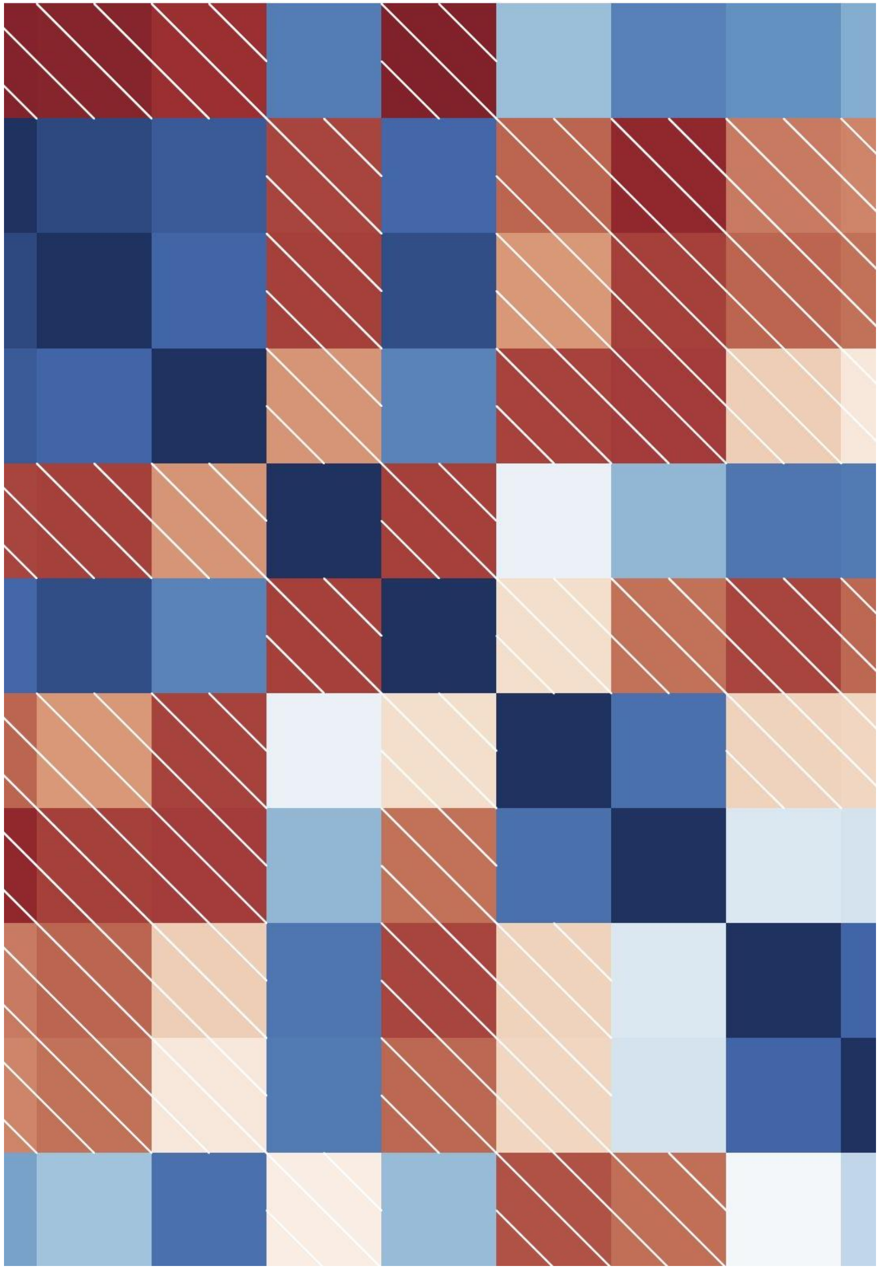
genDA

Using Variational Approximations to efficiently build a generalised discriminant analysis algorithm.

Presented by

Sarah Romanes

School of Mathematics and Statistics



Motivation

The origins of Discriminant Analysis (DA)

- First introduced by Fisher (1936)., DA is a multivariate technique used to classify observations into classes, and/or describe class differences.

- Used in many fields (such as applied psychological research) to develop efficient classification rules and assess relative importance of variables for discriminating between classes.

Assumptions of Discriminant Analysis

DA assumes that the conditional distribution of our data **X** given a class **y = c** is a **multivariate normal distribution:**

$$p(\mathbf{x}|y = c, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$$

We assign a new point **x** to the class which has the highest probability, where

$$p(y = c|\mathbf{x}, \boldsymbol{\theta}) \quad = \quad \frac{\pi_c |2\pi\boldsymbol{\Sigma}_c|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_c)^T \boldsymbol{\Sigma}_c^{-1} (\mathbf{x} - \boldsymbol{\mu}_c)\right]}{\sum_{c'} \pi_{c'} |2\pi\boldsymbol{\Sigma}_{c'}|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{c'})^T \boldsymbol{\Sigma}_{c'}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{c'})\right]}$$

Issues with Discriminant Analysis

Does not work in high dimensions (when p > n) as covariance inverse is singular.

Conditional distribution is assumed Gaussian - does not work when response type is non-Gaussian.

Solution: *Factor Analytic* models can be used to provide low rank representations of covariance matrix.

See HiDimDA (Duarte Silva, 2011) and FADA (Perthame, 2016)

Issues with Discriminant Analysis

Does not work in high dimensions (when p > n) as covariance inverse is singular.

Solution: *Factor Analytic* models can be used to provide low rank representations of covariance matrix.

See HiDimDA (Duarte Silva, 2011) and FADA (Perthame, 2016)

Conditional distribution is assumed Gaussian - does not work when response type is non-Gaussian.

Solution:

Issues with Discriminant Analysis

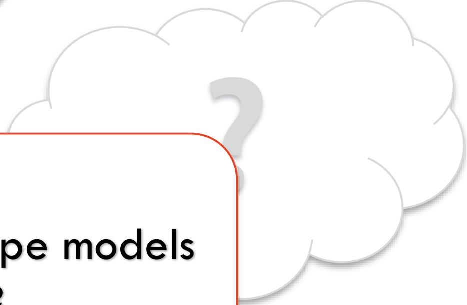
Does not work in high dimensions (when p > n) as covariance inverse is singular.

Conditional distribution is assumed Gaussian - does not work when response type is non-Gaussian.

Solution: *Factor Ana*
models can be used
low rank representations of
covariance matrix.

Sol on:

See HiDimDA (Du
and FADA (Perthc



Can we apply factor analysis type models to non Gaussian data?

Generalised Linear Latent Variable Models

Consider a matrix **Y** of *n* observations consisting of responses for *m* features.

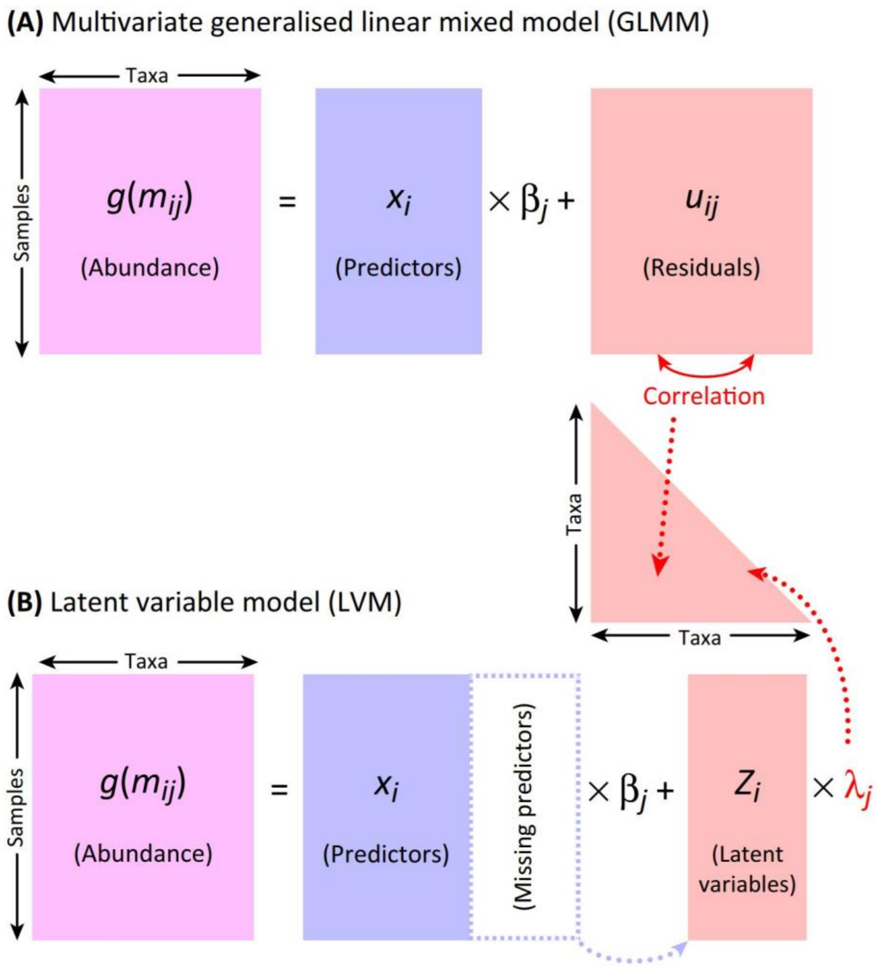
A generalised linear latent variable model (GLLVM) regresses the mean response against a vector of *d* << *m* latent variables ***u***$_i$, along with a vector of covariates **x**$_i$. That is,

$$g(\mu_{ij}) = \eta_{ij} = \tau_i + \beta_{0j} + \mathbf{x}_i^T \boldsymbol{\beta}_j + \mathbf{u}_i^T \boldsymbol{\lambda}_j$$

We assume the latent variables follow a multivariate standard normal distribution, $\mathbf{u}_i \sim \mathcal{N}_d(\mathbf{0}, \mathbf{I}_d)$.

Relationship between GLMMs and GLLVMs

Correlation can be handled in different ways

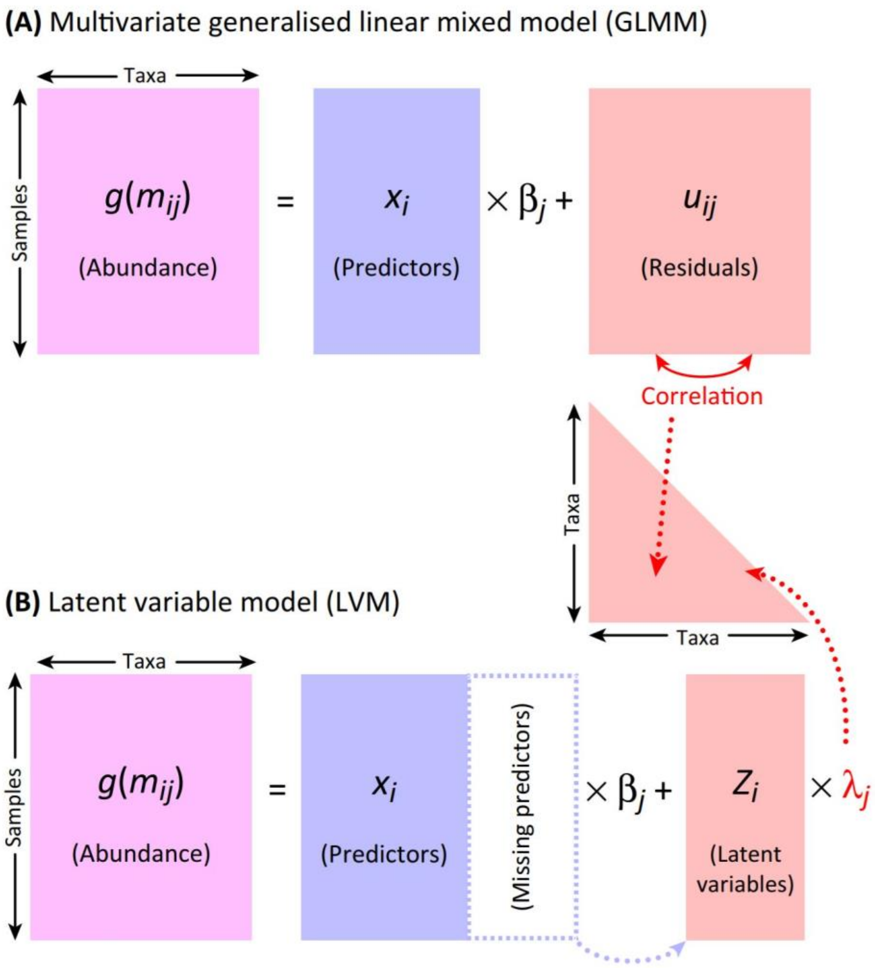


Warton et al, 2015

Relationship between GLMMs and GLLVMs

A GLMM uses correlated random effects to estimate correlation.

Correlation can be handled in different ways



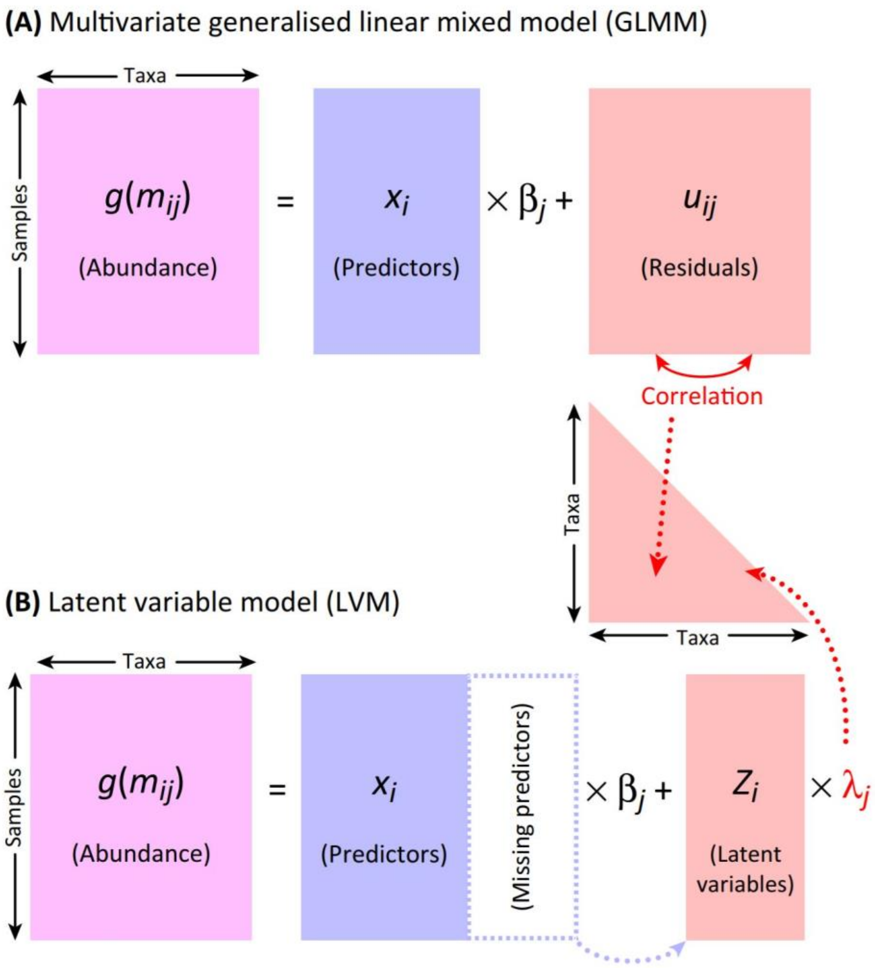
Warton et al, 2015

Relationship between GLMMs and GLLVMs

A GLMM uses correlated random effects to estimate correlation.

Correlation can be handled in different ways

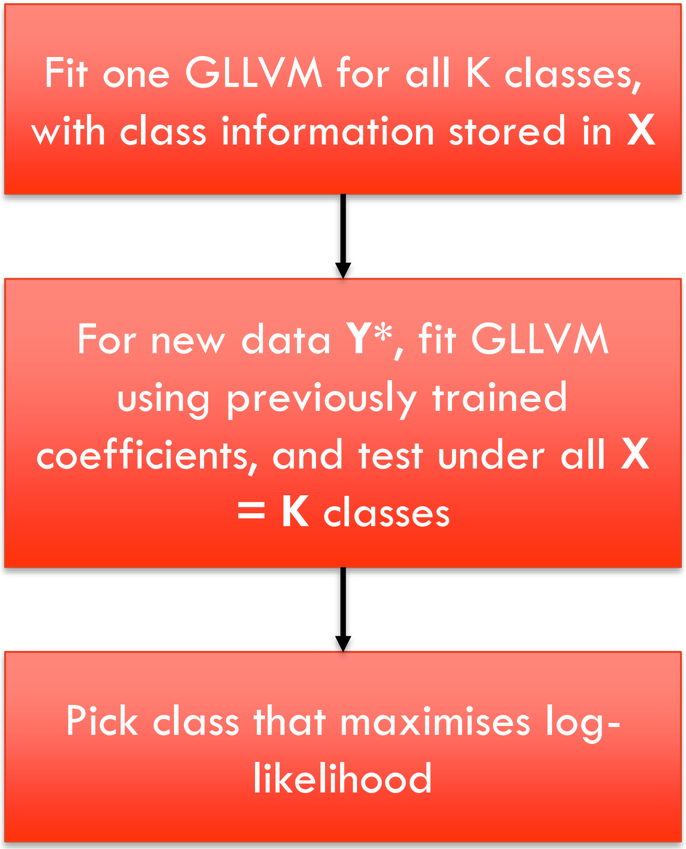
A GLLVM uses a smaller number of latent variables, which play the role of missing predictors. Their factor loadings approx. the correlation across features, but uses fewer parameters than a GLMM.



Warton et al, 2015

Using GLLVMs to build a generalised DA method

Common Covariance Model (LDA analogue)



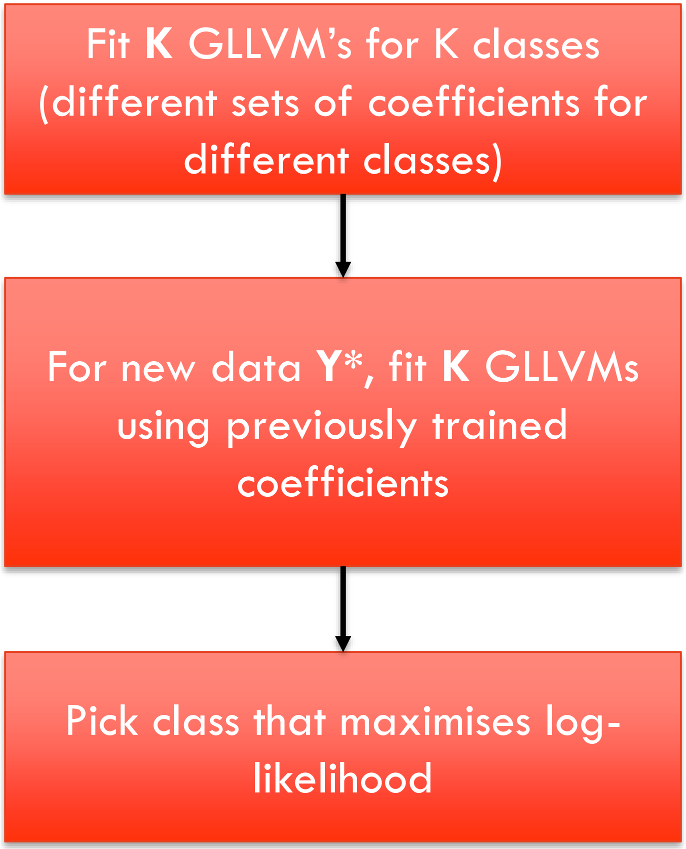
Fit one GLLVM for all K classes, with class information stored in **X**

For new data **Y***, fit GLLVM using previously trained coefficients, and test under all **X** = **K** classes

Pick class that maximises log-likelihood

Using GLLVMs to build a generalised DA method

Separate Covariance Model (QDA analogue)



Fit **K** GLLVM's for K classes (different sets of coefficients for different classes)

For new data **Y***, fit **K** GLLVMs using previously trained coefficients

Pick class that maximises log-likelihood

Step One: Fit GLLVM(s) to training data

GLLVM – model formulation

To complete the formulation, we assume conditional on the LV's and parameter vector, the responses are independent observations from the exponential family of distributions with probability density function

$$f(y_{ij}|\mathbf{u}_i, \boldsymbol{\Psi}) = \exp\left\{ \frac{y_{ij}\eta_{ij} - b(\eta_{ij})}{a(\phi_j)} + c(y_{ij}, \phi_j) \right\}$$

where $\boldsymbol{\Psi} = \{\boldsymbol{\tau}, \boldsymbol{\beta}_0, \boldsymbol{\phi}, \text{vec}(\boldsymbol{\lambda}), \text{vec}(\boldsymbol{\beta})\}$ (all parameters in model),

and $\tilde{\eta}_{ij} = \tau_i + \beta_{0j} + \mathbf{x}_i^T \boldsymbol{\beta}_j + \boldsymbol{\mu}_i^T \boldsymbol{\lambda}_{j:}$.

GLLVM – expression for the marginal distribution

With the independence structure given the latent variables,

$$p(\mathbf{y}_i|\mathbf{u}_i, \mathbf{\Psi}) = \prod_{j=1}^{m} p(y_{ij}|\mathbf{u}_i, \mathbf{\Psi})$$

GLLVM – expression for the marginal distribution

$$p(\mathbf{y}_i|\mathbf{u}_i, \mathbf{\Psi}) = \prod_{j=1}^{m} p(y_{ij}|\mathbf{u}_i, \mathbf{\Psi})$$

$$\begin{aligned}\ell(\mathbf{\Psi}) &= \sum_{i=1}^{n} \log p(\mathbf{y}_i, \mathbf{\Psi}) \\ &= \sum_{i=1}^{n} \log \left(\int \prod_{i=1}^{m} p(y_{ij}|\mathbf{u}_i, \mathbf{\Psi}) p(\mathbf{u}_i) d\mathbf{u}_i \right)\end{aligned}$$

With the independence structure given the latent variables, we can obtain the marginal log-likelihood function for a GLLVM by **integrating over the latent variables u.**

GLLVM – expression for the marginal distribution

$$p(\mathbf{y}_i|\mathbf{u}_i,\mathbf{\Psi}) = \prod_{j=1}^{m} p(y_{ij}|\mathbf{u}_i,\mathbf{\Psi})$$

$$\begin{aligned}\ell(\mathbf{\Psi}) &= \sum_{i=1}^{n} \log p(\mathbf{y}_i,\mathbf{\Psi}) \\ &= \sum_{i=1}^{n} \log\left(\int \prod_{i=1}^{m} p(y_{ij}|\mathbf{u}_i,\mathbf{\Psi})p(\mathbf{u}_i)d\mathbf{u}_i\right)\end{aligned}$$

With the independence structure given the latent variables, we can obtain the marginal log-likelihood function for a GLLVM by **integrating over the latent variables u.**

Unfortunately, this cannot be solved analytically for non-Gaussian response.

GLLVMs are difficult to estimate

Research has been done to efficiently estimate GLLVMs. These include:

GLLVMs are difficult to estimate

Research has been done to efficiently estimate GLLVMs. These include:

- **Huber, 2004** showed that the Laplace Approximation can be used to estimate GLLVMs from the exponential family.

GLLVMs are difficult to estimate

Research has been done to efficiently estimate GLLVMs. These include:

- **Huber, 2004** showed that the Laplace Approximation can be used to estimate GLLVMs from the exponential family.
- **Niku et. al, 2017** extended this work in Laplace Approximations to cover Tweedie, Negative Binomial, and ZIP.

GLLVMs are difficult to estimate

Research has been done to efficiently estimate GLLVMs. These include:

- **Huber, 2004** showed that the Laplace Approximation can be used to estimate GLLVMs from the exponential family.
- **Niku et. al, 2017** extended this work in Laplace Approximations to cover Tweedie, Negative Binomial, and ZIP.
- **Hui et. al, 2017** used Variational Approximations to estimate various types of GLLVMs, using a GVA approach.

GLLVMs are difficult to estimate

Research has been done to efficiently estimate GLLVMs. These include:

– **Huber, 2004** showed that the Laplace Approximation can be used to estimate GLLVMs from the exponential family.

– **Niku et. al, 2017** extended this work in Laplace Approximations to cover Tweedie, Negative Binomial, and ZIP.

– **Hui et. al, 2017** used Variational Approximations to estimate various types of GLLVMs, using a GVA approach.

– **Niku et. al, 2019** released an efficient package for estimating GLLVMS (`gllvm`) based on Laplace and Variational Approximations.

GLLVMs are difficult to estimate

Although the `gllvm` package is excellent, it can only support the data coming from one response type. Hence, we need to build our own functions to estimate GLLVMs for **differing response types.**

We currently support a mixture of

- Bernoulli,
- Poisson,
- Negative Binomial,
- Gaussian,
- Log-Normal, and
- Zero Inflated Poisson responses.

A Bayesian GLLVM – prior specification

Another twist we will add, in comparison to previous work, is approach this from a *Bayesian* framework, with the priors on our coefficients allowing us to incorporate *regularisation* in the fitting process. With this in mind, we set

$$\tau_i \sim N(0, \sigma^2_{\tau_i}), \qquad \beta_{0j} \sim N(\mathbf{0}, \sigma^2_{\beta_{0j}}\mathbf{I})$$

$$\boldsymbol{\beta}_j \sim N(\mathbf{0}, \sigma^2_{\beta_j}\mathbf{I}) \qquad \boldsymbol{\lambda}_j \sim N(\mathbf{0}, \sigma^2_{\lambda_j}\mathbf{I})$$

and for the dispersion parameter (if applicable): $p(\phi_j) \propto 1$

Variational family

We consider the following parameterisation to construct our variational lower bound

$$q(\mathbf{u}, \mathbf{\Psi}) = \prod_{i=1}^{n} q(\mathbf{u}_i) q(\mathbf{\Psi})$$

where we assume both $q(\mathbf{u}_i)$ and $q(\mathbf{\Psi})$ are multivariate normal distributions.

A Gaussian Variational Approximation (GVA)

Hui et. al (2017) showed that for GLLVMs, it is optimal to use GVA and assume the following *q* density for the LVs:

$$q(\mathbf{u}_i) \equiv N_d(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \ .$$

A Gaussian Variational Approximation (GVA)

Hui et. al (2017) showed that for GLLVMs, it is optimal to use GVA and assume the following *q* density for the LVs:

$$q(\mathbf{u}_i) \equiv N_d(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \ .$$

In particular, they show that for GLLVMs from the exponential family, the VA log likelihood is in the following form:

$$\underline{\ell}(\boldsymbol{\Psi}, \boldsymbol{\xi}) = \sum_{i=1}^{n} \sum_{j=1}^{m} \left\{ \frac{y_{ij}\tilde{\eta}_{ij} - E_q\{b(\eta_{ij})\}}{\phi_j} + c(y_{ij}, \phi_j) \right\} + \frac{1}{2} \sum_{i=1}^{n} \left(\log\det(\boldsymbol{\Sigma}_i) - \operatorname{tr}(\boldsymbol{\Sigma}_i) - \boldsymbol{\mu}_i^T \boldsymbol{\mu}_i \right)$$

where $\tilde{\eta}_{ij} = \tau_i + \beta_{0j} + \mathbf{x}_i^T \boldsymbol{\beta}_j + \boldsymbol{\mu}_i^T \boldsymbol{\lambda}_j$, and $\boldsymbol{\xi}_i = \{\boldsymbol{\mu}_i, \operatorname{vech}(\boldsymbol{\Sigma}_i)\}$.

A Gaussian Variational Approximation (GVA)

Hui et. al (2017) showed that for GLLVMs, it is optimal to use GVA and assume the following *q* density for the LVs:

$$q(\mathbf{u}_i) \equiv N_d(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \,.$$

GVA Terms

Given this, we construct our variational lower bound as follows:

$$\begin{aligned}\log \underline{p}(\mathbf{y}, \boldsymbol{\tau}, \boldsymbol{\beta}_0, \mathbf{B}, \mathbf{L}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} & \tfrac{1}{2}\log|\boldsymbol{\Sigma}_i| + \mathbf{y}_i^T(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) \\ & - \mathbf{1}^T\mathbb{E}_{\mathbf{u}_i}\left(b(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\mathbf{u}_i)\right) \\ & - \frac{\|\boldsymbol{\mu}_i\|^2}{2} - \frac{\mathrm{tr}(\boldsymbol{\Sigma}_i)}{2} - \frac{\|\boldsymbol{\tau}\|^2}{2\sigma_\tau^2} - \frac{\|\boldsymbol{\beta}_0\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\beta}_j\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\lambda}_j\|^2}{2\sigma_\lambda^2}.\end{aligned}$$

A Gaussian Variational Approximation (GVA)

Hui et. al (2017) showed that for GLLVMs, it is optimal to use GVA and assume the following *q* density for the LVs:

$$q(\mathbf{u}_i) \equiv N_d(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\ .$$

GLLVM Terms

Given this, we construct our variational lower bound as follows:

$$\log \underline{p}(\mathbf{y}, \boldsymbol{\tau}, \boldsymbol{\beta}_0, \mathbf{B}, \mathbf{L}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} \tfrac{1}{2} \log |\boldsymbol{\Sigma}_i| + \mathbf{y}_i^T (\mathbf{1}_m \tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i)$$

$$- \mathbf{1}^T \mathbb{E}_{\mathbf{u}_i} \left(b(\mathbf{1}_m \tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\mathbf{u}_i) \right)$$

$$- \frac{\|\boldsymbol{\mu}_i\|^2}{2} - \frac{\mathrm{tr}(\boldsymbol{\Sigma}_i)}{2} - \frac{\|\boldsymbol{\tau}\|^2}{2\sigma_\tau^2} - \frac{\|\boldsymbol{\beta}_0\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m} \frac{\|\boldsymbol{\beta}_j\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m} \frac{\|\boldsymbol{\lambda}_j\|^2}{2\sigma_\lambda^2}.$$

A Gaussian Variational Approximation (GVA)

Hui et. al (2017) showed that for GLLVMs, it is optimal to use GVA and assume the following *q* density for the LVs:

$$q(\mathbf{u}_i) \equiv N_d(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\ .$$

Prior Terms

Given this, we construct our variational lower bound as follows:

$$\begin{aligned}\log \underline{p}(\mathbf{y}, \boldsymbol{\tau}, \boldsymbol{\beta}_0, \mathbf{B}, \mathbf{L}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} & \tfrac{1}{2}\log|\boldsymbol{\Sigma}_i| + \mathbf{y}_i^T(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) \\ & - \mathbf{1}^T\mathbb{E}_{\mathbf{u}_i}\left(b(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\mathbf{u}_i)\right) \\ & - \frac{||\boldsymbol{\mu}_i||^2}{2} - \frac{\mathrm{tr}(\boldsymbol{\Sigma}_i)}{2} - \frac{||\boldsymbol{\tau}||^2}{2\sigma_\tau^2} - \frac{||\boldsymbol{\beta}_0||^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{||\boldsymbol{\beta}_j||^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{||\boldsymbol{\lambda}_j||^2}{2\sigma_\lambda^2}.\end{aligned}$$

A Gaussian Variational Approximation (GVA)

Hui et. al (2017) showed that for GLLVMs, it is optimal to use GVA and assume the following *q* density for the LVs:

$$q(\mathbf{u}_i) \equiv N_d(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\ .$$

How to deal with this expectation?

Given this, we construct our variational lower bound as follows:

$$\begin{aligned}\log \underline{p}(\mathbf{y}, \boldsymbol{\tau}, \boldsymbol{\beta}_0, \mathbf{B}, \mathbf{L}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^{n} &\tfrac{1}{2}\log|\boldsymbol{\Sigma}_i| + \mathbf{y}_i^T(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) \\ &- \mathbf{1}^T\mathbb{E}_{\mathbf{u}_i}\left(b(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\mathbf{u}_i)\right) \\ &- \frac{\|\boldsymbol{\mu}_i\|^2}{2} - \frac{\mathrm{tr}(\boldsymbol{\Sigma}_i)}{2} - \frac{\|\boldsymbol{\tau}\|^2}{2\sigma_\tau^2} - \frac{\|\boldsymbol{\beta}_0\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\beta}_j\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\lambda}_j\|^2}{2\sigma_\lambda^2}.\end{aligned}$$

A Second Order Delta Method Approximation

Using the Delta Method, we approximate the expectation using second order terms as follows:

$$\mathbb{E}(f(\mathbf{X})) \approx f(\mathbb{E}(\mathbf{X})) + \frac{1}{2}\mathrm{tr}\big(H(\mathbb{E}(\mathbf{X}))\mathrm{Cov}(\mathbf{X})\big)$$

$$\begin{aligned}\log \underline{p}(\mathbf{y}, \boldsymbol{\tau}, \boldsymbol{\beta}_0, \mathbf{B}, \mathbf{L}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \approx \sum_{i=1}^{n} & \tfrac{1}{2}\log|\boldsymbol{\Sigma}_i| + \mathbf{y}_i^T(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) \\ & - \mathbf{1}^T b(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) - \frac{\|\boldsymbol{\mu}_i\|^2}{2} \\ & - \tfrac{1}{2}\mathrm{tr}\left[\boldsymbol{\Sigma}_i\left\{\mathbf{L}^T\mathrm{diag}(b''(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i))\mathbf{L} + \mathbf{I}\right\}\right] \\ & - \frac{\|\boldsymbol{\tau}\|^2}{2\sigma_\tau^2} - \frac{\|\boldsymbol{\beta}_0\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\beta}_j\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\lambda}_j\|^2}{2\sigma_\lambda^2}.\end{aligned}$$

Profile likelihood

Given the approximated likelihood, we obtain a profile likelihood by first optimising for nuisance parameters, which in this case is the covariance of the GVA *q* density.

$$
\begin{aligned}
\log \underline{p}(\mathbf{y}, \boldsymbol{\tau}, \boldsymbol{\beta}_0, \mathbf{B}, \mathbf{L}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \approx \sum_{i=1}^{n} & \tfrac{1}{2}\log|\boldsymbol{\Sigma}_i| + \mathbf{y}_i^T(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) \\
& - \mathbf{1}^T b(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) - \frac{\|\boldsymbol{\mu}_i\|^2}{2} \\
& - \tfrac{1}{2}\mathrm{tr}\left[\boldsymbol{\Sigma}_i\left\{\mathbf{L}^T\mathrm{diag}(b''(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i))\mathbf{L} + \mathbf{I}\right\}\right] \\
& - \frac{\|\boldsymbol{\tau}\|^2}{2\sigma_\tau^2} - \frac{\|\boldsymbol{\beta}_0\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\beta}_j\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\lambda}_j\|^2}{2\sigma_\lambda^2}.
\end{aligned}
$$

First order optimality conditions imply:

$$
\widehat{\boldsymbol{\Sigma}}_i = \left[\mathbf{L}^T\mathrm{diag}(b''(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i))\mathbf{L} + \mathbf{I}\right]^{-1}
$$

Profile likelihood

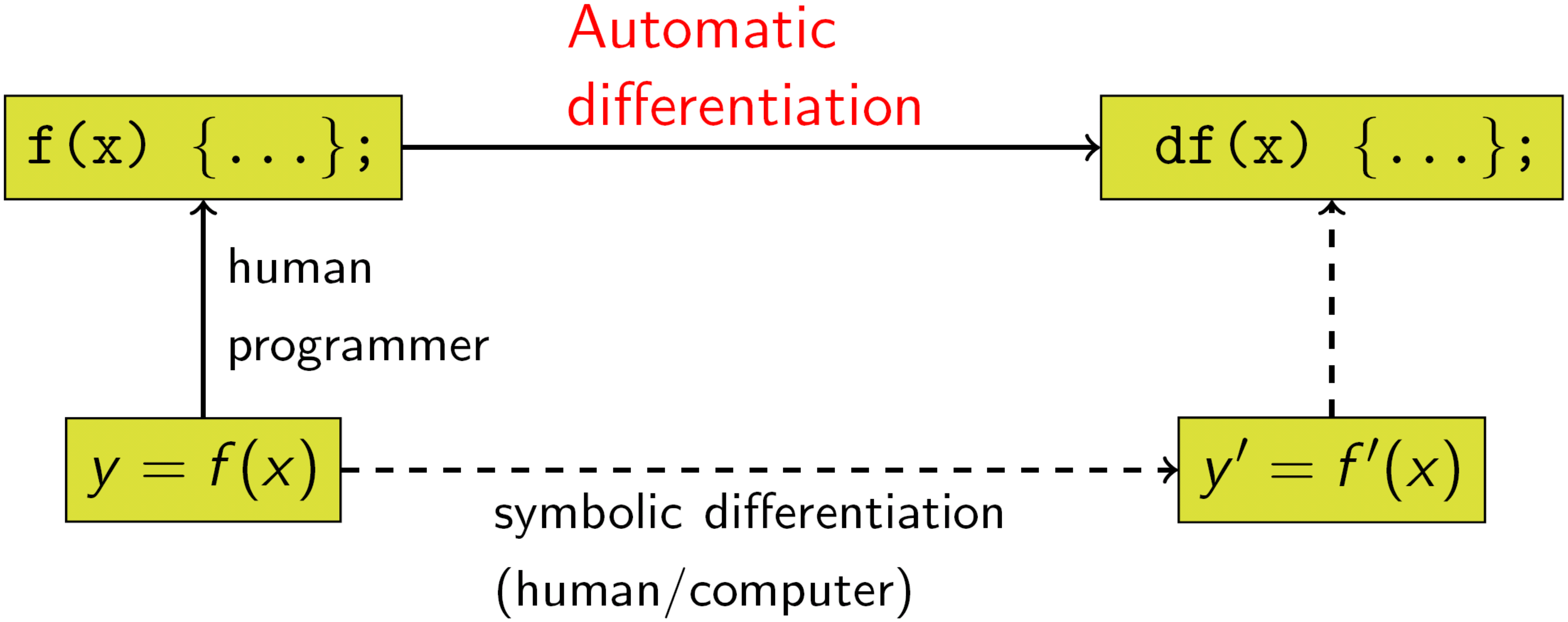
Substituting this expression back into our likelihood, we obtain the following profile likelihood:

$$\begin{aligned}\log \underline{p}(\mathbf{y}, \boldsymbol{\tau}, \boldsymbol{\beta}_0, \mathbf{B}, \mathbf{L}, \boldsymbol{\mu}) \approx \sum_{i=1}^{n} & -\tfrac{1}{2}\log|\mathbf{L}^T\mathrm{diag}(b''(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i))\mathbf{L} + \mathbf{I}| \\ & + \mathbf{y}_i^T(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) - \mathbf{1}^T b(\mathbf{1}_m\tau_i + \boldsymbol{\beta}_0 + \mathbf{B}\mathbf{x}_i + \mathbf{L}\boldsymbol{\mu}_i) \\ & - \frac{\|\boldsymbol{\mu}_i\|^2}{2} - \frac{\|\boldsymbol{\tau}\|^2}{2\sigma_\tau^2} - \frac{\|\boldsymbol{\beta}_0\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\beta}_j\|^2}{2\sigma_\beta^2} - \sum_{j=1}^{m}\frac{\|\boldsymbol{\lambda}_j\|^2}{2\sigma_\lambda^2}.\end{aligned}$$

We then perform Laplace’s method to obtain a multivariate normal approximation for $q(\boldsymbol{\Psi})$.

Software Implementation – Automatic Differentiation (AD)

Optimisation of LB is performed with the help of AD to calculate function gradients


Automatic differentiation
f(x) {...};
df(x) {...};
human programmer
$y = f(x)$
$y' = f'(x)$
symbolic differentiation (human/computer)

Software Implementation – Automatic Differentiation (AD)

To take advantage of this technique – the TMB package returns an AD gradient of functions which can be passed into optimisation routines such as `nlminb` instead of relying on inbuilt numeric differentiation.

The catch? The function to be optimised must be written using a C++ template.

genDA – the algorithm

1. Determine families of columns of data to be estimated, as well as separate out class variable (as a factor).
2. Initialise parameters to be estimated. LV parameters can be estimated using a FA approach (Niku, 2019).
3. Optimise derived approximate log lower bound by using TMB and nlminb, and report fitted values.
4. (Optional) standard errors can also be calculated by looking at inverse Hessian matrix.

Step Two: Use fitted model to predict new testing points

A simple prediction approach

Suppose we want to predict a new class value, $\mathbf{x}_i$*, given new feature information $\mathbf{Y}_i$*.

To obtain an expression for the joint lower bound for the new datapoints, we perform a first order delta method to substitute previously optimised parameters not depending on *i*, and optimise over values of the new row effects $\boldsymbol{\tau}_i$*, and LVs $\mathbf{u}_i$*:

$$\log \underline{p}(\mathbf{Y}_i^*, \mathbf{x}_i^*) = \log \int \prod_{j=1}^{m} p(\mathbf{Y}_{i,\cdot}^* | \boldsymbol{\tau}_i^*, \widehat{\boldsymbol{\beta}}_j, \mathbf{u}_i^*, \widehat{\boldsymbol{\lambda}}_j, \mathbf{x}_i^*) p(\boldsymbol{\tau}_i^*, \mathbf{u}_i^*) p(\mathbf{x}_i^* | \boldsymbol{\alpha}) p(\boldsymbol{\alpha}) d\mathbf{u}_i^* d\boldsymbol{\tau}_i^* d\boldsymbol{\rho}$$

A Dirichlet Multinomial Distribution for class

We assume that $\mathbf{x}_i^*$ follows a multinomial distribution, with prior probabilities $\boldsymbol{\rho}_k$ depending on concentration parameters $\boldsymbol{\alpha}_k$.

We can directly obtain an analytical expression for p($\mathbf{x}_i^*$| $\boldsymbol{\alpha}_k$) as follows:

$$p(\mathbf{x}^*|\boldsymbol{\alpha}) = \int_\rho p(\mathbf{x}^*|\boldsymbol{\rho})p(\boldsymbol{\rho}|\boldsymbol{\alpha})d\boldsymbol{\rho}$$

$$p(\mathbf{x}^*|\boldsymbol{\alpha}) = \mathrm{DirMult}(\mathbf{x}^*|\boldsymbol{\alpha}) = \frac{\Gamma\left(\sum_k \alpha_k\right)}{\Gamma\left(\sum_k n_k + \alpha_k\right)} \prod_{k=1}^{K} \frac{\Gamma(n_k + \alpha_k)}{\Gamma(\alpha_k)}$$

where $n_k = \sum_i \mathbb{I}(\mathbf{x} = k)$. .

MAP estimates

To classify a new point $\mathbf{x}_i$*, we find the probability of class membership under each class, and pick the probability that maximises the likelihood of inclusion. Probabilities are generated through the following:

$$
\begin{aligned}
\tilde{p}(\mathbf{x}_i^* = k|\mathbf{Y}_i^*) \propto \tilde{p}(\mathbf{Y}_i^*, \mathbf{x}_i^*) \\
&= \frac{\tilde{p}(\mathbf{Y}_i^*, \mathbf{x}_i^* = k)}{\sum_{s=1}^{K} \tilde{p}(\mathbf{Y}_i^*, \mathbf{x}_i^* = s)} \\
&= \frac{\kappa_i(\mathbf{Y}_i^*, \mathbf{x}_i^* = k)}{\sum_{s=1}^{K} \kappa_i(\mathbf{Y}_i^*, \mathbf{x}_i^* = s)}
\end{aligned}
$$

where $\kappa_i = \max \ \log \underline{p}(\mathbf{Y}_i^*, \mathbf{x}_i^*)$. with respect to $\boldsymbol{\tau}_i$*, $\mathbf{u}_i$*.

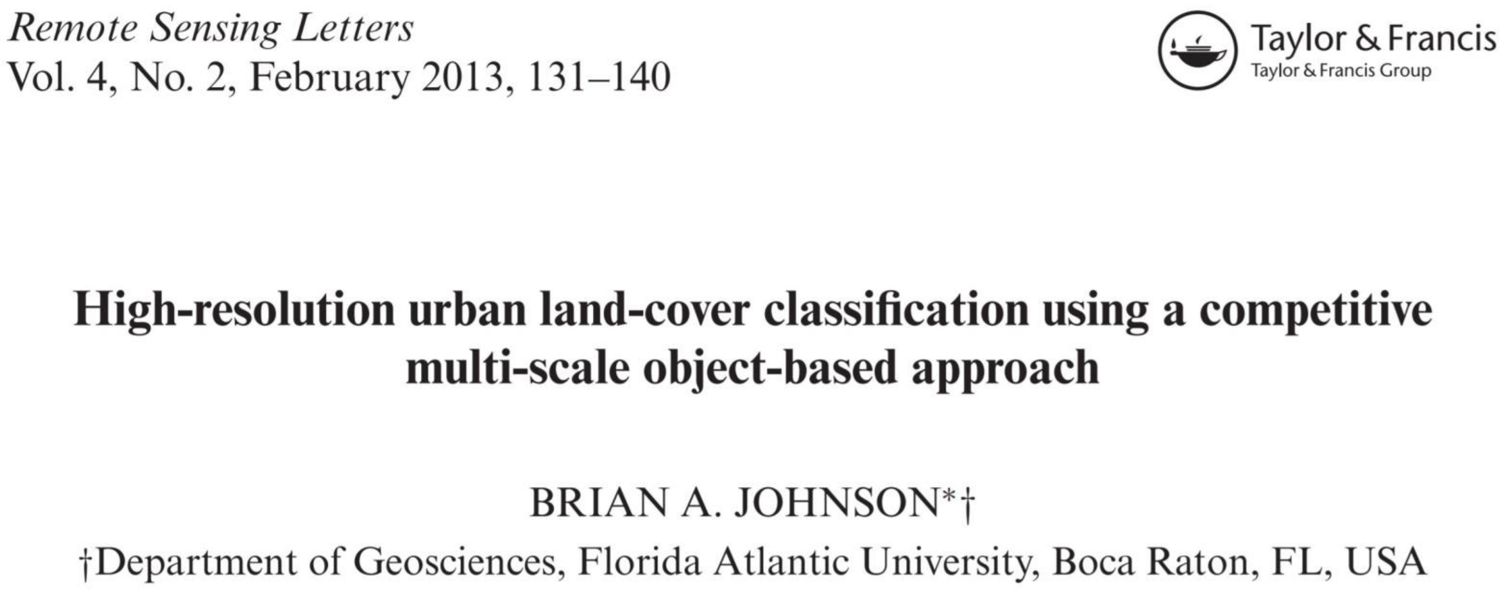
genDA – the prediction algorithm

1. Initialise new LVs
2. Using fitted values from genDA and new data **Y***, estimate log likelihood of new data under each class.
3. **Assign class such that the log-likelihood is maximised.**
4. Return predicted class as well as probability of class membership.

Lets see if this works!

Benchmark data analysis

genDA in action – Urban Land-cover data



Remote Sensing Letters
Vol. 4, No. 2, February 2013, 131–140

Taylor & Francis
Taylor & Francis Group

High-resolution urban land-cover classification using a competitive multi-scale object-based approach

BRIAN A. JOHNSON*†

†Department of Geosciences, Florida Atlantic University, Boca Raton, FL, USA

Can we predict segment class in images?

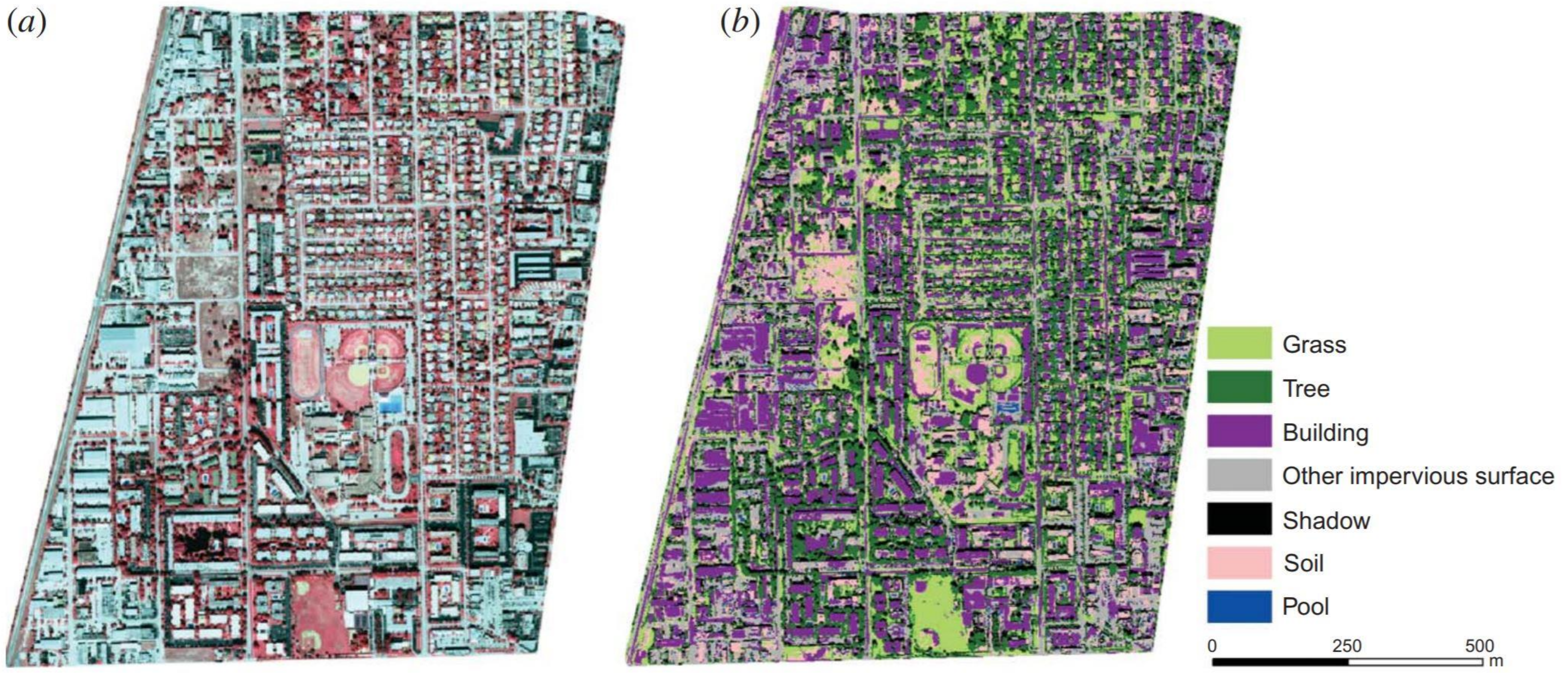


Figure 2. A 30-cm resolution digital orthoimage of the study area (*a*) and the land-cover map produced by the most accurate multi-scale classification (*b*).

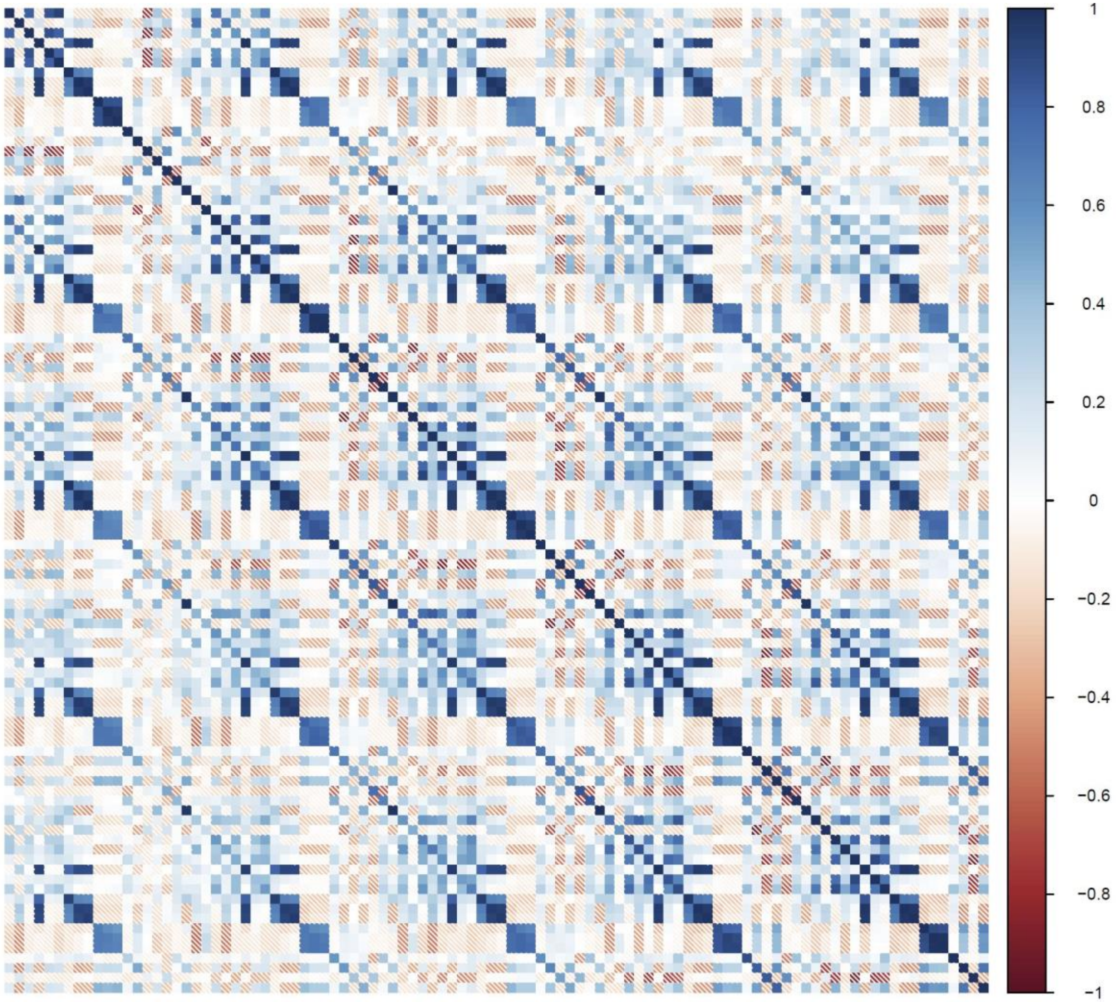
Can we predict segment class in images?

- The study area is an urban area in Deerfield Beach, FL, USA, with a 30cm resolution colour infrared aerial orthoimagery of the study area acquired.

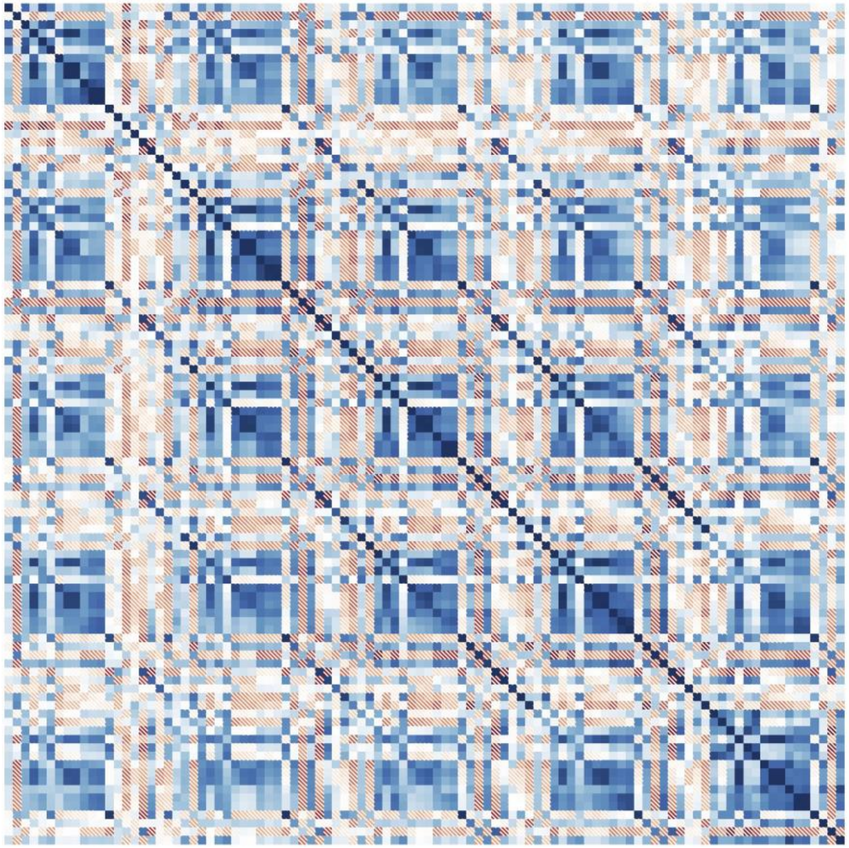
- Contains 9 different types of landcover including buildings, concrete, asphalt, trees, grass, pools, soil, cars, and shadows.

- Data consists of **n = 168** image segments to be classified with **m = 147** features associated with each image segment such as area, brightness, texture, etc at different resolutions. Features are a mix of Gaussian, Log-Normal, and Negative Binomial response.

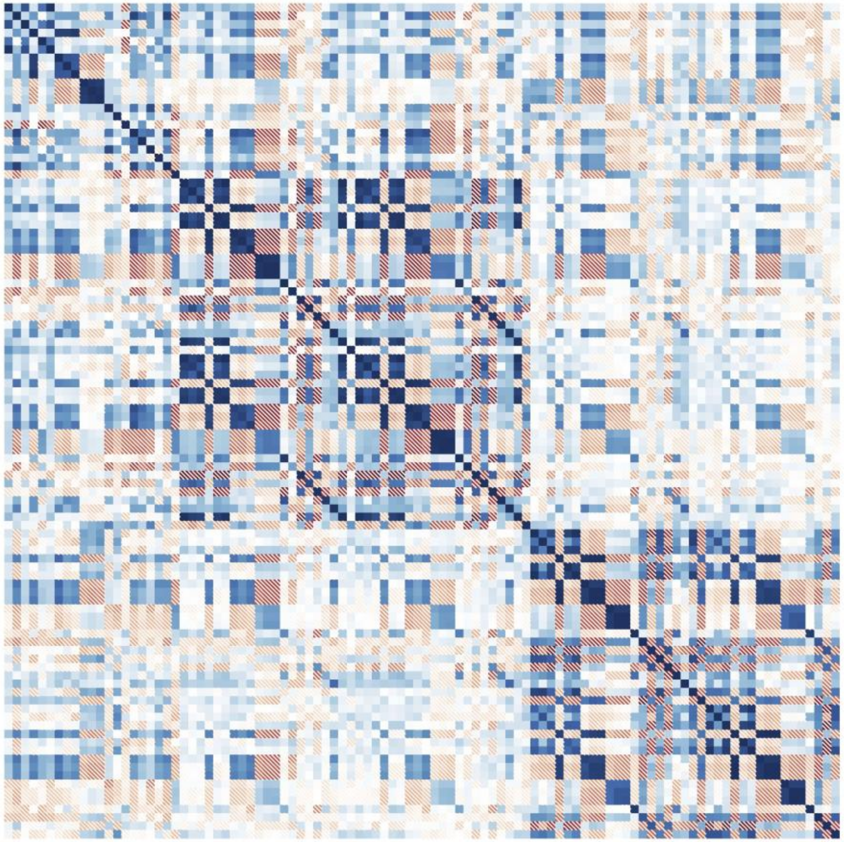
Correlation of Features (first 100 variables)



Dig a little deeper – a difference in correlation structure

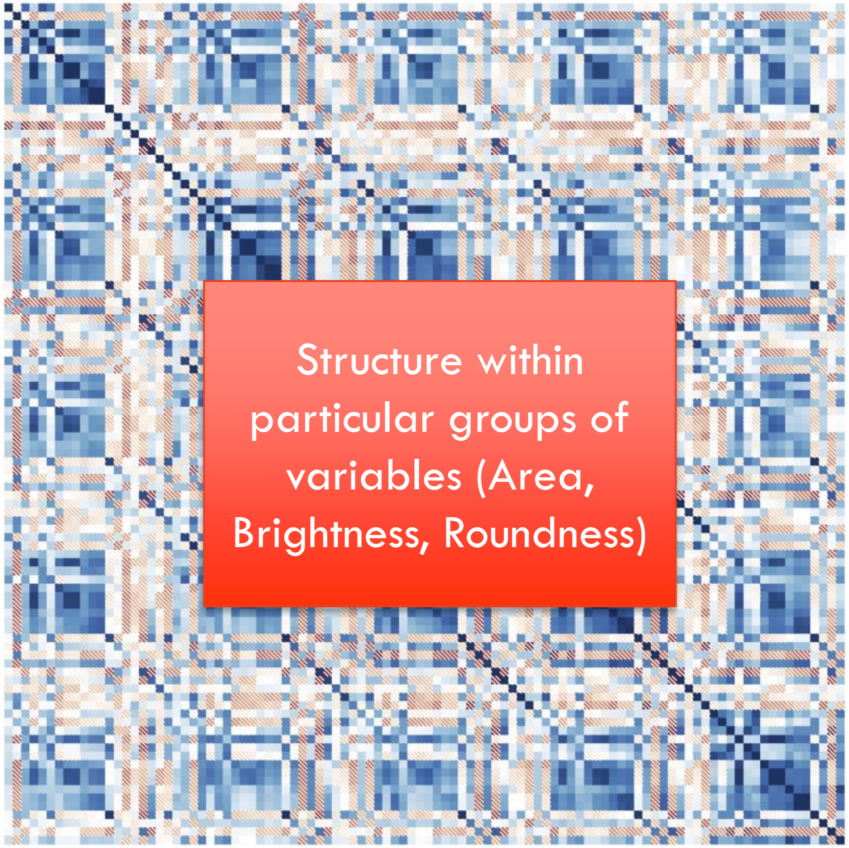


Class = **Shadow**



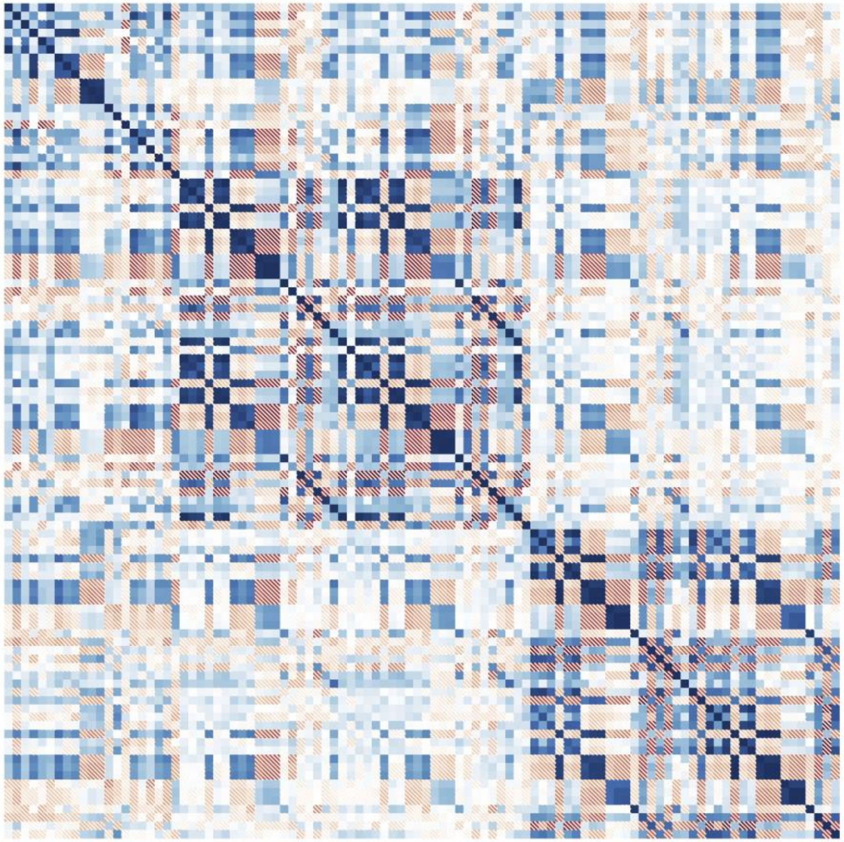
Class = **Car**

Difference in Correlation Structure (first 100 variables)



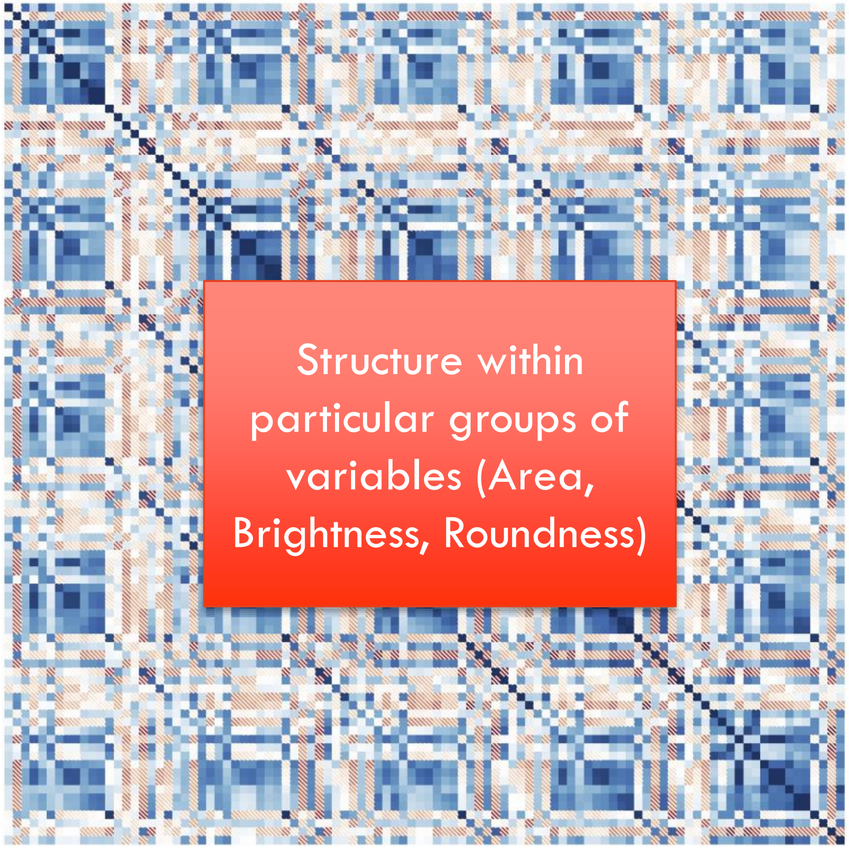
Structure within particular groups of variables (Area, Brightness, Roundness)

Class = **Shadow**



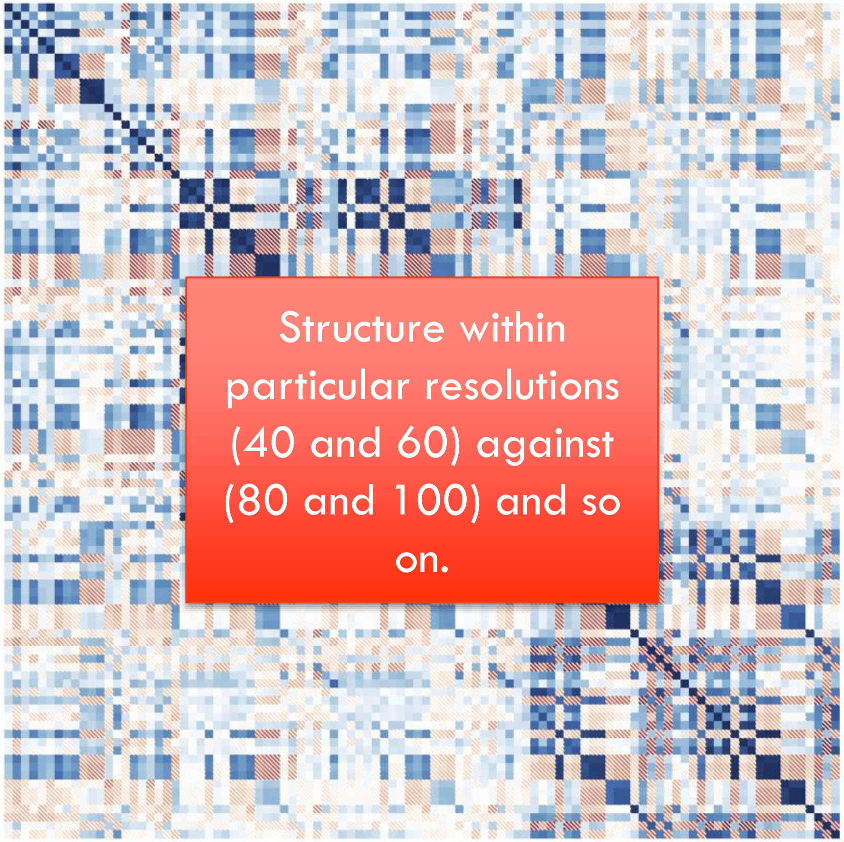
Class = **Car**

Difference in Correlation Structure (first 100 variables)



Structure within particular groups of variables (Area, Brightness, Roundness)

Class = **Shadow**



Structure within particular resolutions (40 and 60) against (80 and 100) and so on.

Class = **Car**

Classification – the genDA approach

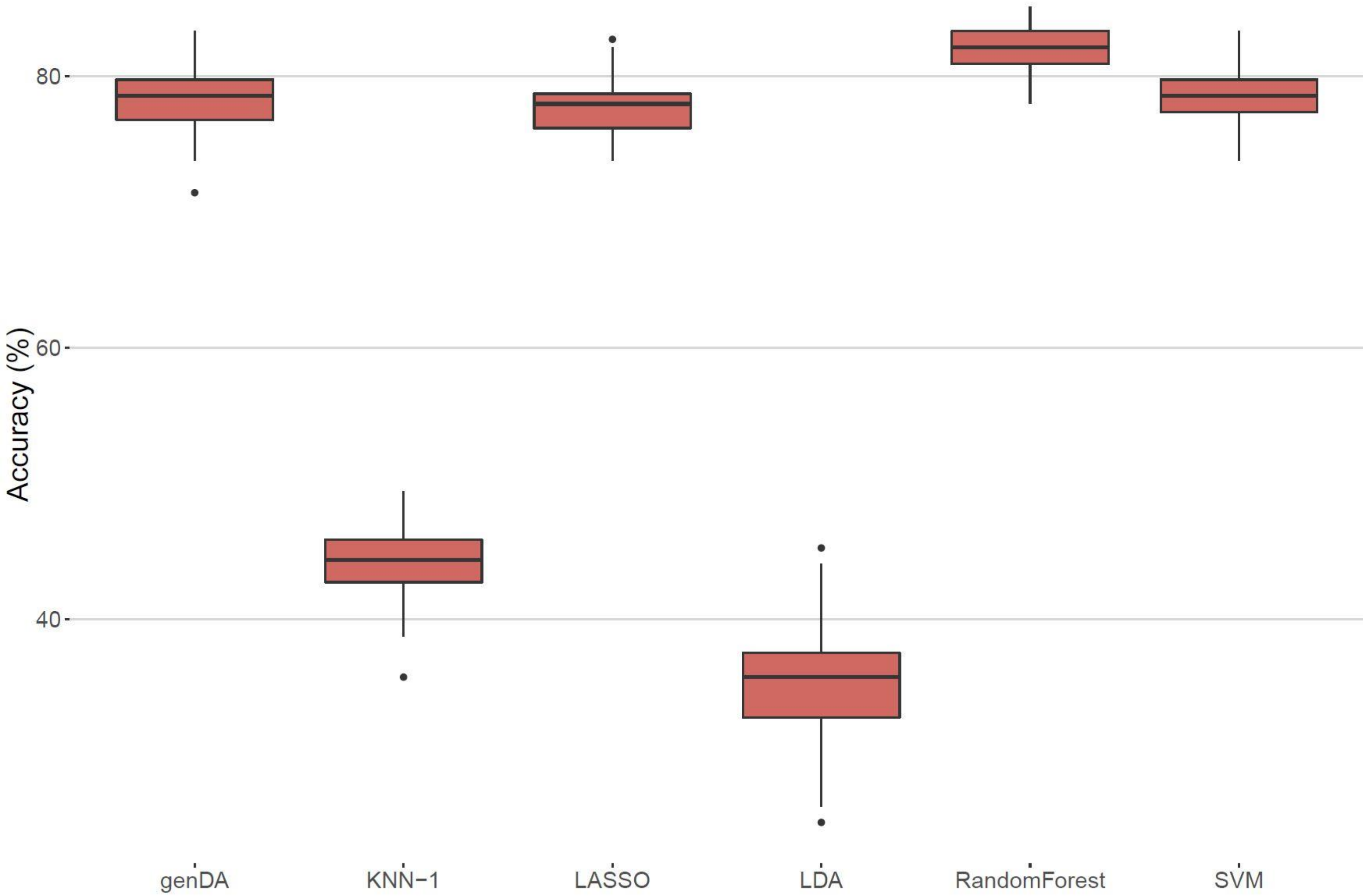
Since correlation structure differs greatly between classes, we fit a QDA type model to Urban Cover data – and test performance with a 100 x 5 Fold CV procedure. Using our genDA R package, we can perform this quite easily:

```
fit = genDA(y = y.train, class = class.train,
common.covariance = FALSE)
```

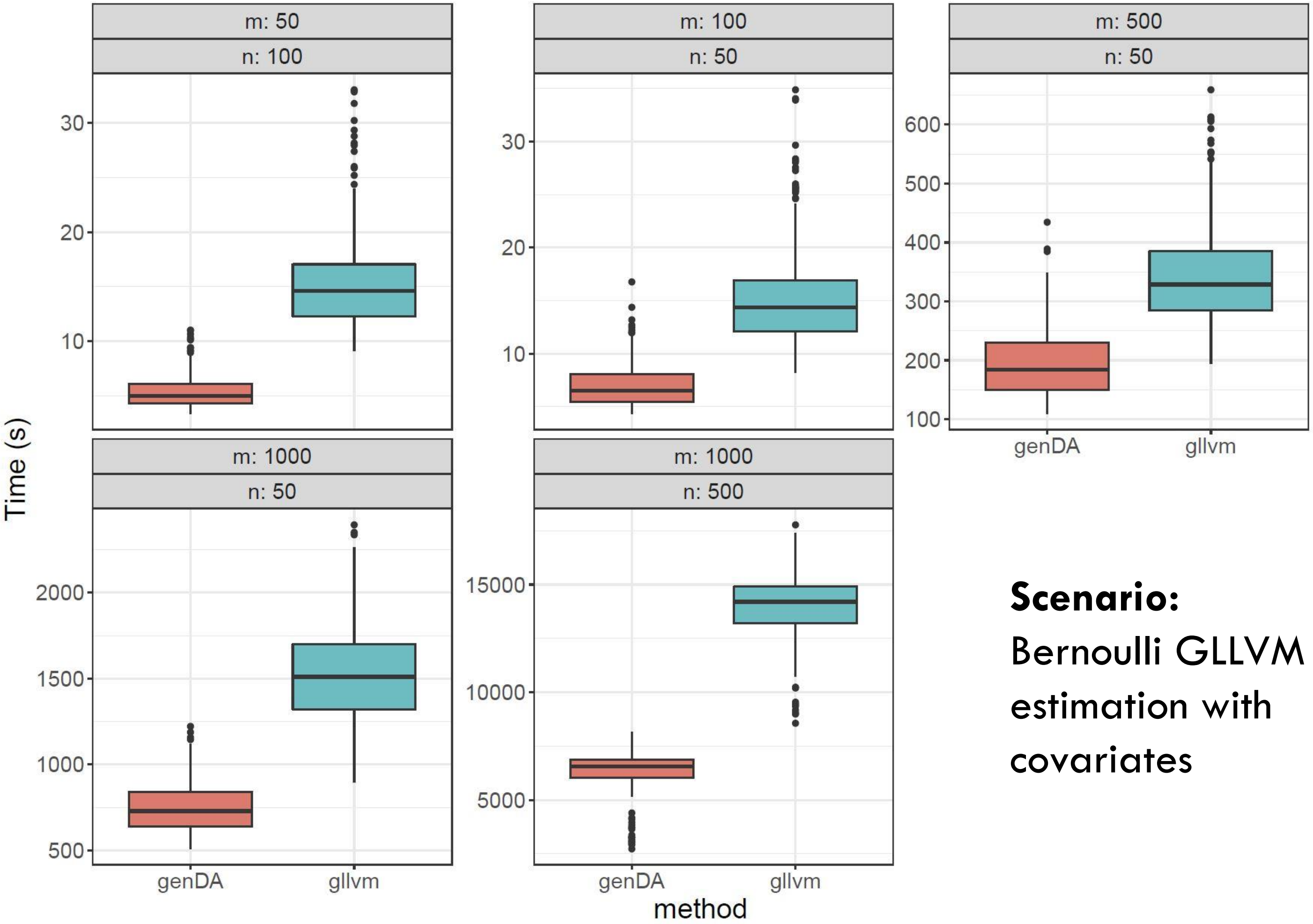
```
pred = predict(fit, newdata = y.test)$class
```

```
mean(pred!=class.test)
```

100 Trial, 5 Fold CV results

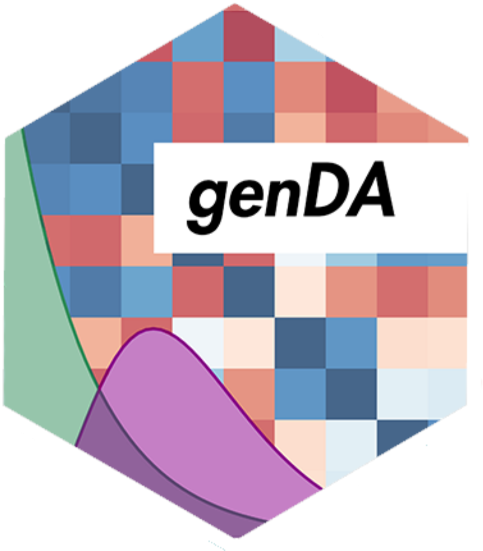


Comments on speed and coverage



Scenario:
Bernoulli GLLVM estimation with covariates

genDA package



A fast, efficient, and easy to use **R** implementation based on this work is available at the following address:

```
https://sarahromanes.github.io/genDA
```

Further research directions

- Feature selection for GLLVMs remains an open problem and would likely see improvements in predictive performance.

- Investigate effects of altering prior dispersion for coefficient parameters. Currently fixed – perhaps a `cv.glmnet` like approach would be optimal.

Thank you!

Work presented today is in collaboration with my supervisor A/Prof John Ormerod.

Many thanks to the USYD Statistical Bioinformatics group, for all their support and guidance!

Get in touch!

email: sarah.romanes@sydney.edu.au

twitter: @sarah_romanes

sarahromanes.github.io

References – Factor Analysis

- Perthame, É., Friguet, C. & Causeur, D. **Stability of feature selection in classification issues for high-dimensional correlated data** *Statistics and Computing, 26: 783, 2016*
- Pedro Duarte Silva, A. **Two Group Classification with High-Dimensional Correlated Data: A Factor Model Approach,** *Computational Statistics and Data Analysis, 55* (1), 2975-2990, *2011*

References - GLLVMs

- Huber, P. Ronchetti, E. Victoria-Feser, M. **Estimation of Generalized Linear Latent Variable Models** *Journal of the Royal Statistical Society, Vol. 66, No. 4, pp. 893-908, 2004.*
- Hui, F. Warton, D. Ormerod, J. Haapaniemi, V. Taskinen, S. **Variational Approximations for Generalized Linear Latent Variable Models.** *Journal of Computational and Graphical Statistics* 26, no. 1 2017
- Niku, J. Warton, D. Hui, F. Taskinen, S. **Generalized Linear Latent Variable Models for Multivariate Count and Biomass Data in Ecology.** *Journal of Agricultural, Biological and Environmental Statistics* 22, no. 4 498–522, *2017*
- Niku, J. Brooks, W. Herliansyah, R. Hui, F. Taskinen, S. Warton, D. **Efficient Estimation of Generalized Linear Latent Variable Models.** *PLOS ONE* 14, no. 5 *2019*