DHARMa for mixed model diagnostics

MATH 6642

2025-03-24

DHARMa for mixed model diagnostics

When we use single-level normal regression models, we have a theoretical basis for the expected shape of the residuals and our diagnostic algorithms can use that theory to create expected quantiles for residuals.

We can plot the actual residuals against the 'expected residuals' as a test of the validity of the model and to get insights into possible changes in the model that would address anomalies in the shape of residuals.

For example, if an important factor has been omitted from a regression, we might get something like this:

```
{
    # Pretend real-world simulation
    set.seed(284788712)
    N <- 1000
    dd <- within(
        data.frame(obs = 1:N, x = 10 + 2*rnorm(N)),
        {
            ..eps <- rnorm(N)
            z <- 1*(rank(x) > N-N/20)  # top 5% of x's
            y <- 2 + x + 5 * z + ..eps
     }
}</pre>
```

```
)
# When you put a chunk of code in braces you can
# run it by pressing Ctrl-Enter when your cursor
# is at the closing brace.
# This is very handy if it's a chunk of code that
# you are likely to change and rerun many times.
}
```

Suppose we don't think of including z:

```
fit <- lm(y ~ x, dd)
plot(fit, which = 1)  # residual vs yhat</pre>
```

Residuals vs Fitted



plot(fit, which = 2) # Normal quantile plot



plot(fit, which = 5) # residual vs leverage plot



The QQ plot plots residuals against 'ideal residuals' with a 'perfect' normal distribution, i.e. spaced apart with $\frac{1}{n+1}$ of the probability between each value.

```
For n = 10:
ł
  n <- 10
  ddens <- within(
    data.frame(z = seq(-3.5, 3.5, .01)),
    ł
      density <- dnorm(z)
  dqs <- within(</pre>
    expand.grid(ytype = 1:3, os = 1:n),
    ſ
      p1 <- os / (n + 1)
      p2 \leq rep(cumsum(c(1/(2*n), rep(1/n, n-1), 1/(2*n))), each = 3)
```

```
z1 \leq qnorm(p1)
    z2 \leq qnorm(p2)
    y1 <- ifelse(ytype == 1, 0,</pre>
                  ifelse(ytype == 2, dnorm(z1), NA)
    y2 \leftarrow ifelse(ytype == 1, 0,
                  ifelse(ytype == 2, dnorm(z2), NA)
xyplot(density ~ z, ddens, type = 'l', lwd = 2,
       scales = list(y = list(axs = 'i')),
       xlab = TeX(paste(n, 'standard normal quantiles'))) +
 xyplot(y1 ~ z1, dqs, type = 'l', lwd =2, col = 'red') +
```

```
## Warning in `[<-.data.frame`(`*tmp*`, nl, value = list(y2 = c(0,
## 0.103135640375371, : replacement element 3 has 33 rows to replace 30 rows
```

}

Warning in `[<-.data.frame`(`*tmp*`, nl, value = list(y2 = c(0, ## 0.103135640375371, : replacement element 5 has 33 rows to replace 30 rows



Sample quantiles (observed residuals) plotted against theoretical quantiles:
qqnorm(resid(fit))

Normal Q-Q Plot



Theoretical Quantiles

xyplot(resid(fit) ~ z, dd)



z

```
xyplot(resid(fit) ~ jitter(z), dd)
```



fit2 <- $lm(y \sim x + z, dd)$ plot(fit2, which = 2)







t Quantiles

[1] 228 504

car::qqPlot(fit)



t Quantiles

[1] 330 630

With mixed models, there are residuals from the fixed effects model, referred to as 'marginal' residuals and BLUPs, which are residuals from fitted values based on both the fixed effects and the random effects, referred to as 'conditional' residuals.

These residuals have a complex distribution so there isn't an easy theoretical basis to compute their expected values. The "DHARMa" package uses the fitted model to simulate residuals, whose distribution can then be compared with the distribution of the fitted residuals. If the model fits the data well, the distribution of the simulated residuals should resemble the distribution of the fitted residuals.

There is a **recent vignette** on the package, (Hartig 2022).

The "DHARMa" package cannot be used with models that use "nlme". It is generally used with two packages that are similar to "nlme": the "lme4" package and the "glmmTMB" package.

Both of these packages allow the use of 'Generalized Linear Mixed Models' for responses whose distributions fall in the class of distributions for GLMs: Bernoulli, binomial, Poisson, etc.

The "lme4" packages only allows G-side modelling. The "glmmTMB" package allows limited R-side modelling (e.g. AR1 but not more complex ARMA). It also supports separated models for zero-inflated Poisson and binomial distributions, as well as models for overdispersion. The "glmmTMB" package is actively developed. A good source of information is the vignettes that come with the package. Install it and use the command: help(p=glmmTMB).

References

Hartig, Florian. 2022. DHARMa: Residual Diagnostics for Hierarchical (Multi-Level / Mixed) Regression Models. Manual. https://CRAN.R-project.org/package=DHARMa.