R package gllvm

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

12/16/2020

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 and/or trait 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: residuals, information criterias, confidence intervals, 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 without covariates $g(E(y_{ij})) = \beta_{0j} + \boldsymbol{u}_i'\boldsymbol{\theta}_j$ with **gllvm**:

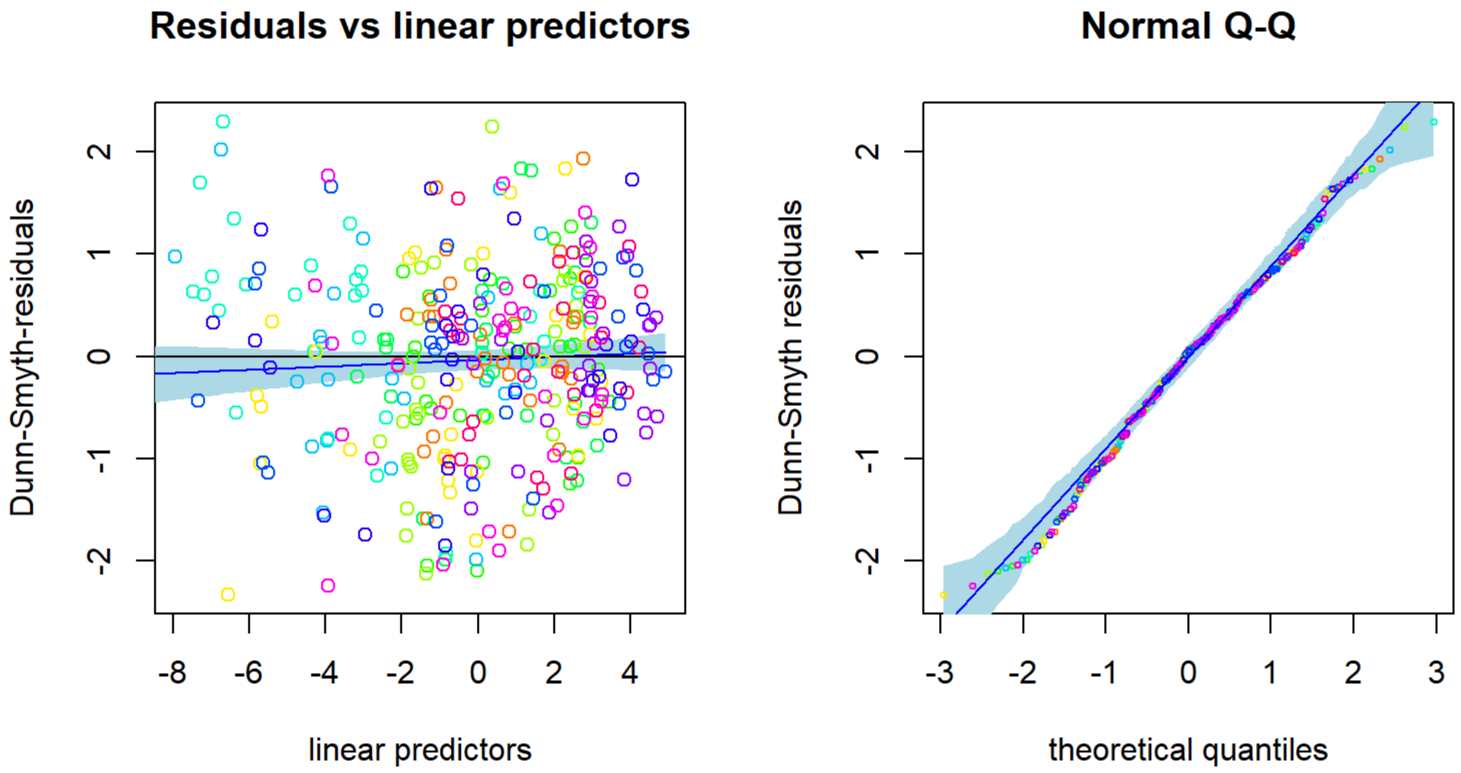
```
library(mvabund)
data("spider")
library(gllvm)
fitnb <- gllvm(y = spider$abund, family = "negative.binomial")
fitnb
## Call:
## gllvm(y = spider$abund, family = "negative.binomial")
## family:
## [1] "negative.binomial"
## method:
## [1] "VA"
##
## log-likelihood:  -733.6806
## Residual degrees of freedom:  289
## AIC:  1561.361
## AICc:  1335.761
## BIC:  1623.975
```

Residual analysis

- Residual analysis can be used to assess the appropriateness of the fitted model (eg. in terms of mean-variance relationship).
- Function `residuals()` calculates randomized quantile residuals for the model, and `plot()` function provides residual plots.
- Randomized quantile/Dunn-Smyth residuals are used in the package, as they provide standard normal distributed residuals, even for discrete responses, in the case of a proper model.

Residual analysis

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



Model selection

- Information criterias can be used for model selection.
- For example, compare distributions or choose suitable number of latent variables.

```
fitp <- gllvm(y = spider$abund, family = poisson())
fitnb <- gllvm(y = spider$abund, family = "negative.binomial")
AIC(fitp)
## [1] 1761.655
AIC(fitnb)
## [1] 1561.361
```

Exercises

1. Load spider data from **mvabund** package and take a look at the dataset.
2. Fit GLLVM to spider data with a suitable distribution.
3. Explore the fitted model. Where are the estimates for parameters? What about predicted latent variables? Standard errors?
4. Fit model with different numbers of latent variables.
5. Include environmental variables to the GLLVM and explore the model fit.

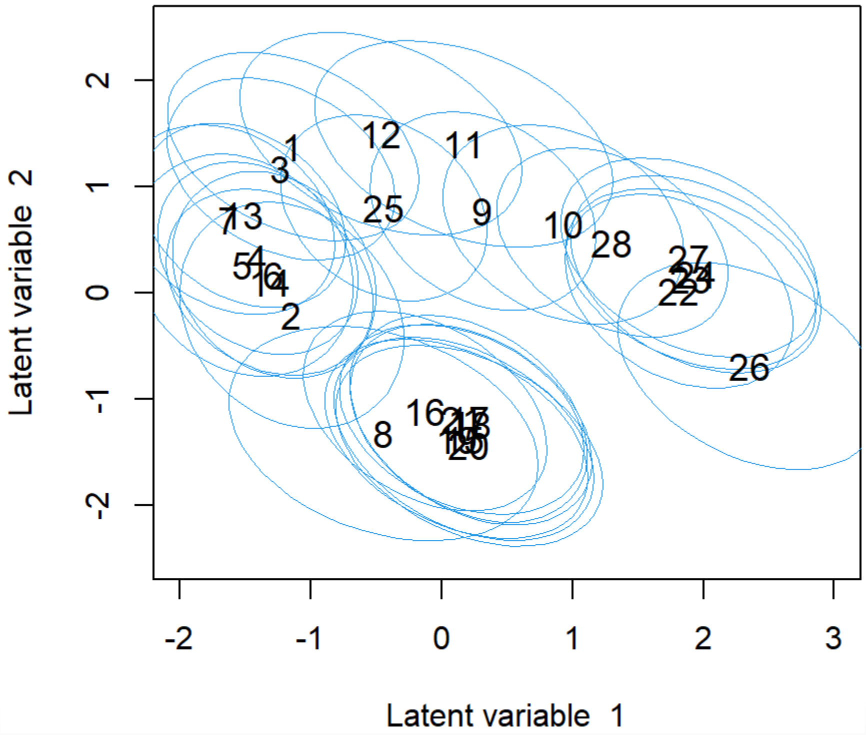
GLLVM as a model based ordination method

- GLLVMs can be used as a model-based approach to unconstrained ordination by including two latent variables in the model: $g(E(y_{ij})) = \beta_{0j} + \boldsymbol{u}_i'\boldsymbol{\theta}_j$
- Predictions for the two latent variables, $\hat{\boldsymbol{u}}_i = (\hat{u}_{i1}, \hat{u}_{i2})$, then provide coordinates for sites in the ordination plot and then provides a graphical representation of which sites are similar in terms of their species composition.
- The corresponding ordination plot then provides a graphical representation of which sites are similar in terms of their species composition.

Ordination plot

- `ordiplot()` produces ordination plots based on fitted GLLVMs.
- Uncertainty of the ordination points in model based ordination can be assessed with prediction errors of latent variables.

```
ordiplot(fitnb, predict.region = TRUE, ylim=c(-2.5,2.5), xlim=c(-2,3))
```

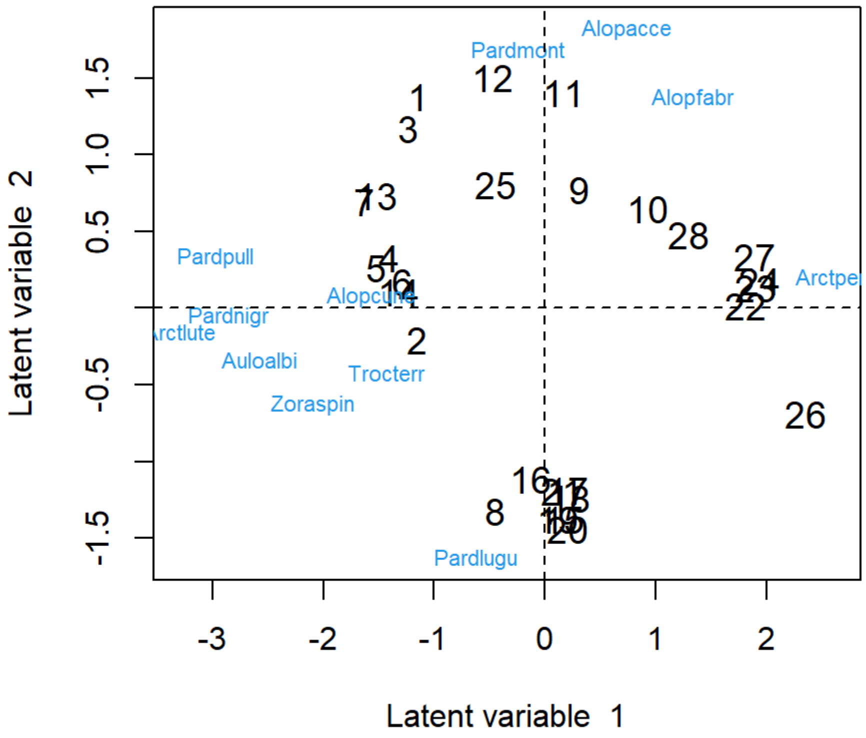


Biplot

- Between species correlations can be visualized with biplot by adding latent variable loadings $\boldsymbol{\theta}_j$ to the ordination of sites, by producing a biplot, (argument `biplot = TRUE` in `ordiplot()`).
- In a biplot latent variables and their loadings are rotated so that the LV loadings of the species are in the same direction with the sites where they are most abundant.
- Biplot can be used for finding indicator species corresponding to specific sites, or finding groups of correlated species.

Biplot

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

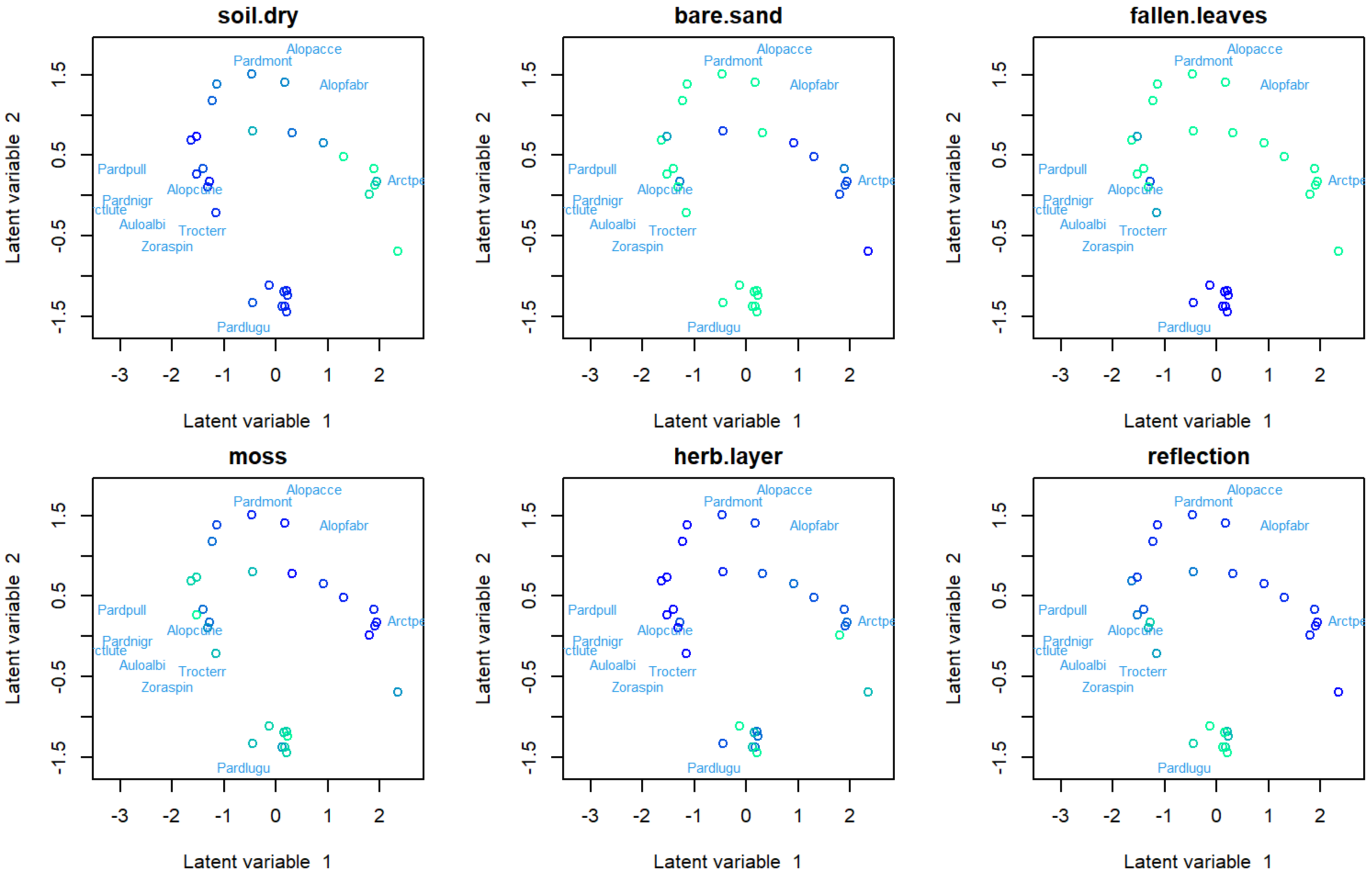


Environmental gradients

The potential impact of environmental variables on species communities can be viewed by coloring ordination points according to the variables.

```
# Arbitrary color palette, a vector length of 20
rbPal <- c("#00FA9A", "#00EC9F", "#00DFA4", "#00D2A9", "#00C5AF", ...)
X <- spider$x
par(mfrow = c(2,3), mar=c(4,4,2,2))
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, alpha = 0.55)
}
```

Environmental gradients



Environmental gradients

- The next step would be to include covariates to the model and this matter will be dealt with tomorrow.