

# Compare zero-inflated mixed models across R packages

*Mollie Brooks and Ben Bolker*

*2016-11-24*

## Preliminaries

### Load packages

```
library(glmTMB)
library(glmADMB)
library(MCMCglmm)
library(brms)
library(broom) #for tidy
library(plyr)
library(ggplot2); theme_set(theme_bw())
library(ggstance) #for position_dodgev
```

### Data organization and helper functions

```
data(Owls)
Owls = rename(Owls, c(SiblingNegotiation="NCalls"))
Owls = transform(Owls, ArrivalTime=scale(ArrivalTime, center=TRUE, scale=FALSE))

## time
tfun <- function(...) unname(system.time(capture.output(...))["elapsed"])
```

## Fitting the same model in 4 packages

### glmTMB

```
form <- NCalls~(FoodTreatment + ArrivalTime) * SexParent +
  offset(logBroodSize) + (1|Nest)
time.tmb = tfun(m1.tmb <- glmTMB(form,
  ziformula=~1, data = Owls, family="poisson"))
```

### glmADMB

```
time.admb = tfun(m1.admb <- glmadmb(form,
  zeroInflation=TRUE, data = Owls, family="poisson"))
```

## MCMCglmm

```
offvec = c(1,1,2,rep(1,5))
fixef2 = NCalls~trait-1+ ## intercept terms for both count and binary terms
  at.level(trait,1):logBroodSize+
  at.level(trait,1):((FoodTreatment+ArrivalTime)*SexParent)
prior_overdisp = list(R=list(V=diag(c(1,1)),nu=0.002,fix=2),
  G=list(list(V=diag(c(1,1e-6)),nu=0.002,fix=2)))

prior_overdisp_broodoff = c(prior_overdisp,
  list(B=list(mu=c(0,1)[offvec],
    V=diag(c(1e8,1e-6)[offvec]))))

time.mcmc=tfun(m1.mcmc <- MCMCglmm(fixef2,
  rcov=~idh(trait):units,
  random=~idh(trait):Nest,
  prior=prior_overdisp_broodoff,
  data=Owls,
  family="zipoisson",
  verbose=FALSE))
```

## brms

```
time.brms = tfun(m1.brms <- brm(form, data = Owls,
  family="zero_inflated_poisson",
  save_dso=TRUE))

## Compiling the C++ model
time.brms2 = tfun(m1.brms2 <- update(m1.brms))
```

## Comparing the results

### Timings

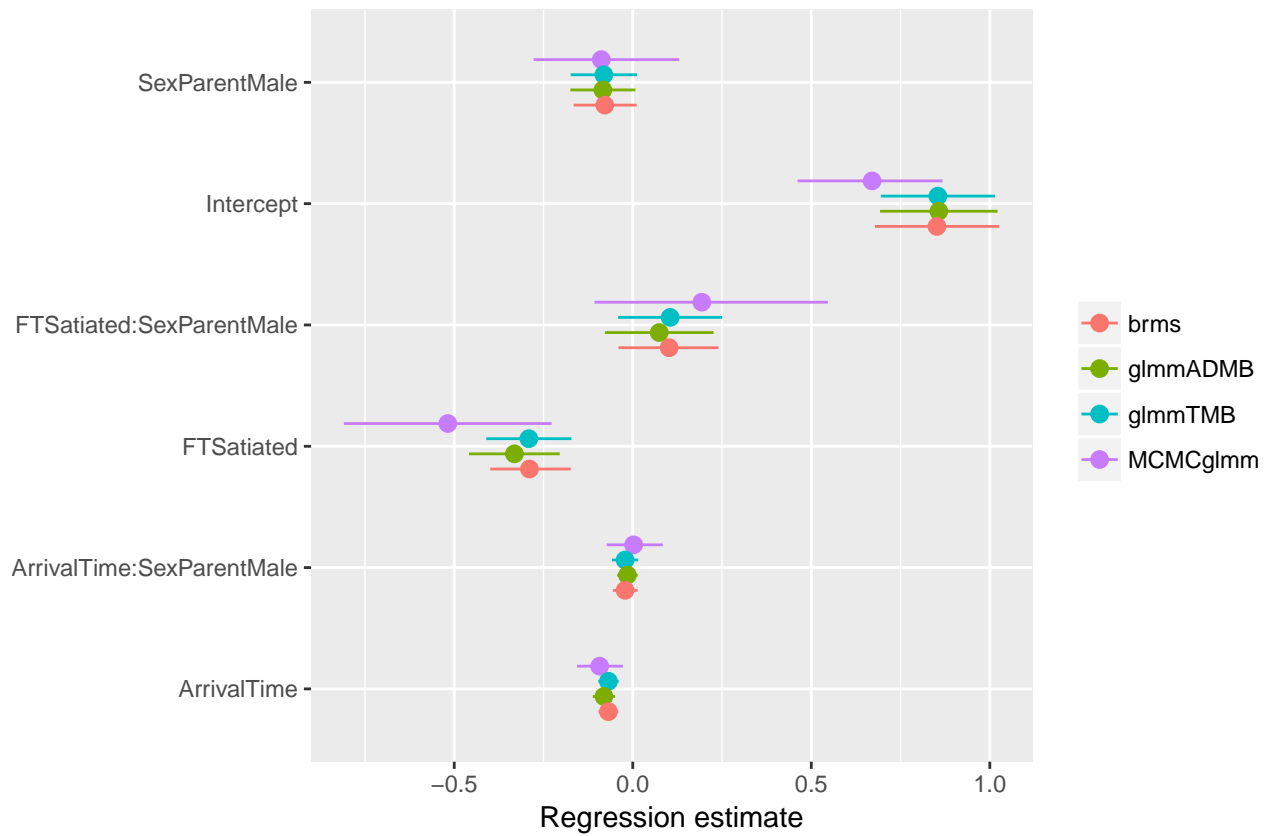
```
sort(c(TMB=time.tmb,ADMB=time.admb,MCMCglmm=time.mcmc,brms=time.brms,
  brms2=time.brms2))
```

##	TMB	MCMCglmm	ADMB	brms2	brms
##	5.429	22.165	42.851	77.629	145.151

(Time is recorded in seconds.)

glmmTMB fit the model in less than 5 seconds. Other methods were slower, but MCMCglmm was in the same order of magnitude (brms and brms2 are times including and excluding compilation time, respectively).

## Estimated fixed-effect coefficients



Because we ran `brms` with flat priors, the estimates are very close to the maximum likelihood estimates of `glmmTMB`. Maximum likelihood estimates from `glmmTMB` and `glmmADMB` differ slightly because `glmmADMB` uses some numerical tricks to increase robustness and these change the objective function by a small amount.