Fitting mixed models and selecting among them

QERM 514 - Homework 6 Answer Key

8 May 2020

R Markdown file

You can find the R Markdown file used to create this answer key here.

Background

This week's homework assignment focuses on fitting and evaluating linear mixed models. In particular, you will consider different forms for a stock-recruit relationship that describes the density-dependent relationship between fish spawning biomass in "brood year" t (S_t) and the biomass of fish arising from that brood year that subsequently "recruit" to the fishery (R_t).

Ricker model

The Ricker model (Ricker 1954) is one of the classical forms for describing the stock-recruit relationship. The deterministic form of the model is given by

$$R_t = S_t \exp\left[r\left(1 - \frac{S_t}{k}\right)\right]$$

where r is the intrinsic growth rate and k is the carrying capacity of the environment. In fisheries science, the model is often rewritten as

$$R_t = aS_t \exp\left(-bS_t\right)$$

where $a = \exp r$ and $b = r/k$. We can make the model stochastic by including a multiplicative error term $\epsilon_t \sim \mathrm{N}(0, \sigma^2)$, such that

$$R_t = aS_t \exp\left(-bS_t\right) \exp(\epsilon_t)$$

This model is clearly non-linear, but we can use a log-transform to linearize it. Specifically, we have

$$\begin{gathered}
\log R_t = \log a + \log S_t - bS_t + \epsilon_t \\
\Downarrow \\
\log R_t - \log S_t = \log a - bS_t + \epsilon_t \\
\Downarrow \\
\log(R_t/S_t) = \log a - bS_t + \epsilon_t \\
\Downarrow \\
y_t = \alpha - \beta S_t + \epsilon_t
\end{gathered}$$

where $y_t = \log(R_t/S_t)$, $\alpha = \log a$, and $\beta = b$.

Data

The data for this assignment come from 21 populations of Chinook salmon (*Oncorhynchus tshawytscha*) in Puget Sound. The original data come from the NOAA Fisheries Salmon Population Summary (SPS) database, which was subsequently cleaned and summarized for use in a recent paper by Bal et al. (2019). The data are contained in the accompanying file `ps_chinook.csv`, which contains the following columns:

- `pop`: name of the population
- `pop_n`: integer ID for population (1-21)
- `year`: year of spawning
- `spawners`: total number of spawning adults (1000s)
- `recruits`: total number of surviving offspring that "recruit" to the fishery (1000s)

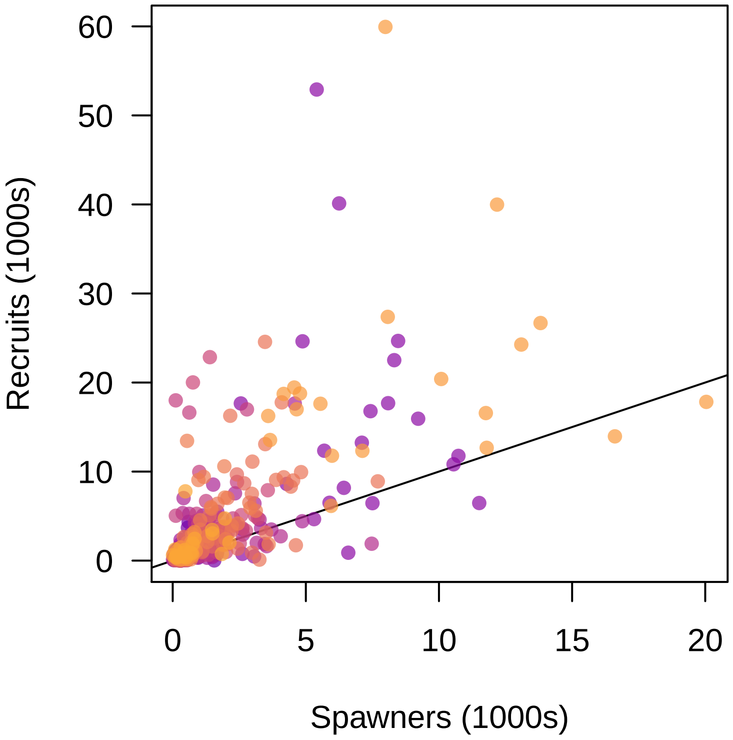
Problems

As you work through the following problems, be sure to show all of the code necessary to produce your answers. (Hint: You will need to define a new response variable before you can do any model fitting.)

```
## load the data
psc <- read.csv("ps_chinook.csv")

## number of popns
n_pops <- length(unique(psc$pop))

## number of years
n_yrs <- length(unique(psc$year))

## new response variable: log(R/S)
psc$logRS <- log(psc$recruits / psc$spawners)
```

a) Plot the number of recruits by population (y) against the number of spawners by population (x), and add a line indicating the replacement level where recruits = spawners. Describe what you see.

```
## set colors
clrs <- viridis::plasma(n_pops, alpha = 0.7, begin = 0.2, end = 0.8)

## set plot region
par(mai = c(0.9, 0.9, 0.6, 0.1))

## plot data
plot(0, 0, type = "n", las = 1,
     xlim = range(psc$spawners), ylim = range(psc$recruits),
     ylab = "Recruits (1000s)", xlab = "Spawners (1000s)")
abline(a = 0, b = 1)
for(i in 1:n_pops) {
  pdat <- psc[psc$pop_n == i,]
  points(pdat$spawners, pdat$recruits, pch = 16, col = clrs[i])
}
```



b) Fit the following model and report your estimates for α and β. Also report your estimate of σ_ϵ^2. Based on the R^2 value, does this seem like a promising model?

$$\log(R_{i,t}/S_{i,t}) = \alpha - \beta S_{i,t} + \epsilon_{i,t}$$
$$\epsilon_{i,t} \sim \mathrm{N}(0, \sigma^2_\epsilon)$$

```
## base model with global parameters
mod_base <- lm(logRS ~ spawners, data = psc)
summary(mod_base)
```

```
##
## Call:
## lm(formula = logRS ~ spawners, data = psc)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.3069 -0.5120  0.1275  0.6340  4.4641
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.60045    0.06524   9.204   <2e-16 ***
## spawners    -0.02398    0.02124  -1.129     0.26
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.114 on 418 degrees of freedom
## Multiple R-squared:  0.00304,    Adjusted R-squared:  0.0006545
## F-statistic: 1.274 on 1 and 418 DF,  p-value: 0.2596
```

The estimate of α is 0.6 and β is -0.024. The estimate of σ^2_ϵ is 1.24. The R^2 value is only 0.003, which is *very* small, so this does *not* seem to be a promising model.

---

c) Fit the following model and report your estimates for α, each of the δ_i, and β. Also report your estimate of σ^2_ϵ and σ^2_δ. Based on the R^2 value, how does this model compare to that from part (b)?

$$\log(R_{i,t}/S_{i,t}) = (\alpha + \delta_i) - \beta S_{i,t} + \epsilon_{i,t}$$
$$\delta_i \sim \mathrm{N}(0, \sigma^2_\delta)$$
$$\epsilon_{i,t} \sim \mathrm{N}(0, \sigma^2_\epsilon)$$

```
library(lme4)
## RE for alpha
mod_re_popn_alpha <- lmer(logRS ~ 1 + spawners + (1 | pop_n), data = psc)
summary(mod_re_popn_alpha)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: logRS ~ 1 + spawners + (1 | pop_n)
##    Data: psc
##
## REML criterion at convergence: 1285.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -6.0964 -0.4471  0.0452  0.5482  3.8110
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  pop_n    (Intercept) 0.07869  0.2805
##  Residual             1.18007  1.0863
## Number of obs: 420, groups:  pop_n, 21
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.67647    0.09310   7.266
## spawners    -0.06871    0.02703  -2.542
##
## Correlation of Fixed Effects:
##          (Intr)
## spawners -0.493
```

The estimate of α is 0.68 and β is -0.069. The estimates of the δ_i are

```
round(ranef(mod_re_popn_alpha)$pop_n, 3)
```

```
##    (Intercept)
## 1       -0.016
## 2       -0.230
## 3       -0.197
## 4        0.187
## 5       -0.209
## 6        0.083
## 7        0.093
## 8       -0.021
## 9       -0.031
## 10       0.253
## 11       0.139
## 12      -0.383
## 13      -0.104
## 14      -0.160
## 15      -0.104
## 16       0.117
## 17       0.323
## 18      -0.208
## 19      -0.139
## 20       0.454
## 21       0.154
```

```
## get Var(epsilon) & Var(delta)
(var_re_site <- as.data.frame(VarCorr(mod_re_popn_alpha)))
## variance of random effects
sigma2_delta <- var_re_site$vcov[1]
## variance of residuals
sigma2_epsilon <- var_re_site$vcov[2]
```

```
##        grp        var1 var2       vcov     sdcor
## 1    pop_n (Intercept) <NA> 0.07869448 0.2805254
## 2 Residual        <NA> <NA> 1.18007270 1.0863115
```

The estimate of σ^2_ϵ is 1.18 and the estimate of σ^2_δ is 0.08.

```
## R^2
SSE <- sum(residuals(mod_re_popn_alpha)^2)
SSTO <- sum((psc$logRS - mean(psc$logRS))^2)
(R2 <- 1 - SSE / SSTO)
```

```
## [1] 0.0766575
```

The R^2 value for this model is only ~0.077, which is much better than that for (b), but still quite low.

---

d) Fit the following model and report your estimates for α, each of the η_i, and β. Also report your estimate of σ^2_ϵ and σ^2_η. Based on the R^2 value, how does this model compare to that from part (c)?

$$
\begin{gathered}
\log(R_{i,t}/S_{i,t}) = \alpha - (\beta + \eta_i)S_{i,t} + \epsilon_{i,t} \\
\eta_i \sim \text{N}(0, \sigma^2_\eta) \\
\epsilon_{i,t} \sim \text{N}(0, \sigma^2_\epsilon)
\end{gathered}
$$

The trick here is to recognize that you only want a random effect for the slope η, and not the intercept, which means you need to specify the random effect as `(-1 + spawners | pop_n)`.

```
## RE for beta
mod_re_popn_beta <- lmer(logRS ~ 1 + spawners + (-1 + spawners | pop_n), data = psc)
summary(mod_re_popn_beta)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: logRS ~ 1 + spawners + (-1 + spawners | pop_n)
##    Data: psc
##
## REML criterion at convergence: 1288.5
##
```

```
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -6.2761 -0.4227  0.0607  0.5318  3.9257
##
## Random effects:
##  Groups   Name     Variance Std.Dev.
##  pop_n    spawners 0.3082   0.5552
##  Residual          1.1137   1.0553
## Number of obs: 420, groups:  pop_n, 21
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.92446    0.08902  10.385
## spawners    -0.56726    0.15951  -3.556
##
## Correlation of Fixed Effects:
##          (Intr)
## spawners -0.483
```

The estimate of α is 0.92 and β is -0.567. The estimates of the η_i are

```
round(ranef(mod_re_popn_beta)$pop_n, 3)
```

```
##    spawners
## 1    -0.274
## 2    -0.693
## 3    -0.554
## 4     0.504
## 5     0.096
## 6     0.034
## 7     0.416
## 8     0.223
## 9     0.126
## 10    0.518
## 11    0.412
## 12   -1.110
## 13   -0.419
## 14   -0.403
## 15    0.217
## 16    0.468
## 17    0.501
## 18   -0.532
## 19   -0.252
## 20    0.542
## 21    0.180
```

```
## get Var(epsilon) & Var(eta)
(var_re_site <- as.data.frame(VarCorr(mod_re_popn_beta)))
```

```
## variance of random effects
sigma2_eta <- var_re_site$vcov[1]
## variance of residuals
sigma2_epsilon <- var_re_site$vcov[2]
```

```
##        grp     var1 var2      vcov     sdcor
## 1    pop_n spawners <NA> 0.3082175 0.5551734
## 2 Residual     <NA> <NA> 1.1137233 1.0553309
```

The estimate of σ^2_ϵ is 1.11 and the estimate of σ^2_η is 0.31.

```
## R^2
SSE <- sum(residuals(mod_re_popn_beta)^2)
(R2 <- 1 - SSE / SSTO)
```

```
## [1] 0.1364586
```

The R^2 value for this model is only ~0.136, which is much better than that for (c), but still quite low.

---

e) Fit the following model and report your estimates for α, each of the δ_i, β, and each of the η_i. Also report your estimate of σ^2_ϵ, σ^2_δ, and σ^2_η. Based on the R^2 value, how does this model compare to that from part (d)? (Hint: Refer back to the beginning of Lab 6 for how to fit uncorrelated random effects for both intercept and slope.)

$$\begin{aligned}\log(R_{i,t}/S_{i,t}) &= (\alpha + \delta_i) - (\beta + \eta_i)S_{i,t} + \epsilon_{i,t} \\ \delta_i &\sim \text{N}(0, \sigma^2_\delta) \\ \eta_i &\sim \text{N}(0, \sigma^2_\eta) \\ \epsilon_{i,t} &\sim \text{N}(0, \sigma^2_\epsilon)\end{aligned}$$

Here you want *uncorrelated* random effects for both the intercept and slope, which means you need to specify the random effects as `(1 + spawners || pop_n)`.

```
## RE for beta
mod_re_popn_both <- lmer(logRS ~ 1 + spawners + (1 + spawners || pop_n), data = psc)
summary(mod_re_popn_both)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: logRS ~ 1 + spawners + ((1 | pop_n) + (0 + spawners | pop_n))
##    Data: psc
##
## REML criterion at convergence: 1281.2
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -6.3703 -0.4275  0.0468  0.5303  3.7363
##
## Random effects:
```

```
##  Groups   Name        Variance Std.Dev.
##  pop_n    (Intercept) 0.11093  0.3331
##  pop_n.1  spawners    0.02088  0.1445
##  Residual             1.12927  1.0627
## Number of obs: 420, groups:  pop_n, 21
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.82360    0.10948   7.523
## spawners    -0.23983    0.06569  -3.651
##
## Correlation of Fixed Effects:
##          (Intr)
## spawners -0.459
```

The estimate of α is 0.82 and β is -0.24. The estimates of the δ_i and η_i are

```
REs <- round(ranef(mod_re_popn_both)$pop_n, 3)
colnames(REs) <- c("delta", "eta")
REs
```

```
##     delta    eta
## 1  -0.083 -0.019
## 2  -0.264 -0.106
## 3  -0.284 -0.046
## 4   0.151  0.164
## 5  -0.161 -0.055
## 6   0.104 -0.055
## 7   0.185  0.050
## 8   0.096 -0.064
## 9  -0.012 -0.020
## 10  0.289  0.052
## 11  0.270  0.017
## 12 -0.497 -0.064
## 13 -0.181 -0.027
## 14 -0.251 -0.012
## 15 -0.020 -0.017
## 16  0.198  0.103
## 17  0.427  0.023
## 18 -0.273 -0.046
## 19 -0.200 -0.021
## 20  0.312  0.191
## 21  0.191 -0.049
```

```
## get Var(epsilon) & Var(delta)
(var_re_site <- as.data.frame(VarCorr(mod_re_popn_both)))
## variance of random effects
sigma2_delta <- var_re_site$vcov[1]
```

```
sigma2_eta <- var_re_site$vcov[2]
## variance of residuals
sigma2_epsilon <- var_re_site$vcov[3]
```

```
##        grp        var1 var2       vcov     sdcor
## 1    pop_n (Intercept) <NA> 0.11093034 0.3330621
## 2  pop_n.1    spawners <NA> 0.02088231 0.1445071
## 3 Residual        <NA> <NA> 1.12927263 1.0626724
```

The estimate of σ_ϵ^2 is 1.13, the estimate of σ_δ^2 is 0.11, and the estimate of σ_η^2 is 0.02

```
## R^2
SSE <- sum(residuals(mod_re_popn_both)^2)
(R2 <- 1 - SSE / SSTO)
```

```
## [1] 0.1282285
```

The R^2 value for this model is only ~0.128, which is slightly worse than that for (c).

---

f) Based on the 3 models you fit in parts (c - e), test whether or not there is data support for including a random effect for population-level intercepts. Also test whether or not there is data support for including a random effect for population-level slopes. Make sure to specify your null hypothesis for both of the tests.

To compare our models with a single random effect, we need to compare them against a full model with both random effects. To do so, we need 3 different models:

1) model with single RE of interest to be tested (`model_A`)
2) full model with 2+ RE's (`model_AB`)
3) full model minus the RE in model (1) (`model_B`)

To conduct the test we use `extractRLRT(model_A, model_AB, model_B)`.

```
## load RLRsim package
library(RLRsim)

## test RE for intercept
exactRLRT(mod_re_popn_alpha, mod_re_popn_both, mod_re_popn_beta)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 7.3102, p-value = 0.003
```

We can reject $H_0 : \sigma_\delta^2 = 0$ and conclude that there is support for inclusion of a population-level offset to the intercept.

Here is the test for the population-level offset to the slope.

```
## test RE for slope
exactRLRT(mod_re_popn_beta, mod_re_popn_both, mod_re_popn_alpha)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 4.6473, p-value = 0.0089
```

Here, too, we can reject $H_0 : \sigma^2_\eta = 0$ and conclude that there is support for inclusion of the `year` random effect.

g) Now fit the following model and report your estimates for α, each of the δ_i, β, each of the η_i, and each of the γ_t. Also report your estimate of σ^2_ϵ, σ^2_δ, σ^2_γ, and σ^2_η. Based on the R^2 value, how does this model compare to that from part (d)?

$$\begin{aligned}
\log(R_{i,t}/S_{i,t}) &= (\alpha + \delta_i + \gamma_t) - (\beta + \eta_i)S_{i,t} + \epsilon_{i,t} \\
\delta_i &\sim \text{N}(0, \sigma^2_\delta) \\
\gamma_t &\sim \text{N}(0, \sigma^2_\gamma) \\
\eta_i &\sim \text{N}(0, \sigma^2_\eta) \\
\epsilon_{i,t} &\sim \text{N}(0, \sigma^2_\epsilon)
\end{aligned}$$

Here you want *uncorrelated* random effects for both the intercept and slope, plus a random effect for year, which means you need to specify the random effects as `(1 + spawners || pop_n) + (1 | year)`.

```
## RE for beta
mod_re_popn_3 <- lmer(logRS ~ 1 + spawners + (1 + spawners || pop_n)  + (1 | year), data = psc
summary(mod_re_popn_3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: logRS ~ 1 + spawners + ((1 | pop_n) + (0 + spawners | pop_n)) +
##     (1 | year)
##    Data: psc
##
## REML criterion at convergence: 1264
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -6.8465 -0.3849  0.0678  0.5017  3.8513
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  pop_n    (Intercept) 0.10359  0.3219
##  pop_n.1  spawners    0.01488  0.1220
##  year     (Intercept) 0.10720  0.3274
##  Residual             1.03112  1.0154
## Number of obs: 420, groups:  pop_n, 21; year, 20
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  0.79085    0.12855   6.152
## spawners    -0.20182    0.06002  -3.363
##
## Correlation of Fixed Effects:
##          (Intr)
## spawners -0.392
```

The estimate of α is 0.82 and β is -0.24. The estimates of the δ_i (`pop_n$(Intercept)`), η_i (`pop_n$spawners`), and γ_t (`year$(Intercept)`) are

```
# round(ranef(mod_re_popn_3), 3)
lapply(ranef(mod_re_popn_3), round, 3)
```

```
## $pop_n
##    (Intercept) spawners
## 1       -0.070   -0.010
## 2       -0.268   -0.079
## 3       -0.275   -0.026
## 4        0.177    0.126
## 5       -0.156   -0.074
## 6        0.100   -0.036
## 7        0.178    0.033
## 8        0.072   -0.053
## 9       -0.015   -0.019
## 10       0.288    0.046
## 11       0.254    0.009
## 12      -0.489   -0.048
## 13      -0.169   -0.017
## 14      -0.238   -0.011
## 15      -0.035   -0.020
## 16       0.203    0.073
## 17       0.414    0.022
## 18      -0.268   -0.035
## 19      -0.192   -0.018
## 20       0.308    0.158
## 21       0.181   -0.021
##
## $year
##      (Intercept)
## 1986       0.207
## 1987       0.090
## 1988       0.328
## 1989      -0.013
## 1990      -0.226
## 1991      -0.123
## 1992       0.031
## 1993      -0.026
## 1994       0.095
## 1995      -0.168
## 1996       0.165
## 1997       0.490
## 1998       0.214
## 1999       0.041
## 2000       0.258
## 2001      -0.617
## 2002       0.036
## 2003      -0.357
## 2004       0.060
## 2005      -0.484
```

```
## get Var(epsilon) & Var(delta)
(var_re_site <- as.data.frame(VarCorr(mod_re_popn_3)))
## variance of random effects
sigma2_delta <- var_re_site$vcov[1]
sigma2_eta <- var_re_site$vcov[2]
sigma2_gamma <- var_re_site$vcov[3]
## variance of residuals
sigma2_epsilon <- var_re_site$vcov[4]
```

```
##        grp        var1 var2       vcov     sdcor
## 1    pop_n (Intercept) <NA> 0.10358789 0.3218507
## 2  pop_n.1    spawners <NA> 0.01488145 0.1219895
## 3     year (Intercept) <NA> 0.10720197 0.3274171
## 4 Residual        <NA> <NA> 1.03112332 1.0154424
```

The estimate of σ^2_ϵ is 1.03, the estimate of σ^2_δ is 0.1, the estimate of σ^2_η is 0.015, and the estimate of σ^2_γ is 0.11.

```
## R^2
SSE <- sum(residuals(mod_re_popn_3)^2)
(R2 <- 1 - SSE / SSTO)
```

```
## [1] 0.2290055
```

The R^2 value for this model is ~0.229, which is our best yet.

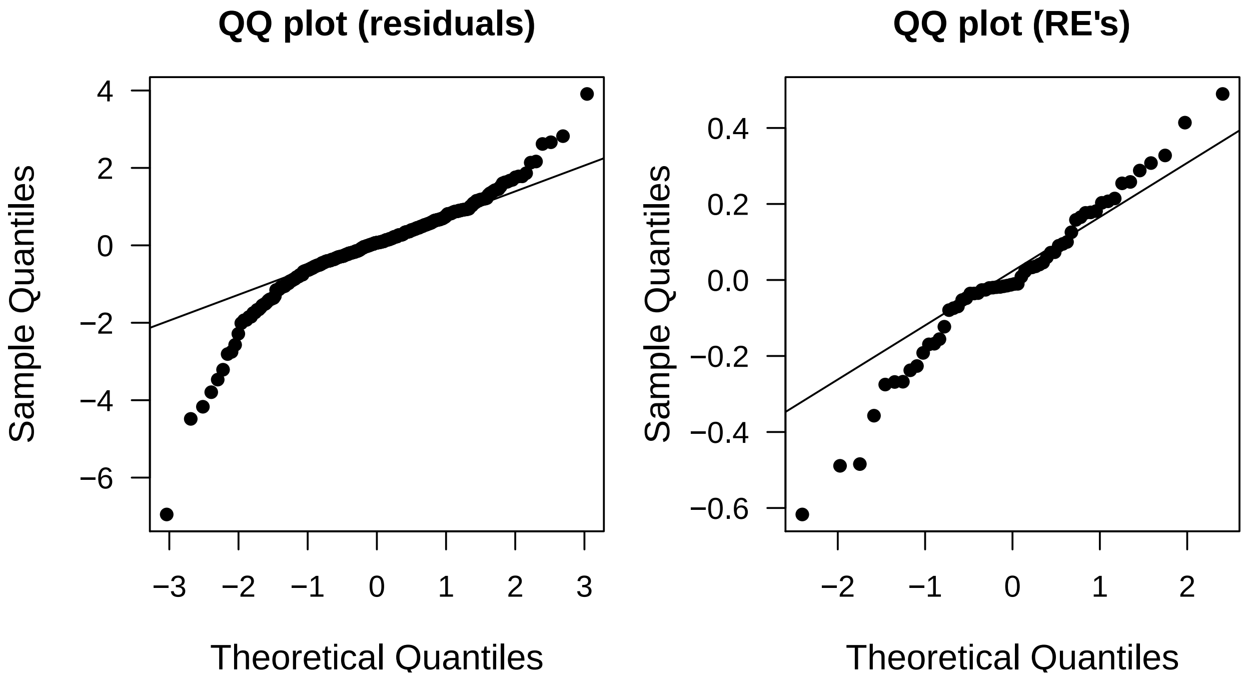
---

h) Conduct a diagnostic check of the model you fit in (g) to evaluate the adequacy of the model assumptions. Do you see any cause for concern?

We should be checking a *Q-Q* plot, a plot of the residuals versus the fitted values, and the degree of autocorrelation in the residuals for each population.

Q-Q plots

```
## set plot area
par(mai = c(0.9, 0.9, 0.6, 0.1),
    omi = c(0, 0, 0, 0),
    mfrow = c(1,2), cex.lab = 1.2)

## qq resids
qqnorm(residuals(mod_re_popn_3), main = "QQ plot (residuals)", las = 1, pch = 16)
qqline(residuals(mod_re_popn_3))

## qq RE's
qqnorm(unlist(ranef(mod_re_popn_3)), main = "QQ plot (RE's)", las = 1, pch = 16)
qqline(unlist(ranef(mod_re_popn_3)))
```

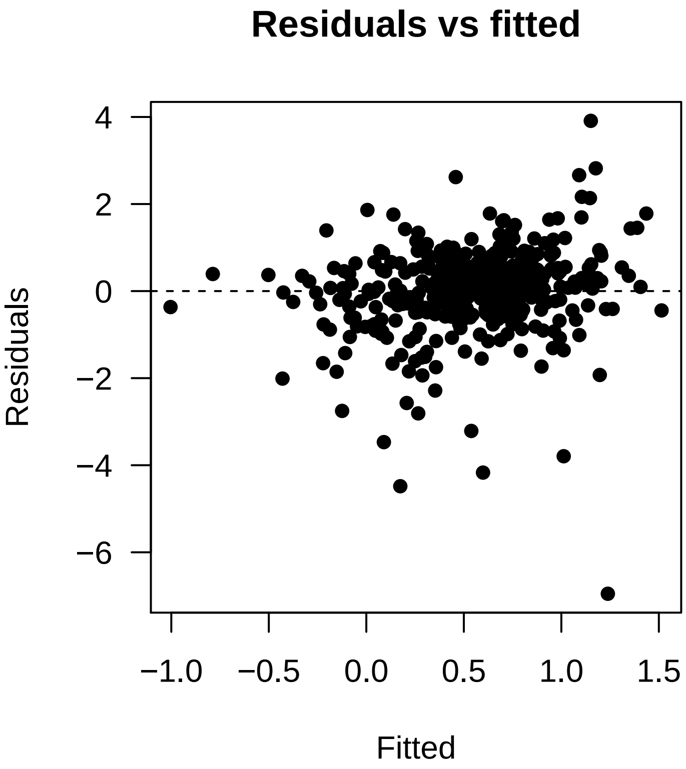


These plots indicate some leptokurtosis (heavy tails) in the residuals, suggesting our model assumptions are somewhat questionable.

Residuals versus fitted

We can also plot the model residuals against the fitted values to look for evidence of heteroscedasticity or non-linearity in the residuals.

```
## resids vs fitted
plot(fitted(mod_re_popn_3), residuals(mod_re_popn_3), las = 1, pch = 16,
     xlab = "Fitted", ylab = "Residuals",
     main = "Residuals vs fitted")
abline(h=0, lty = "dashed")
```



This residual plot looks pretty good with the exception of one outlier in the lower right.

Autocorrelation

Because these data were collected over time, we should be aware of possible autocorrelation among the residuals. It would be a bit messy to create plots for all 9 of the time series, so we'll just get a table of the results from `acf()` and see whether any of them exceed the critical value given by

$$0 \pm \frac{z_{\alpha/2}}{\sqrt{n}}$$

where $z_{\alpha/2}$ is the $1 - \alpha/2$ quantile of the standard normal distribution. For example, if $\alpha = 0.05$, then $z_{\alpha/2} = 1.96$. Here we'll only examine correlations out to a lag of 5 years because it's unlikely that counts in this year would be related to counts 6 or more years in the past (and hopefully not at any years in the past).

```
## Type-I error
alpha_crit <- 0.05

## threshold value for rho (correlation)
(rho_crit <- qnorm(1 - alpha_crit/2) / sqrt(n_yrs))
```

```
## [1] 0.4382613
```

```
## rearrange residuals into matrix
rr <- matrix(residuals(mod_re_popn_3), n_yrs, n_pops)
```

```
## get ACF
ACF <- apply(rr, 2, acf, lag.max = 5, plot = FALSE)
ACF <- lapply(ACF, function(x) x$acf)
## convert list to matrix; don't need row 1 b/c rho_0 = 1
ACF <- do.call(cbind, ACF)[-1,]

## check if any values > rho_crit by popn
bad_rho <- apply(ACF, 2, function(x) abs(x) > rho_crit)
apply(bad_rho, 2, any)
```

```
##  [1] FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE
## [12]  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
```

It looks like the random effect for year γ_t did *not* do an adequate job of accounting for all of the autocorrelation in the data. However, there are *a lot* of null hypothesis tests here, so some of the correlations should exceed the critical value by chance alone.