdispersion()`?
- check model goodness of fit, e.g., `pchisq()`
- check relative fit of model, e.g., `pr2()`, `r.squaredGLMM()`

✓ use quasipoisson or quasibinomial models to scale standard errors of coefficients
✓ employ negative binomial model for counts, e.g., `glm.nb()`
✓ add an observation-level random effect to account for extra variance

## No highly influential observations

- assess graphically with leverage plot; `plot(model)`
- calculate Cook's distance, e.g., `cooks.distance()`, observations with cook's distance > 4 times the mean may be influential
- check with `outlierTest()`

✓ check for data entry or transcription errors
✓ remove extreme or highly influential datapoints (must have a reason to do so!)
✓ present results with and without highly influential datapoints

# example 1 owls

The data, taken from Zuur et al. (2009), quantify begging among owl nestlings in different nests prior to the arrival of a provisioning parent as a function of food treatment (deprived or satiated), the sex of the parent, and arrival time. The total number of calls from the nest is recorded.

```
summary(Owls) # from glmmTMB
Owls <- transform(Owls, Nest=reorder(Nest, NegPerChick),
                  NCalls=SiblingNegotiation,
                  FT=FoodTreatment)
```

# example 1 owls

*Does the number of calls per nestling depend on whether the nestlings are deprived or satiated with food or whether their mother or father are delivering the food? And does it depend on arrival time?*

*Model the mean number of calls per nestling.*

- *NCalls*: number of calls from the nest
- *Nest*: factor describing individual nest locations
- *FT* (factor): food treatment, Deprived or Satiated
- *SexParent* (factor): sex of provisioning parent, Female or Male
- *ArrivalTime*: a numeric vector
- *BroodSize*: number of owlets in brood

# example 1 owls

```
## ZIP
owl1 <-  glmmTMB(NCalls ~ (FT + ArrivalTime) * SexParent + (1|Nest),
               offset = log(BroodSize), ziformula = ~1,
               family = poisson, data = Owls)
owl2 <- update(owl1, ~. - ArrivalTime:SexParent)
owl3 <- update(owl2, ~. - FT:SexParent)
owl4 <- update(owl3, ~. -SexParent)

bbmle::AICctab(owl1, owl2, owl3, weights = TRUE)
res <- simulateResiduals(fittedModel = owl2)
plot(res)

## ZINB
owl5 <- update(owl4, family = nbinom2)
bbmle::AICctab(owl1, owl2, owl3, owl4, owl5, weights = TRUE)
res <- simulateResiduals(fittedModel = owl5)
plot(res)

## Hurdle model
owl6 <- update(owl5, family = truncated_nbinom2)

bbmle::AICctab(owl1, owl2, owl3, owl4, owl5, owl6, weights = TRUE)
```

# example 1 owls

```
## ZINB
> summary(owl5)
 Family: nbinom2  ( log )
Formula:          NCalls ~ FT + ArrivalTime + (1 | Nest)
Zero inflation:          ~1
Data: Owls
 Offset: log(BroodSize)

Random effects:

Conditional model:
 Groups Name        Variance Std.Dev.
 Nest   (Intercept) 0.06072  0.2464
Number of obs: 599, groups:  Nest, 27

Dispersion parameter for nbinom2 family ():  2.3

Conditional model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.81255    0.48932   5.748 9.04e-09 ***
FTSatiated  -0.30438    0.08711  -3.494 0.000475 ***
ArrivalTime -0.08227    0.01978  -4.159 3.19e-05 ***
---
Zero-inflation model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.2727     0.1214  -10.49   <2e-16 ***
---
```

```
## Hurdle
> summary(owl6)
 Family: truncated_nbinom2  ( log )
Formula:          NCalls ~ FT + ArrivalTime + (1 | Nest)
Zero inflation:          ~1
Data: Owls
 Offset: log(BroodSize)

Random effects:

Conditional model:
 Groups Name        Variance Std.Dev.
 Nest   (Intercept) 0.05663  0.238
Number of obs: 599, groups:  Nest, 27

Dispersion parameter for truncated_nbinom2 family (): 2.54

Conditional model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.60604    0.46899   5.557 2.75e-08 ***
FTSatiated  -0.20433    0.07853  -2.602  0.00927 **
ArrivalTime -0.07407    0.01896  -3.908 9.32e-05 ***
---
Zero-inflation model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.0437     0.0931  -11.21   <2e-16 ***
```

# example 1

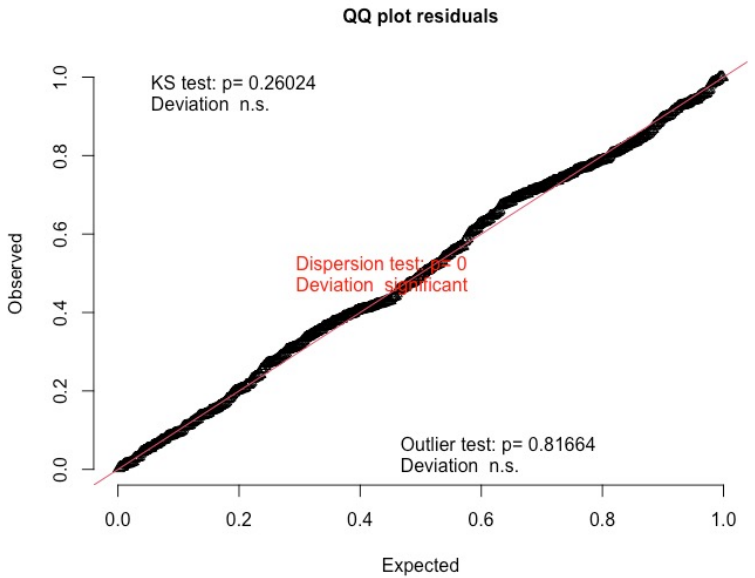
owls

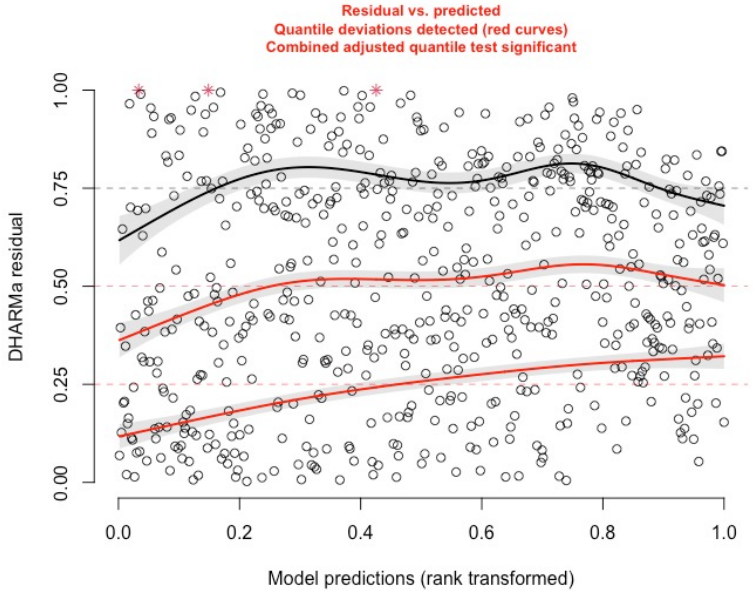
```
## ZINB with variables for structural 0's

owl7 <- glmmTMB(NCalls ~ FT + ArrivalTime + (1 | Nest),
                offset = log(BroodSize),
                 ziformula = ~SexParent,
                family = nbinom2, data = Owls)

owl8 <- update(owl7, ziformula = ~ SexParent + FT +
        ArrivalTime + (1|Nest))
```

```
## ZINB with variables for structural 0's

> bbmle::AICctab(owl6, owl7, owl8, weights = TRUE)

     dAICc df weight
owl8   0.0 10 1
owl7  93.0 7  <0.001
owl6 112.3 6  <0.001
```

```
> summary(owl8)
 Family: nbinom2  ( log )
Formula:          NCalls ~ FT + ArrivalTime + (1 | Nest)
Zero inflation:          ~SexParent + FT + ArrivalTime + (1 | Nest)
Data: Owls
 Offset: log(BroodSize)

     AIC      BIC   logLik deviance df.resid
  3314.8   3358.8  -1647.4   3294.8      589

Random effects:

Conditional model:
 Groups Name        Variance Std.Dev.
 Nest   (Intercept) 0.05375  0.2318
Number of obs: 599, groups:  Nest, 27

Zero-inflation model:
 Groups Name        Variance Std.Dev.
 Nest   (Intercept) 1.637    1.279
Number of obs: 599, groups:  Nest, 27

Dispersion parameter for nbinom2 family (): 2.49

Conditional model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  2.66407    0.47020   5.666 1.46e-08 ***
FTSatiated  -0.19556    0.07970  -2.454   0.0141 *
ArrivalTime -0.07668    0.01905  -4.025 5.69e-05 ***
---


Zero-inflation model:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -8.86991    1.80842  -4.905 9.35e-07 ***
SexParentMale -0.53570    0.28274  -1.895 0.058132 .
FTSatiated     2.26740    0.33176   6.835 8.23e-12 ***
ArrivalTime    0.26146    0.06792   3.850 0.000118 ***
```

DHARMa residual



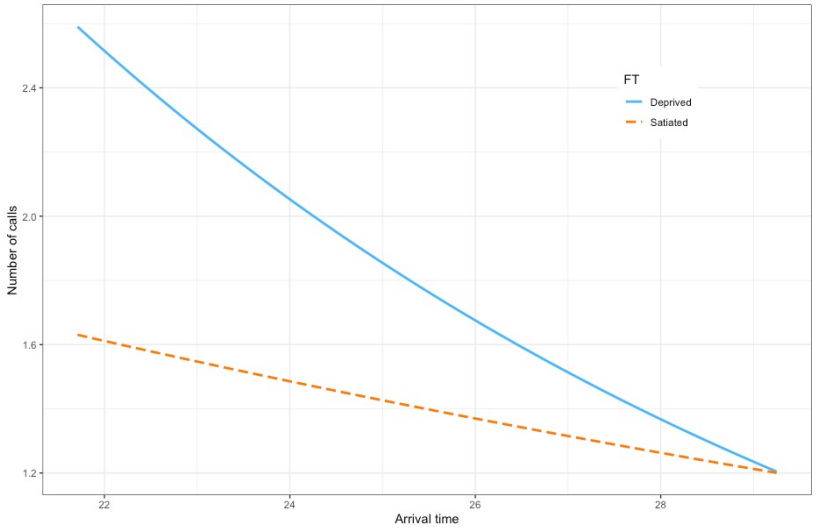



# example 1 owls

```
> owl9 <- glmmTMB(NCalls ~ FT * ArrivalTime + (1|Nest),
                  offset = log(BroodSize), ziformula = ~1,
                  family = nbinom2, data = Owls)
> summary(owl9)
```

```
 Family: nbinom2  ( log )
Formula:          NCalls ~ FT * ArrivalTime + (1 | Nest)
Zero inflation:          ~1
Data: Owls
 Offset: log(BroodSize)

Random effects:

Conditional model:
 Groups Name        Variance Std.Dev.
 Nest   (Intercept) 0.05415  0.2327
Number of obs: 599, groups:  Nest, 27

Dispersion parameter for nbinom2 family (): 2.35

Conditional model:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)             3.28595    0.58500   5.617 1.94e-08 ***
FTSatiated             -1.78643    1.01982  -1.752   0.0798 .
ArrivalTime            -0.10158    0.02370  -4.285 1.82e-05 ***
FTSatiated:ArrivalTime  0.06096    0.04177   1.459   0.1445
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Zero-inflation model:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.2559     0.1191  -10.55   <2e-16 ***
```

```
interactions::interact_plot(owl9, pred = ArrivalTime,
  modx = FT) + theme_bw() +
   theme(legend.position = c(0.8, 0.8)) +
     labs(x = "Arrival time", y = "Number of calls")
```



# Questions?

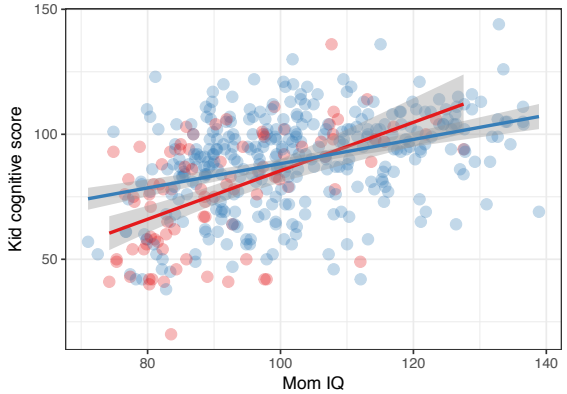
# ENV 710

overview

Duke | NICHOLAS SCHOOL OF THE ENVIRONMENT

# example 1 interaction

```
fit.3: lm(kid.score ~ mom.hs + mom.iq + mom.age + mom.hs:mom.iq)

                Estimate  Std. Error  t value     Pr(>|t|)
   (Intercept)  -20.5989    16.1912   -1.272      0.203983
   mom.hs        52.8326    15.4050    3.430      0.000663 ***
   mom.iq         0.9850     0.1491    6.607      1.17e-10 ***
   mom.age        0.3524     0.3301    1.068      0.286338
   mom.hs:mom.iq -0.5061     0.1635   -3.096      0.002092 **

   Residual standard error: 17.97 on 429 degrees of freedom
   Multiple R-squared:  0.2321,      Adjusted R-squared:  0.225
   F-statistic: 32.42 on 4 and 429 DF,  p-value: < 2.2e-16
```




no high school vs. high school

We identified a significant negative interaction between high school graduation and a mother's IQ ($t = -3.10$, $p < 0.001$). As the mother's IQ increases, the importance of high school graduation to child cognitive scores decreases. For example, at an IQ of 80, graduation increases the average test score by 12 points; whereas at an IQ of 100, graduation increases the test score by less than two points. A child's cognitive score increased by a third of a point for each additional year of age of the mother, but this effect was not statistically significant (est = 0.352, $t = 1.07$, $p = 0.286$).

# example 1 interaction

```
cfs <- c(-20.6, 52.83, 0.98, 0.35, -0.51)
ks_fun <- function(iq, age, hs){
  cfs[1] + cfs[2]*hs + cfs[3]*iq + cfs[4]*age + cfs[5]*hs*iq
}

# iq = 100
ks_fun(iq = 100, age = 22, hs = 1)
ks_fun(iq = 100, age = 22, hs = 0)
# iq = 80
ks_fun(iq = 80, age = 22, hs = 1)
ks_fun(iq = 80, age = 22, hs = 0)
# iq = 50
ks_fun(iq = 50, age = 22, hs = 1)
ks_fun(iq = 50, age = 22, hs = 0)
```

We identified a significant negative interaction between high school graduation and a mother's IQ ($t = -3.10$, $p < 0.001$). As the mother's IQ increases, the importance of high school graduation to child cognitive scores decreases. For example, at an IQ of 80, graduation increases the average test score by 12 points; whereas at an IQ of 100, graduation increases the test score by less than two points. A child's cognitive score increased by a third of a point for each additional year of age of the mother, but this effect was not statistically significant (est = 0.352, $t = 1.07$, $p = 0.286$).

# Questions?

# example 1 arsenic

## Problem 1

Aresenic contamination is a problem that affects millions of people worldwide, as poisoning occurs in multicellular life if too much arsenic is consumed. In Arahazar upazila, Bangladesh, researchers labeled wells with their level of arsenic and an indication of whether the well was "safe" or "unsafe." People using unsafe wells were encouraged to switch to a different well. After several years, the researchers returned to determine whether each household using an unsafe well had changed to a safe well. The dataset *wells.txt* contains the following variables related to this research:

`switch`: whether or not the household switched from an unsafe well to a safe
`well`: no or yes
`arsenic`: the level of arsenic contamination in the household's original well, in hundreds of micrograms per liter
`dist`: distance in meters to the closest known safe well
`educ`: education in years of the head of the household.

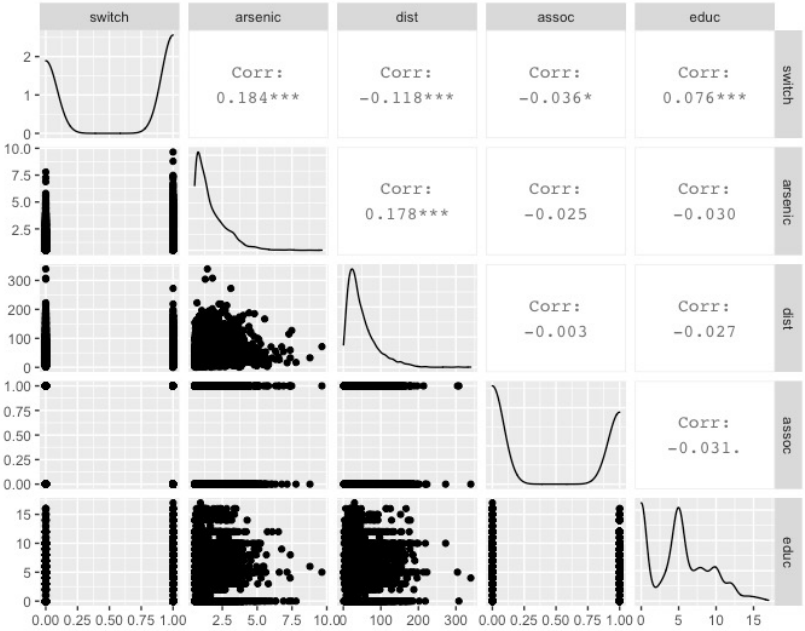
1. Rescale `dist` to 100'm of meters from a well rather than a per meter effect so that the effect is more sensible.
2. Start with a full model, including interactions, and find the best-fitting model.
3. Graph the effect of one of the significant independent variables on the *probability* of switching wells.

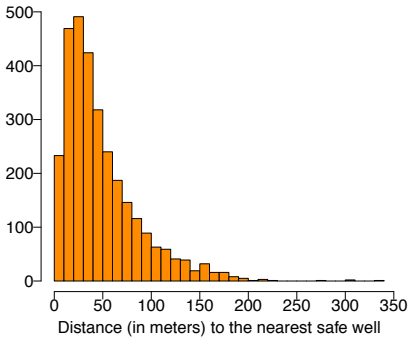
Download the data from Sakai:

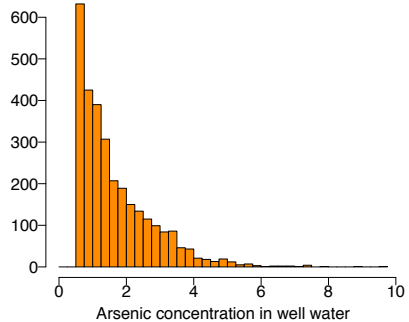
```
wells <- read.table ("wells.txt")
```

# example 1 arsenic

- explore data: `switch` is binary, therefore run logistic regression
- no signs of multicollinearity in IV's



# example 1 arsenic

```
fit.1 <- glm (switch ~ dist, family=binomial(link="logit"),
data = wells)

                coef.est coef.se
(Intercept)  0.61     0.06
dist        -0.01     0.00
---
n = 3020, k = 2  residual deviance = 4076.2, null deviance =
4118.1 (difference = 41.9)
```

```
dist100 <- with(wells, dist/100)

fit.2 <- glm (switch ~ dist100,
family=binomial(link="logit"), data = wells)

coef.est coef.se
(Intercept)  0.61     0.06
dist100     -0.62     0.10
---
n = 3020, k = 2  residual deviance = 4076.2, null deviance =
4118.1 (difference = 41.9)
```

- `dist` is originally expressed in m, but will going an extra 1 m actually have an effect on whether someone goes to a well?
- divide `dist` by 100 so coefficients are expressed in 100's of m
- these models show the change in coefficients after scaling `dist`

- model effect of all variables on probability of switching: *switch ~ arsenic * distance * education*

```
# model testing
well0 <- glm(switch ~ dist100*educ*arsenic, data = wells, family = binomial)
 well1 <- update(well0,.~.-dist100:educ:arsenic)
  well2 <- update(well1,.~. -educ:arsenic)
   well3 <- update(well2, .~. -dist100:arsenic)
```

```
anova(well0, well1, well2, well3, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: switch ~ dist100 * educ * arsenic
## Model 2: switch ~ dist100 + educ + arsenic + dist100:educ + dist100:arsenic +
##     educ:arsenic
## Model 3: switch ~ dist100 + educ + arsenic + dist100:educ + dist100:arsenic
## Model 4: switch ~ dist100 + educ + arsenic + dist100:educ
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      3012     3889.6
## 2      3013     3891.7 -1  -2.1553  0.14208
## 3      3014     3894.5 -1  -2.7334  0.09827 .
## 4      3015     3896.2 -1  -1.6732  0.19583
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- run all models and compare with likelihood ratio test
- model `well3` fits the data the best

# example 1 arsenic

```
summary(well3)
```

```
##
## Call:
## glm(formula = switch ~ dist100 + educ + arsenic + dist100:educ,
##     family = binomial, data = wells)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6603  -1.2085   0.7535   1.0613   1.9448
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.0004957  0.1096145   0.005 0.996392
## dist100      -1.3898523  0.1718840  -8.086 6.17e-16 ***
## educ         -0.0020771  0.0152548  -0.136 0.891693
## arsenic       0.4805993  0.0419866  11.446  < 2e-16 ***
## dist100:educ  0.0956362  0.0256798   3.724 0.000196 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4118.1  on 3019  degrees of freedom
## Residual deviance: 3896.2  on 3015  degrees of freedom
```

- McFadden $R^2$ very low, want it to be between 0.2-0.4
- model doesn't predict data well

```
pR2(well3)
```

```
## fitting null model for pseudo-r2
##           llh       llhNull            G2      McFadden          r2ML
## -1.948075e+03 -2.059050e+03  2.219491e+02  5.389600e-02  7.085742e-02
##          r2CU
##  9.520471e-02
```

```
pchisq(well3$null.deviance-well3$deviance, well3$df.null-well3$df.residual,
       lower.tail = F)
```

```
## [1] 7.136531e-47
```

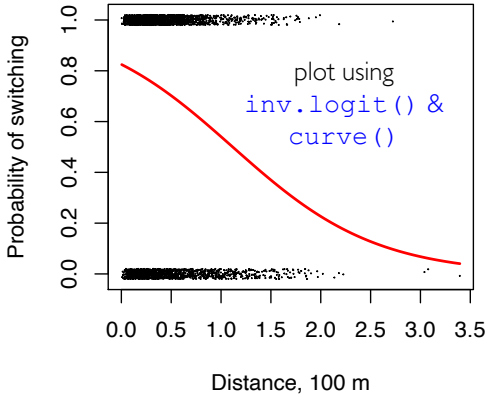
## Conclusions

- The odds of switching wells increases by 62% with every additional unit of arsenic (odds = 1.62, $z$ = 11.46, $p < 0.001$).
- The effect of education increases with distance from the well by approximately 10%, and vice versa (odds = 1.10, $z$ = 3.72, $p < 0.001$).
- There is a significant lack of fit of the model to the data (residual deviance = 3896, df = 3015).

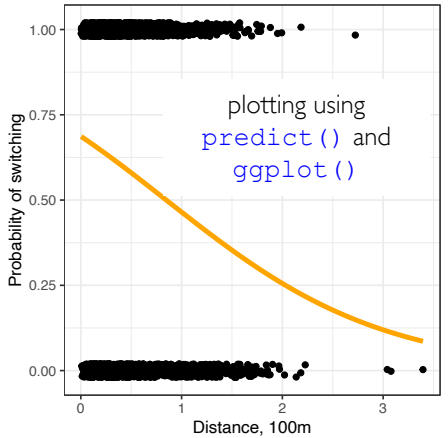
# example 1 arsenic

- graph the effects of `dist100` on `switch` at the mean arsenic level

```
# plotting distance
with(wells, plot(dist100, jitter(switch, 0.1), cex = 0.1, pch = 20,
                 xlab = "Distance, 100 m", ylab = "Probability of switching"))
 x <- with(wells, seq(min(dist100), max(dist100), length = 100))
 meduc <- mean(wells$educ)
  marsenic <- mean(wells$arsenic)
  curve(inv.logit(cbind(1, x, meduc, marsenic, meduc*marsenic) %*% coef(well3)), add = T,
        col = "red", lwd = 2)
```




```
 yb <- predict.glm(well3, data.frame(dist100 = x, educ = meduc, arsenic = marsenic),
                   type = "response")
  yb <- data.frame(cbind(myline = yb, dist = x, n = c(1:length(yb))))

  gg.well <- ggplot(data = wells, aes(x = dist100, y = jitter(switch, 0.1))) +
              geom_point() + geom_line(data = yb, aes(x = x, y = myline), lwd = 1.5,
                                       color = "orange", size = 0.8) +
               xlab("Distance, 100m") + ylab("Probability of switching") +
                theme_bw()
  gg.well
```



# example 2 horseshoe crabs

## Problem 2

Horseshoe crabs are marine water arthropods. During the breeding season, horseshoe crabs migrate to shallow coastal waters. A male selects a female and clings to her back. Often, several males surround the female and all fertilize together. In a study on horseshoe crabs, each female horseshoe crab had a male crab attached to her in her nest. The study investigated factors that influence whether the female crab had any other males, called satellites, residing near her. Explanatory variables investigated include:

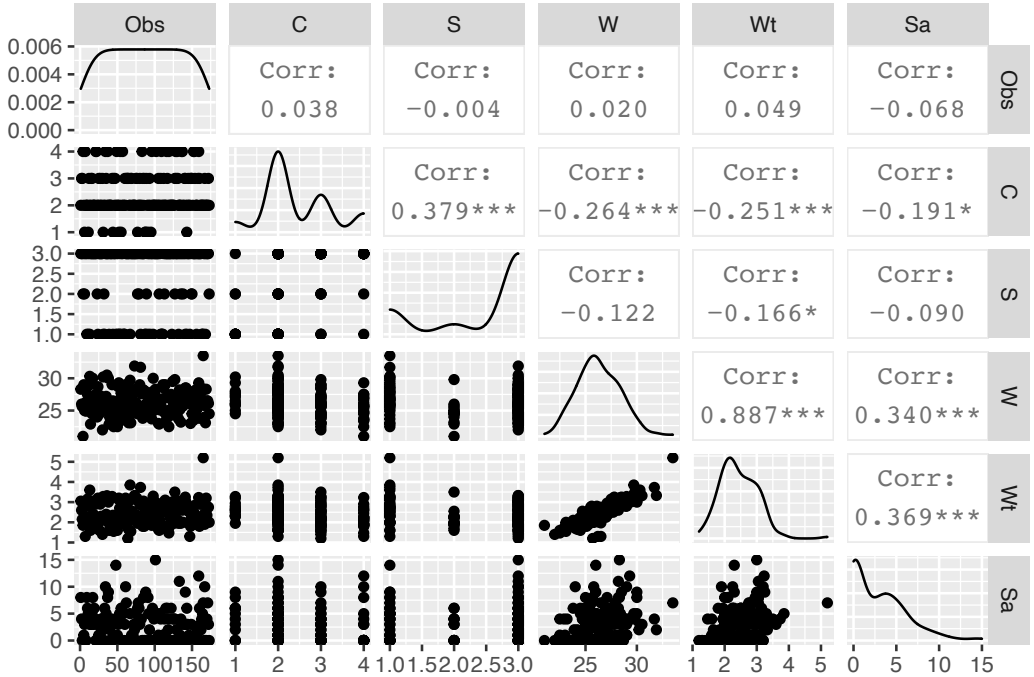
- `Sa`: the number of satellite males
- `C`: female crab's color
- `S`: spine condition
- `Wt`: weight (of the female)
- `W`: carapace width

1. Note that in this dataset `Wt` and `W` are highly correlated, so just model the effect of color, `C`, and carapace width,`W`, on the number of satellite males, `Sa`. Also, only model the main effects, no interactions.
2. Plot the effect of `W` on the number of satellites, `Sa` at two different levels of color.

- model the effects of carapace width on number of satellites

```
crab <- read.table("crab2.txt")
```

# example 2 horseshoe crabs




- explore the data
- high correlation b/n `W` and `Wt`, can only use one of them
- `Sa` seems to have a Poisson distribution and looks positively related to `W`

# example 2 horseshoe crabs

- model data first with just `W`
- high ratio of residual deviance to df's indicates lack of it

```
# model with weight
crb1 <- glm(Sa ~ W, data = crab, family = poisson(link = log))
 summary(crb1)
```

```
## glm(formula = Sa ~ W, family = poisson(link = log), data = crab)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8526  -1.9884  -0.4933   1.0970   4.9221
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.30476    0.54224  -6.095  1.1e-09 ***
## W            0.16405    0.01997   8.216  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 632.79  on 172  degrees of freedom
## Residual deviance: 567.88  on 171  degrees of freedom
## AIC: 927.18
```

```
pchisq(deviance(crb1), crb1$df.residual,lower = F)
```

```
## [1] 4.490964e-44
```

# example 2 horseshoe crabs

```
# model with weight and color
crb2 <- glm(Sa ~ W + factor(C), data = crab, family = poisson(link = log))

anova(crb1, crb2, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Sa ~ W
## Model 2: Sa ~ W + factor(C)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       171     567.88
## 2       168     559.34  3   8.5338  0.03618 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(crb1, crb2)
```

```
##      df      AIC
## crb1  2 927.1762
## crb2  5 924.6425
```

- Model with C and W fits better than model with just W, residual deviance has decreased and AIC is lower
- crb2 has significant overdispersion
- one reason for overdispersion is heterogeneity where subjects within each covariate combination still differ greatly (i.e., crabs with similar width still have different number of satellites)
- another reason is missing explanatory variables

```
# model doesn't fit well, examine overdispersion
pchisq(deviance(crb2), crb2$df.residual, lower = F)
```

```
## [1] 1.471037e-43
```

```
dispersiontest(crb2)
```

```
##
##  Overdispersion test
##
## data:  crb2
## z = 5.3255, p-value = 5.033e-08
## alternative hypothesis: true dispersion is greater than 1
## sample estimates:
## dispersion
##   3.154127
```

- when do you use an offset?
- when do you use random effects?
- when do you include interactions?

# example 2 horseshoe crabs

- model with quasipoisson and by adding an observation-level random effect

```
# use quasipoisson to scale p-values
crb3 <- glm(Sa ~ W + factor(C), data = crab, family = quasipoisson)
 summary(crb3)
```

```
##
## Call:
## glm(formula = Sa ~ W + factor(C), family = quasipoisson, data = crab)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0415  -1.9581  -0.5575   0.9830   4.7523
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.65004    1.05740  -2.506   0.0132 *
## W            0.14934    0.03748   3.985   0.0001 ***
## factor(C)2  -0.19969    0.27628  -0.723   0.4708
## factor(C)3  -0.43636    0.31713  -1.376   0.1707
## factor(C)4  -0.44736    0.37604  -1.190   0.2359
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 3.233628)
##
##     Null deviance: 632.79  on 172  degrees of freedom
## Residual deviance: 559.34  on 168  degrees of freedom
## AIC: NA
```

```
# model overdispersion with random effect
crab$Obs <- as.factor(crab$Obs)

# this model doesn't want to easily converge
crb4 <- glmer(Sa ~ W + factor(C) + (1|Obs), data = crab, family = poisson)
 summary(crb4)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: poisson  ( log )
## Formula: Sa ~ W + factor(C) + (1 | Obs)
##    Data: crab
##
##      AIC      BIC   logLik deviance df.resid
##    770.9    789.8   -379.5    758.9      167
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.07998 -0.74220  0.03087  0.33583  1.03057
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  Obs    (Intercept) 0.9872   0.9936
## Number of obs: 173, groups:  Obs, 173
##
## Fixed effects:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.74201    1.35735  -3.494 0.000477 ***
## W            0.21597    0.04844   4.459 8.25e-06 ***
## factor(C)2  -0.28785    0.35775  -0.805 0.421051
## factor(C)3  -0.56010    0.39147  -1.431 0.152502
## factor(C)4  -0.81781    0.45836  -1.784 0.074389 .
## ---
```

# example 2 horseshoe crabs

```
# use negative binomial
crb5 <- glm.nb(Sa ~ W + factor(C), data = crab)
 summary(crb5)
```

```
##
## Call:
## glm.nb(formula = Sa ~ W + factor(C), data = crab, init.theta = 0.9320986132,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8714  -1.3769  -0.2663   0.4410   2.2496
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.35239    1.26871  -2.642  0.00823 **
## W            0.17839    0.04529   3.939 8.18e-05 ***
## factor(C)2  -0.27593    0.35235  -0.783  0.43356
## factor(C)3  -0.52033    0.38396  -1.355  0.17536
## factor(C)4  -0.53603    0.43348  -1.237  0.21624
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.9321) family taken to be 1)
##
##     Null deviance: 216.56  on 172  degrees of freedom
## Residual deviance: 196.23  on 168  degrees of freedom
## AIC: 760.6
```

- model with negative binomial distribution
- AIC demonstrates that negative binomial model fits the data best
- could take out C, but going to leave it
- each cm of carapace width increases number of satellite males by ~20%
- compared to color 1, other colors reduce number of satellite males by ~40-25%

```
AIC(crb2, crb4, crb5)
```

```
##      df      AIC
## crb2  5 924.6425
## crb4  6 770.9076
## crb5  6 760.5958
```

```
# crb5 coefficients as multiplicative effects
  exp(coef(crb5))
```

```
## (Intercept)           W  factor(C)2  factor(C)3  factor(C)4
##  0.03500076  1.19529162  0.75886462  0.59432532  0.58506859
```

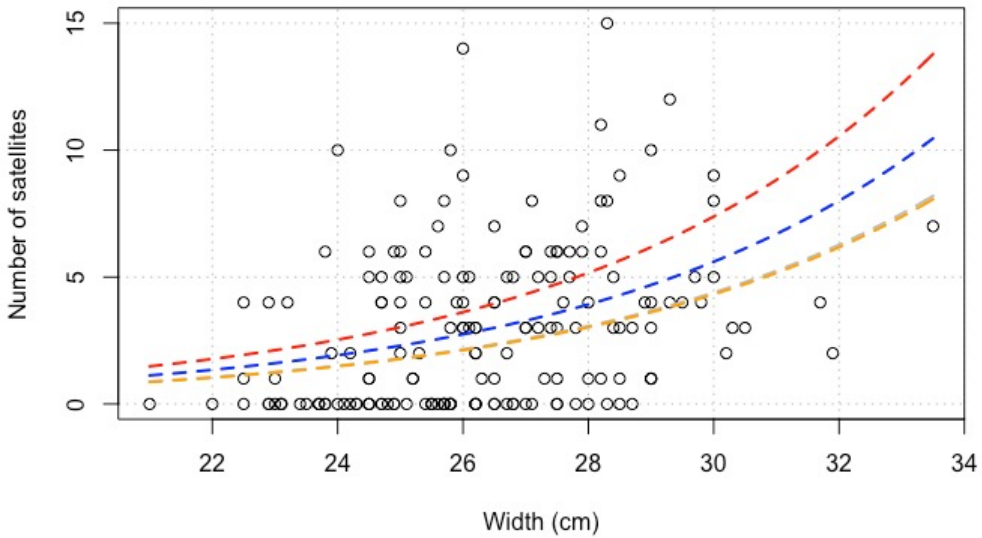
Conclusions:

- Color of the carapace does not have a significant effect on the number of satellites near a female crab.
- With every addition cm of carapace width, the number of satellite males increases by roughly 20% ($z = 3.94$, $p < 0.001$).

# example 2 horseshoe crabs

- plot results of negative binomial model
- only 3 of 4 lines appear because the effect of colors 3 and 4 are nearly identical (grey line is under yellow line)

```
plot(x = crab$W, y = crab$Sa, xlab = "Width (cm)", ylab =
       "Number of satellites", panel.first = grid(col = "gray",
                                                  lty = "dotted"))

 x <- with(crab, seq(min(W), max(W), length = nrow(crab)))

 crs <- coef(crb5)
 curve(exp(crs[1]+crs[2]*x),
       col = "red", add = TRUE, lty = 2, lwd = 2)
   curve(exp(crs[1]+crs[2]*x + crs[3]),
       col = "blue", add = TRUE, lty = 2, lwd = 2)
     curve(expr = exp(crs[1]+crs[2]*x + crs[4]),
       col = "grey", add = TRUE, lty = 2, lwd = 2)
      curve(expr = exp(crs[1]+crs[2]*x + crs[5]),
       col = "orange", add = TRUE, lty = 2, lwd = 2)
```



# example 3 algal bloom

## Problem 3

The dataset `bloom` contains information on densities of `algae` and measurements of `sunlight`, and `nutrients` at two categories of water depth, `depthcat` (s = shallow, d = deep). Algae require warmth, sunlight, and nutrients to grow and reproduce, so they live in the upper 60 to 90 meters (200 to 300 feet) of ocean water. Hypothesizing that warmer ocean waters resulting from global warming might alter the effects of these factors, researchers sampled algal densities (mg/L) at 20 randomly chosen sites (`site`: A through T) along the US eastern coast, taking 10 samples at each site. For each sample, they measured algal density, temperature (°C), nutrient load (mg/L), and recorded the depth category. The researchers are interested in the effects of temperature, nutrients and depth on algal density for the entire coast, not particular sites. In this problem, determine what factors drive algal density on the US eastern coast accounting for variation across sampling sites.

1. What are the null and alternative hypotheses for your analysis?

- $H_0$: Nutrients and sunlight have no effect on algal density.
- $H_a$: One of the independent variables, nutrients or sunlight, affects algal density.

2. *Without including interactions*, what is the minimum adequate model? Write out the model as an equation *or* R code.

# example 3 algal bloom

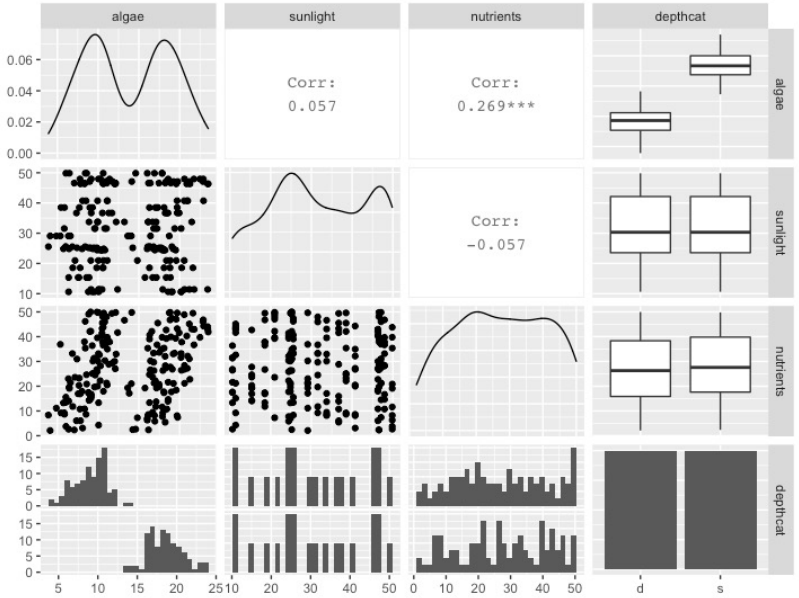
```
require(lmerTest)
bloom <- read.csv("bloom.csv")

al1 <- lmer(algae ~ sunlight + nutrients + factor(depthcat) + (1|site),
            data = bloom, REML = F)
al2 <- update(al1, .~.-sunlight)
al3 <- update(al2, .~. -factor(depthcat))

anova(al1, al2, al3)
```

```
## Data: bloom
## Models:
## al3: algae ~ nutrients + (1 | site)
## al2: algae ~ nutrients + factor(depthcat) + (1 | site)
## al1: algae ~ sunlight + nutrients + factor(depthcat) + (1 | site)
##     Df     AIC     BIC  logLik deviance    Chisq Chi Df Pr(>Chisq)
## al3  4 1219.07 1232.26 -605.53  1211.07
## al2  5  616.92  633.41 -303.46   606.92 604.1498      1     <2e-16 ***
## al1  6  617.57  637.36 -302.79   605.57   1.3448      1     0.2462
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(al1, al2, al3)
```

```
##     df       AIC
## al1  6  617.5729
## al2  5  616.9177
## al3  4 1219.0675
```

- `algae` is continuous, so choose linear model, but it also appears bimodal?
- no multicollinearity
- run all models; `al3` doesn't converge because `depthcat` was taken out
- `al2` is best model, assuming we want the minimum adequate model



# example 3 algal bloom

- `depthcat` has a very large effect
- residuals look good, although there is still a little bit of pattern even after including `depthcat`

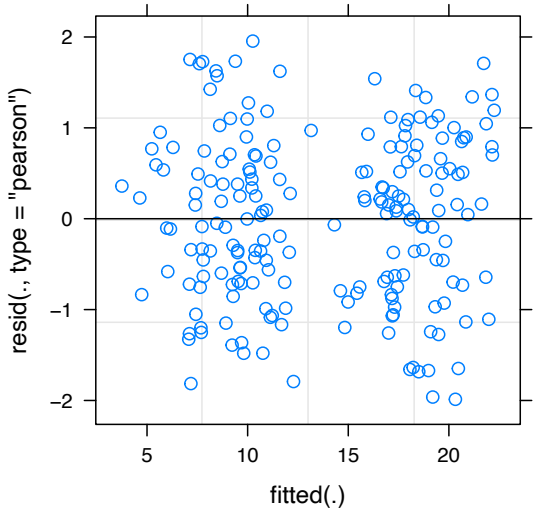
```
summary(al2)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: algae ~ nutrients + factor(depthcat) + (1 | site)
##    Data: bloom
##
## REML criterion at convergence: 618.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.10554 -0.74506  0.08688  0.72820  2.06662
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  site     (Intercept) 2.2041   1.4846
##  Residual             0.8935   0.9452
## Number of obs: 200, groups:  site, 20
##
## Fixed effects:
##                      Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)         6.410e+00  3.716e-01 2.738e+01   17.25 3.09e-16 ***
## nutrients           9.866e-02  5.098e-03 1.797e+02   19.35  < 2e-16 ***
## factor(depthcat)s 9.379e+00  1.337e-01 1.780e+02   70.14  < 2e-16 ***
## ---
```

```
require(MuMIn)
 r.squaredGLMM(al2)
```

```
##             R2m       R2c
## [1,] 0.8869737 0.9673982
```

```
plot(al2)
```

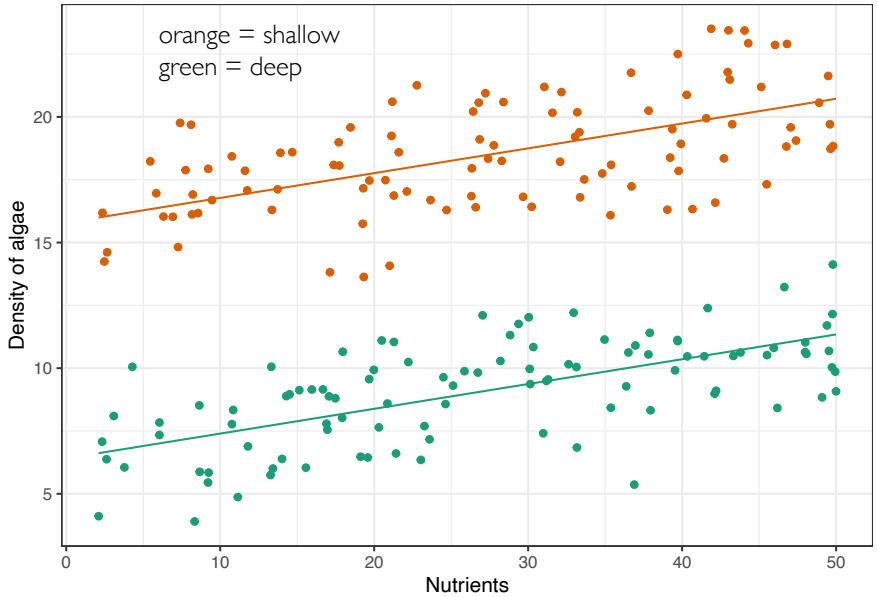



## Conclusions

- Density of algae increases by approximately 0.10 with each additional mg/l of nutrients ($t$ = 19.35, $p < 0.001$).
- Shallow depths have on average 9.38 mg/L high algae density than deeper waters ($t$ = 70.14, $p < 0.001$).
- Water depth and nutrients account for 88.6% of the variation in algae density. Algae density also varied across sites ($\sigma$ = 1.48).

# example 3 algal bloom

```
cfs <- fixef(al2)
 xx <- with(bloom, seq(min(nutrients), max(nutrients), length = nrow(bloom)))
  yy <- predict(al2)
   yys <- cfs[1] + cfs[2]*xx + cfs[3]
   yyd <- cfs[1] + cfs[2]*xx + cfs[3]*0
    bl.df <- data.frame(xx, yys, yyd)

 ggplot(data = bloom, aes(x = nutrients, y = algae, colour = factor(depthcat))) +
   geom_point() +
    geom_line(data = bl.df, aes(y = yys, x = xx, colour = factor(depthcat)[1] )) +
     geom_line(data = bl.df, aes(y = yyd, x = xx, colour = factor(depthcat)[2])) +
      ylab("Density of algae") + xlab("Nutrients") +
       scale_fill_brewer(palette="Dark2") +
       scale_color_brewer(palette="Dark2") +
     theme_bw() + theme(legend.position = 'none')
```



# example 4 elephants

## Problem 4

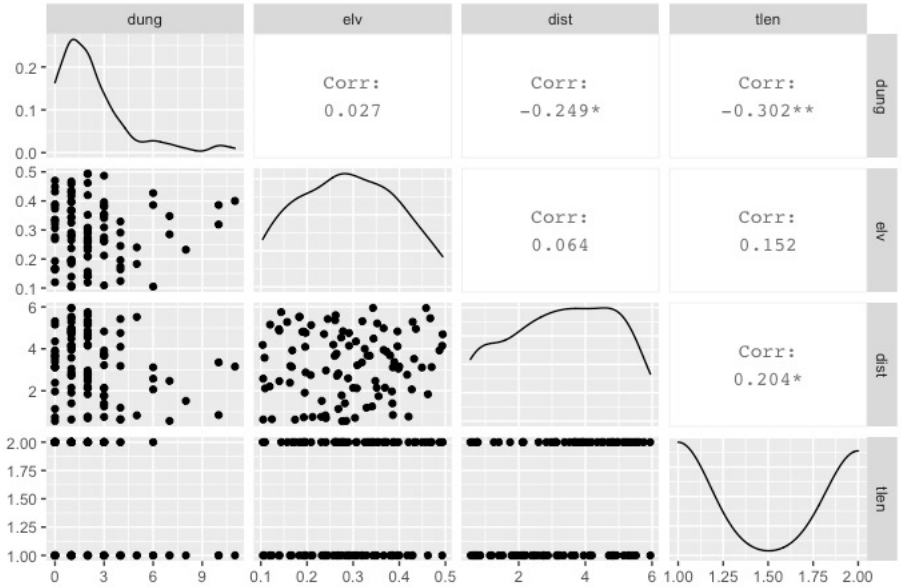
The relative abundance of elephants is related to the number of dung piles they deposit on the ground. A survey was conducted to evaluate the variables that determine the abundance of elephants by counting dung piles along transects (straight-line trails). The dataset *ele.csv* contains counts of elephant dung piles collected along transects, as well as information regarding the following:

- `dung`: count of dung piles on each transect
- `dist`: the distance of the transects from the nearest village in 10-km
- `elv`: the elevation of the transect in 100-m
- `park`: whether or not the transect is located in a national park (`out` = outside of park, `in` = inside of park)
- `tlen`: the number of km walked on each transect.

Please model the variables that affect the number of `dung` on a transect, using transect distance (`tlen`) to model dung as a rate (dung counted per km), and assessing all three independent variables.

1. Start with a model that includes all the main effects and the interaction between `elv` and `dst`. What is the minimum adequate model?
2. What is the meaning of the coefficients in your minimum adequate model?
3. Check the fit of your model to the data. Does it meet the model assumptions?

# example 4 elephants




- examine the data
- `dung` looks like it follows a Poisson distribution and the mean is small
- scatterplots suggest that maybe `elv` has a weak positive relationship with `dung` and `dist` has a negative effect on `dung`
- `tlen` only has 2 values

# example 4 elephants

- run model including offset because `tlen` (effort) has two values; model number of dung per km
- reduce model to minimum adequate model; check with AIC to confirm

```
g1 <- glm(dung ~ elv * dist + factor(park), offset = log(tlen),
          family = poisson, data = ele)
g2 <- update(g1, .~. -elv:dist)
g3 <- update(g2, .~.-elv)

anova(g1, g2, g3, test = "Chisq")
```

```
## Model 1: dung ~ elv * dist + factor(park)
## Model 2: dung ~ elv + dist + factor(park)
## Model 3: dung ~ dist + factor(park)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1        95     238.98
## 2        96     242.76 -1  -3.7765  0.05198 .
## 3        97     242.76 -1   0.0000  0.99573
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(g1, g2, g3)
```

```
##    df      AIC
## g1  5 471.5163
## g2  4 473.2929
## g3  3 471.2929
```

# example 4 elephants

- always check for overdispersion with Poisson models, especially because the residual deviance / df's is ~ 2.5 – this is a quick check of overdispersion
- dispersion test finds significant dispersion

```
summary(g3)
```

```
##
## Call:
## glm(formula = dung ~ dist + factor(park), family = poisson, data = ele,
##     offset = log(tlen))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3718  -1.0893  -0.2499   0.5083   5.2918
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      1.18830    0.15120   7.859 3.87e-15 ***
## dist            -0.19581    0.04275  -4.581 4.63e-06 ***
## factor(park)out -0.33023    0.13431  -2.459   0.0139 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 270.92  on 99  degrees of freedom
## Residual deviance: 242.76  on 97  degrees of freedom
## AIC: 471.29
##
## Number of Fisher Scoring iterations: 5
```

```
dispersiontest(g3)
```

```
##
##  Overdispersion test
##
## data:  g3
## z = 2.4003, p-value = 0.008191
## alternative hypothesis: true dispersion is greater than 1
## sample estimates:
## dispersion
##   2.935959
```

# example 4 elephants

```
g4 <- update(g3, .~.,family = quasipoisson)
summary(g4)
```

```
##
## Call:
## glm(formula = dung ~ dist + factor(park), family = quasipoisson,
##     data = ele, offset = log(tlen))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3718  -1.0893  -0.2499   0.5083   5.2918
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.18830    0.27146   4.378 3.03e-05 ***
## dist            -0.19581    0.07674  -2.552   0.0123 *
## factor(park)out -0.33023    0.24113  -1.369   0.1740
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasipoisson family taken to be 3.223232)
##
##     Null deviance: 270.92  on 99  degrees of freedom
## Residual deviance: 242.76  on 97  degrees of freedom
## AIC: NA
```

- can deal with overdispersion 3 ways: quasipoisson, negative binomial, individual-level random effect
- quasipoisson scales se's by the $\sqrt{\phi}$

```r
g5 <- glm.nb(dung ~ dist + factor(park) + offset(log(tlen)), data = ele)
 summary(g5)
```

```
##
## Call:
## glm.nb(formula = dung ~ dist + factor(park) + offset(log(tlen)),
##     data = ele, init.theta = 1.59750158, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3003  -0.7574  -0.2740   0.2131   3.0540
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.3746     0.2442   5.629 1.81e-08 ***
## dist             -0.2110     0.0662  -3.188  0.00143 **
## factor(park)out  -0.3872     0.2079  -1.862  0.06256 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(1.5975) family taken to be 1)
##
##     Null deviance: 120.06  on 99  degrees of freedom
## Residual deviance: 106.47  on 97  degrees of freedom
## AIC: 416.09
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  1.598
##           Std. Err.:  0.385
##
##  2 x log-likelihood:  -408.087
```

- account for overdispersion by using the negative binomial distribution

```
ele$n <- as.factor(seq(1,nrow(ele)))
g6 <- glmer(dung ~ dist + factor(park) + (1|n) + offset(log(tlen)),
            family = poisson, data = ele)
 summary(g6)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: poisson  ( log )
## Formula: dung ~ dist + factor(park) + (1 | n) + offset(log(tlen))
##    Data: ele
##
##      AIC      BIC   logLik deviance df.resid
##    412.6    423.0   -202.3    404.6       96
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.30644 -0.30923  0.00387  0.36283  1.21415
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  n      (Intercept) 0.6036   0.7769
## Number of obs: 100, groups:  n, 100
##
## Fixed effects:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      1.01455    0.25840   3.926 8.63e-05 ***
## dist            -0.19454    0.06847  -2.841  0.00449 **
## factor(park)out -0.39360    0.21652  -1.818  0.06908 .
## ---
```

- account for overdispersion by adding an observation-level random effect
- compare best models with and without overdispersion (g4 not included because it is quasipoisson)
- g6 with random effects wins
- dispersion gone and model accounts for 62.5% of variation

```
AIC(g3, g5, g6)
```

```
##    df      AIC
## g3  3 471.2929
## g5  4 416.0866
## g6  4 412.5852
```

```
dispersion_glmer(g6)
```

```
## [1] 1.025032
```

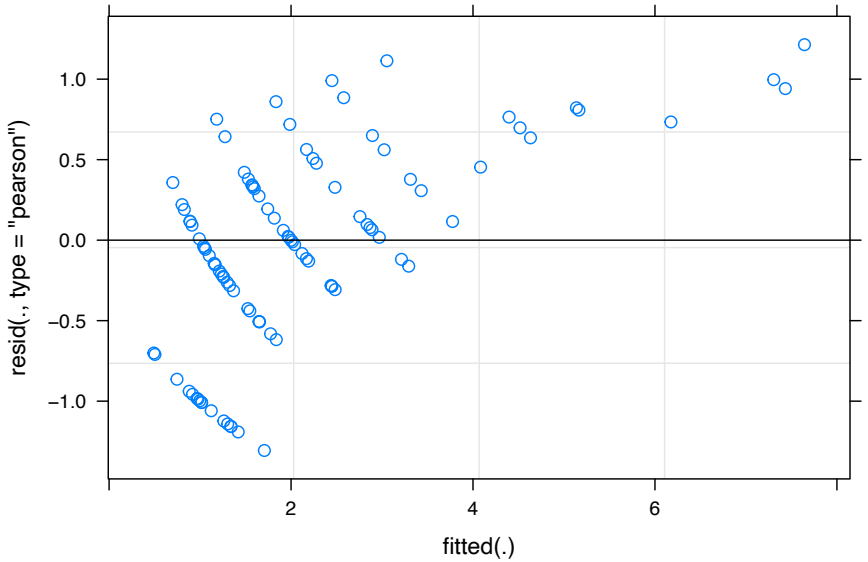
```
r.squaredGLMM(g6)
```

```
##                R2m       R2c
## delta     0.1215033 0.6254637
## lognormal 0.1299677 0.6690359
## trigamma  0.1108477 0.5706116
```

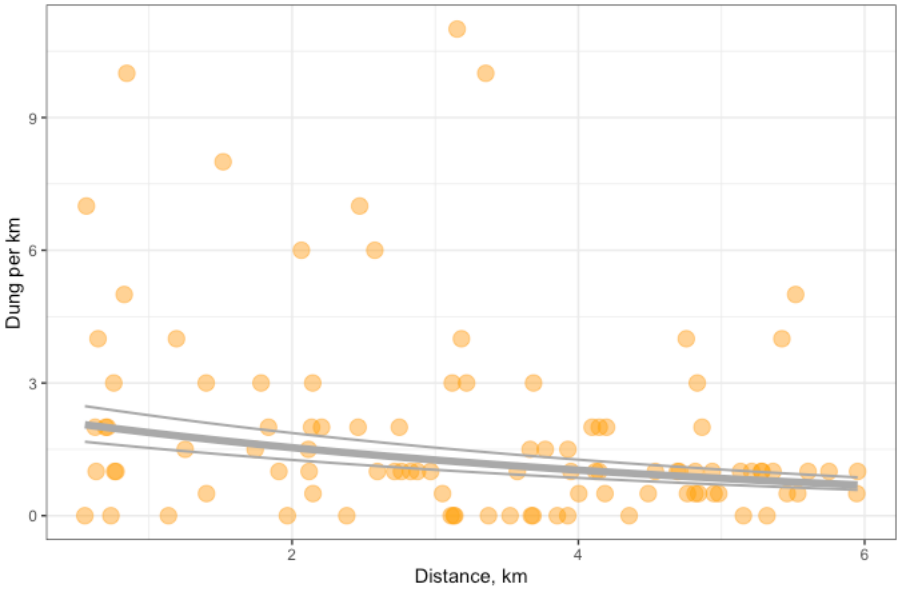
```
confint(g6)
```

```
##                      2.5 %      97.5 %
## .sig01           0.5993277  0.99300744
## (Intercept)      0.4877382  1.51330191
## dist            -0.3304976 -0.05889244
## factor(park)out -0.8257967  0.03324163
```

```
plot(g6)
```




- none of the Pearson residuals are $> |2|$, but there is structure in them




- light regression lines include the effect of being in (upper) and out (lower) of parks

```
xx <- with(ele, seq(min(dist), max(dist), length = length(dist)))
 #y <- predict(g5, newdata = data.frame(dist = xx, park = "out"), type = "response")
g7 <- glmer(dung ~ dist + (1|n) + offset(log(tlen)),
            family = poisson, data = ele)

  cfs <- fixef(g6)
  yy.out <- exp(cfs[1] + cfs[2]*xx + cfs[3])
  yy.in <- exp(cfs[1] + cfs[2]*xx + cfs[3]*0)
  yy.g7 <- exp(fixef(g7)[1] + fixef(g7)[2]*xx)
  newdat <- data.frame(xx, yy.out, yy.in, yy.g7)

  ggplot(data = ele, aes(x = dist, y = dung/tlen)) +
         geom_point(col = "orange", alpha = 0.5, size = 4) +
         geom_line(data = newdat, aes(x = xx, y = yy.g7), lwd = 2, colour = "darkgrey") +
         geom_line(data = newdat, aes(x = xx, y = yy.in), lwd = 0.7, colour = "darkgrey") +
         geom_line(data = newdat, aes(x = xx, y = yy.out), lwd = 0.7, colour = "darkgrey") +
           ylab("Dung per km") + xlab("Distance, km") + theme_bw()
```

## Conclusions

- The effect of park on dung was marginally significant, with the number of dung observed per kilometer being 32.5% lower outside of the park compared to inside the park ($z$ = -1.82, $p$ = 0.07).
- With every 10 km of distance from a village, the number of dung per km decreases by 17.5% ($z$ = -0.195, $p$ = 0.005). (Yes, ironically, the number of dung is higher near villages and decreases away from them.)


THANK YOU!