R package gllvm

Jenni Niku, University of Jyväskylä [jenni.m.e.niku@jyu.fi]

12/17/2020

R package gllvm

- **R** package **gllvm** fits Generalized linear latent variable models (GLLVM) for multivariate data (Niku et al., 2017).
- Developed by J. Niku, W.Brooks, R. Herliansyah, F.K.C. Hui, S. Taskinen, D.I. Warton, B. van der Veen.
- GitHub: https://github.com/JenniNiku/gllvm
- CRAN: https://cran.r-project.org/web/packages/gllvm/index.html

R package gllvm

- GLLVMs are computationally intensive to fit.
- Often such models have been fitted using MCMC approach, which is very time consuming.
- **gllvm** package overcomes computational problems by applying closed form approximations to log-likelihood and using automatic differentiation in C++ to accelerate computation times (**TMB**).
- Estimation is performed using either variational approximation (VA) or Laplace approximation (LA) method implemented via **R** package **TMB**.
- VA method is faster and more accurate than LA, but not applicable for all distributions and link functions.

R package gllvm

Using **gllvm** we can fit

- GLLVM without covariates gives model-based ordination and biplots
- GLLVM with environmental covariates for studying factors explaining species abundance
- Fourth corner models with latent variables for studying environmental-trait interactions
- GLLVM without latent variables fits basic multivariate GLMs

Additional tools: model checking, model selection, inference, visualization.

Distributions

Response	Distribution	Method	Link
Counts	Poisson	VA/LA	log
	NB	VA/LA	log
	ZIP	LA	log
Binary	Bernoulli	VA/LA	probit
		LA	logit
Ordinal	Ordinal	VA	probit
Normal	Gaussian	VA/LA	identity
Positive continuous	Gamma	VA/LA	log
non-negative continuous	Exponential	VA/LA	log
Biomass	Tweedie	LA	log

Data input

Main function of the **gllvm** package is `gllvm()`, which can be used to fit GLLVMs for multivariate data with the most important arguments listed in the following:

```
gllvm(y = NULL, X = NULL, TR = NULL, family, num.lv = 2,
 formula = NULL, method = "VA", row.eff = FALSE, n.init=1, ...)
```

- y: matrix of abundances
- X: matrix or data.frame of environmental variables
- TR: matrix or data.frame of trait variables
- family: distribution for responses
- num.lv: number of latent variables
- method: approximation used “VA” or “LA”
- row.eff: type of row effects
- n.init: number of random starting points for latent variables

```
## Loading required package: TMB
## Loading required package: mvabund
```

Example: Spiders

- Abundances of 12 hunting spider species measured as a count at 28 sites.
- Six environmental variables measured at each site:
  - `soil.dry`: Soil dry mass
  - `bare.sand`: cover of bare sand
  - `fallen.leaves`: cover of fallen leaves/twigs
  - `moss`: cover of moss
  - `herb.layer`: cover of herb layer
  - `reflection`: reflection of the soil surface with a cloudless sky

Data fitting

Fit GLLVM with environmental variables $g(E(y_{ij})) = \beta_{0j} + \boldsymbol{x}_i'\boldsymbol{\beta}_j + \boldsymbol{u}_i'\boldsymbol{\theta}_j$ using **gllvm**:

```
library(gllvm)
data("spider")
fitx <- gllvm(y = spider$abund, X=spider$x, family = "negative.binomial")
fitx
## Call:
## gllvm(y = spider$abund, X = spider$x, family = "negative.binomial")
## family:
## [1] "negative.binomial"
## method:
## [1] "VA"
##
## log-likelihood:  -593.6748
## Residual degrees of freedom:  217
## AIC:  1425.35
## AICc:  1114.915
## BIC:  1583.882
```

Model selection

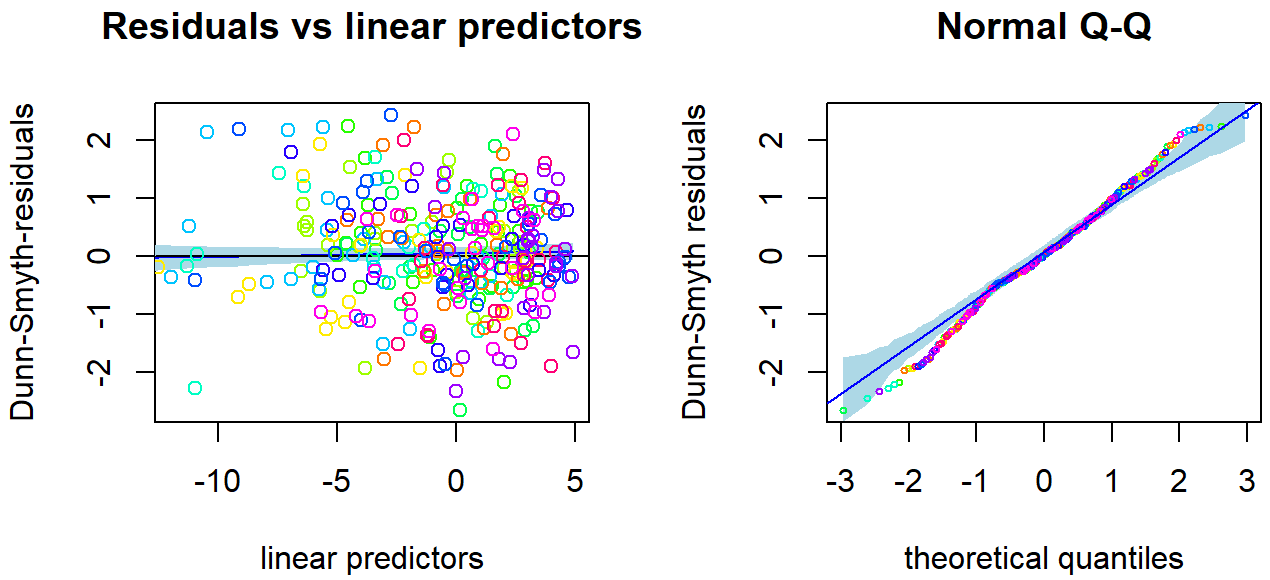
- Number of latent variables is not necessarily clear beforehand when goal is not primarily in ordination, so information criterias can be used for model selection. For example, using Akaike information criterion:

```
X=spider$x
fitx1 <- gllvm(spider$abund, X, family = "negative.binomial", num.lv = 1)
fitx2 <- gllvm(spider$abund, X, family = "negative.binomial", num.lv = 2)
fitx3 <- gllvm(spider$abund, X, family = "negative.binomial", num.lv = 3)
AIC(fitx1)
## [1] 1393.837
AIC(fitx2)
## [1] 1425.35
AIC(fitx3)
## [1] 1445.35
```

Residual analysis

- Residual analysis can be used to assess the appropriateness of the fitted model (eg. in terms of mean-variance relationship). Can be performed using `plot()`:

```
par(mfrow = c(1,2))
plot(fitx1, which = 1:2)
```



Exercises

For the next 10 minutes, try the package yourself:

1. Load spider data from **mvabund** package and take a look at the dataset.
2. Fit GLLVM with two latent variables to the spider data with a suitable distribution.
3. Fit GLLVM with environmental variables `soil.dry` and `reflection` to the data with suitable number of latent variables.
4. Explore the model fit. Find the coefficients for environmental covariates.

R code for the exercises is in Github repository:
https://github.com/BertvanderVeen/BES2020GLLVMworkshop

If you need help, here you can find the exercises with answers and tips:
https://jenniniku.github.io/gllvm/articles/vignette4.html

Studying species correlations

- Latent variables induce correlation across response variables, and so provide means of estimating correlation patterns across species, and the extent to which they can be explained by environmental variables.
- Information on correlation is stored in the LV loadings $\boldsymbol{\theta}_j$, so the residual covariance matrix, storing information on species co-occurrence that is not explained by environmental variables, can be calculated as $\boldsymbol{\Sigma} = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^\top$, where $\boldsymbol{\Gamma} = [\boldsymbol{\theta}_1 \dots \boldsymbol{\theta}_m]'$.
- `getResidualCor` function can be used to estimate the correlation matrix of the linear predictor across species.

Studying species correlations

- Let's consider first the correlation matrix based on a model without predictors: $g(E(y_{ij})) = \beta_{0j} + \boldsymbol{u}_i'\boldsymbol{\theta}_j$

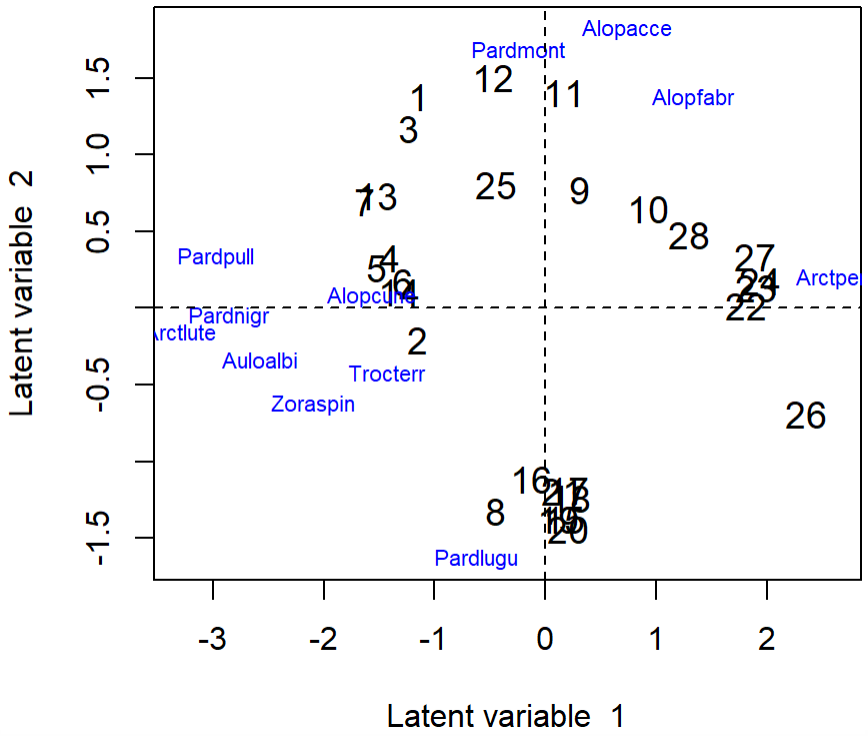
```
fitnb <- gllvm(spider$abund, family = "negative.binomial")
```

- The obtained correlation matrix then does not take into account the environmental conditions driving species abundances at sites, and reflects only what has been observed.
- The residual correlations can be visualized either using biplot, which shows the species ordination, or visualising the actual correlation matrix using, eg., a `corrplot` package.

Biplot

The biplot can be produced using a function `ordiplot()` with an argument `biplot = TRUE`:

```
ordiplot(fitnb, biplot = TRUE)
abline(h = 0, v = 0, lty=2)
```



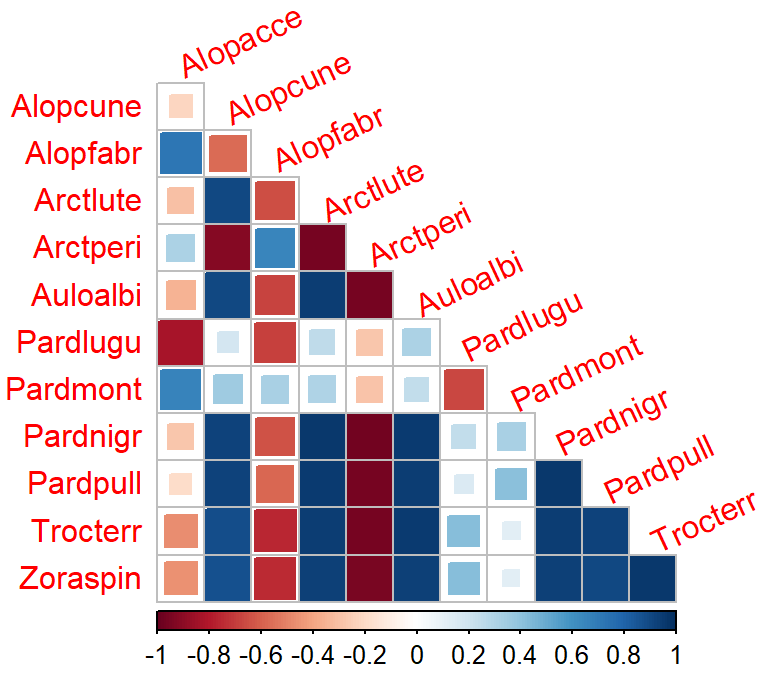
Correlation matrix

Correlations can be visualised more precisely using `corrplot()` function :

```
cr <- getResidualCor(fitnb)
library(corrplot);
## corrplot 0.84 loaded
```

```
corrplot(cr, diag = FALSE, type = "lower", method = "square", tl.srt = 25)
```

Correlation matrix

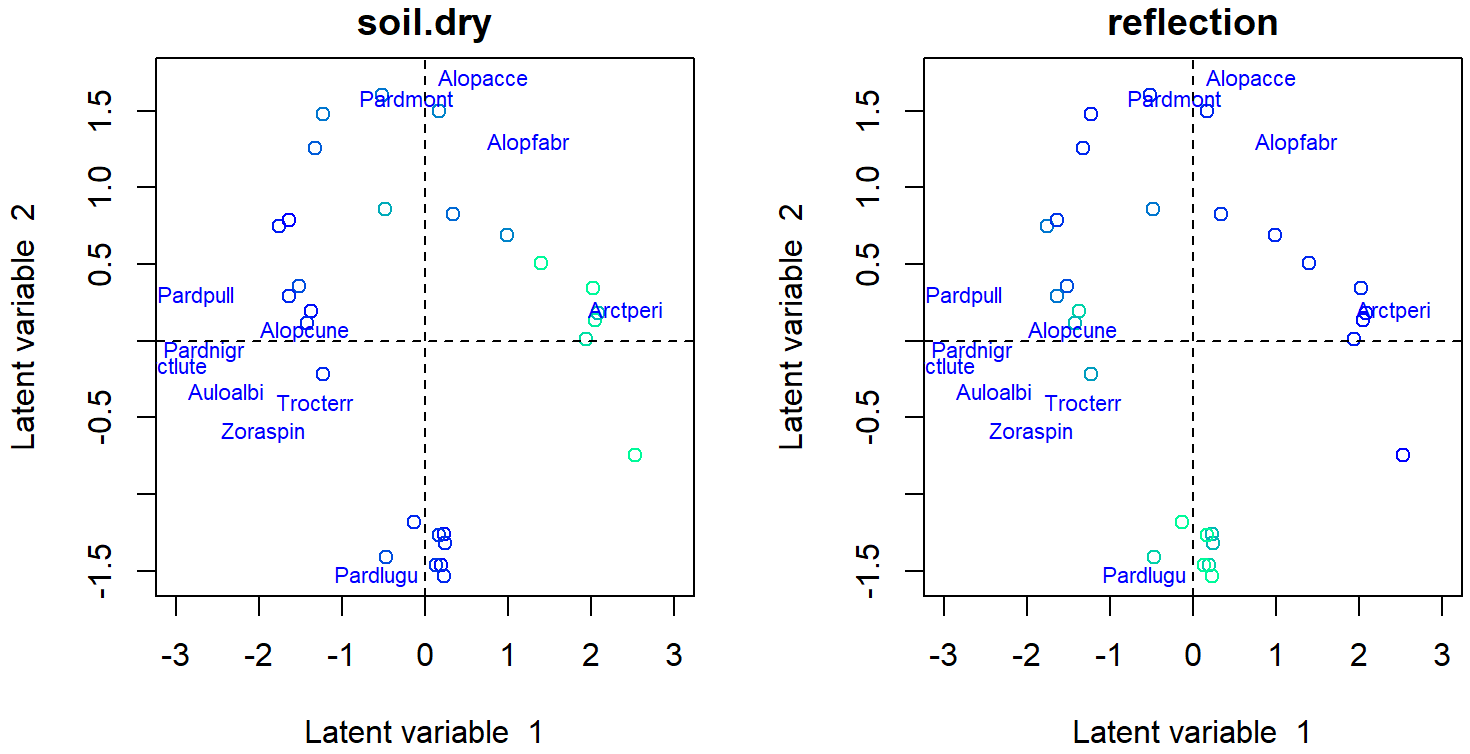


Studying effects of environmental variables

- The effects of environmental variables on species can be studied by including environmental variables $\boldsymbol{x}_i$ to GLLVM:
  $g(E(y_{ij})) = \beta_{0j} + \boldsymbol{x}_i'\boldsymbol{\beta}_j + \boldsymbol{u}_i'\boldsymbol{\theta}_j\,.$
- $\boldsymbol{\beta}_j$ is a vector of species specific coefficients for environmental variables.
- Next consider for example two environmental variables, `soil.dry` (soil dry mass) and `reflection` (reflection of the soil surface with a cloudless sky), which shows different environmental gradients in ordination:

```
# Color scale:
rbPal <- c("#00FA9A", "#00EC9F", "#00DFA4", "#00D2A9", "#00C5AF", "#00B8B4", "#00ABB9", ...
X <- spider$x[,c(1,6)]
for(i in 1:ncol(X)){
Col <- rbPal[as.numeric(cut(X[,i], breaks = 20))]
ordiplot(fitnb, symbols = T, s.colors = Col, main = colnames(X)[i], biplot = TRUE)
}
```

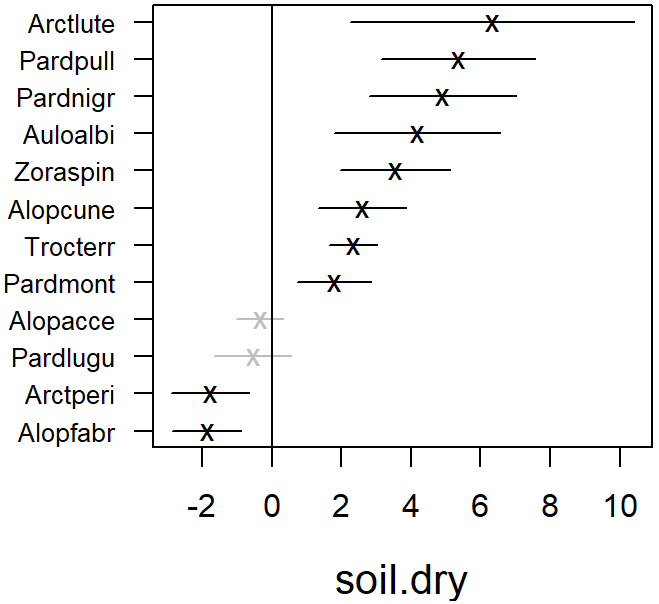
Environmental gradients

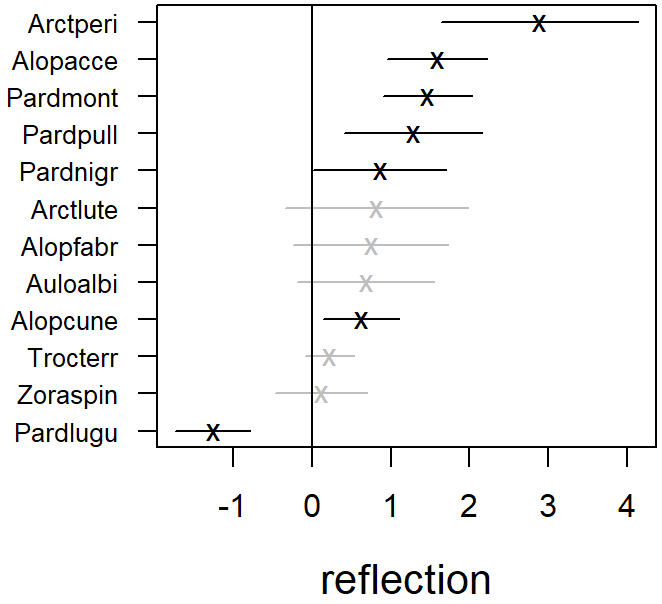


Coefficient plot

`coefplot()` plots point estimates of the species specific environmental coefficients β_j with confidence intervals.

```
fitx1 <- gllvm(spider$abund, X, family = "negative.binomial", num.lv = 1)
coefplot(fitx1, mfrow = c(1,2), cex.ylab = 0.8)
```

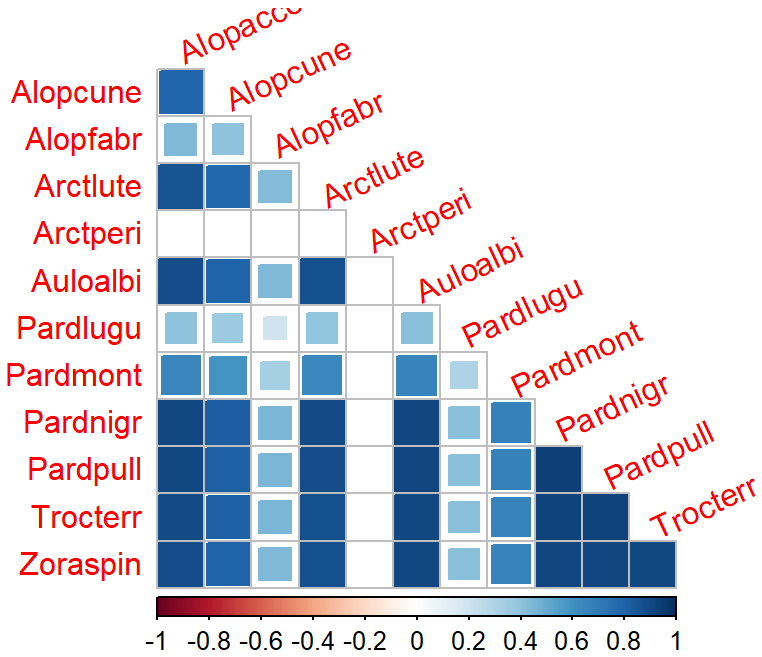




Correlation matrix

Correlation matrix for model with predictors shows correlation patterns between species when the effect of the predictors are taken into account.

```
crx <- getResidualCor(fitx1)
corrplot(crx, diag = FALSE, type = "lower", method = "square", tl.srt = 25)
```



Fourth corner models

- If species trait variables $\boldsymbol{t}_j$, measuring eg. species behaviour or physical appearance, would be available, fourth corner models should be considered: $g(E(\boldsymbol{y}_{ij})) = \beta_{0j} + \boldsymbol{x}_i'\boldsymbol{\beta}_j + \boldsymbol{x}_i'\boldsymbol{B}_I\boldsymbol{t}_j + \boldsymbol{u}_i'\boldsymbol{\theta}_j$
- Such models can also be fitted with `gllvm()` function by including a matrix of traits with argument `TR`.
- Examples can be found in the **gllvm** package's vignettes.

More

More information and examples about the usage of the package can be found from the **gllvm** package's website: https://jenniniku.github.io/gllvm/

Github repository of the workshop: https://github.com/BertvanderVeen/BES2020GLLVMworkshop

Break / Questions

- On twitter: #**GLLVMs**, @**vdVeenB** or @**J__Niku** or @**samperrinNTNU** or @**BobOHara**
- On github: https://github.com/BertvanderVeen/BES2020GLLVMworkshop/discussions