University of South Australia

Regression-based approaches for simulation meta-modelling in the presence of heterogeneity and correlation

Len Patrick Dominic Garces, Timofei Bogomolov, Belinda Chiera

University of South Australia, UniSA STEM, Adelaide, South Australia 5000

8 December 2021

Presentation Outline

Motivation: Simulation Meta-Modelling

- Simulation studies are used to inform recommendations and support decision-making on competing outputs simulated under comparable conditions.
- Simulations typically involve running m replications for each of n unique input combinations/design points.
- However, as $m, n \nearrow \infty$ simulation analysis can become enormously time-consuming.
- The use of *meta-models* to obtain a functional approximation of the (black-box) relationship between the inputs and outputs addresses this issue.

Motivation: Simulation Meta-Modelling

- Simulation studies are used to inform recommendations and support decision-making on competing outputs simulated under comparable conditions.
- Simulations typically involve running m replications for each of n unique input combinations/design points.
- However, as $m, n \nearrow \infty$ simulation analysis can become enormously time-consuming.
- The use of *meta-models* to obtain a functional approximation of the (black-box) relationship between the inputs and outputs addresses this issue.

Motivation: Simulation Meta-Modelling

- Simulation studies are used to inform recommendations and support decision-making on competing outputs simulated under comparable conditions.
- Simulations typically involve running m replications for each of n unique input combinations/design points.
- However, as $m, n \nearrow \infty$ simulation analysis can become enormously time-consuming.
- The use of *meta-models* to obtain a functional approximation of the (black-box) relationship between the inputs and outputs addresses this issue.

Motivation: Simulation Meta-Modelling

- Simulation studies are used to inform recommendations and support decision-making on competing outputs simulated under comparable conditions.
- Simulations typically involve running m replications for each of n unique input combinations/design points.
- However, as $m, n \nearrow \infty$ simulation analysis can become enormously time-consuming.
- The use of *meta-models* to obtain a functional approximation of the (black-box) relationship between the inputs and outputs addresses this issue.

Use of Common Random Numbers (CRNs)

- CRNs are used as a variance-reduction technique for simulation experiments.
- This approach uses the same pseudo-random number stream for each of the design points to subject all scenarios to the same statistical environment.
- Using CRNs induces correlation in the outputs generated by distinct design points [Kleijnen, 1992; Gill et al., 2018], thereby complicating analyses using ordinary least squares (OLS) regression of generalized linear models (GLMs)

Use of Common Random Numbers (CRNs)

- CRNs are used as a variance-reduction technique for simulation experiments.
- This approach uses the same pseudo-random number stream for each of the design points to subject all scenarios to the same statistical environment.
- Using CRNs induces correlation in the outputs generated by distinct design points [Kleijnen, 1992; Gill et al., 2018], thereby complicating analyses using ordinary least squares (OLS) regression of generalized linear models (GLMs)

Use of Common Random Numbers (CRNs)

- CRNs are used as a variance-reduction technique for simulation experiments.
- This approach uses the same pseudo-random number stream for each of the design points to subject all scenarios to the same statistical environment.
- Using CRNs induces correlation in the outputs generated by distinct design points [Kleijnen, 1992; Gill et al., 2018], thereby complicating analyses using ordinary least squares (OLS) regression of generalized linear models (GLMs)

Simulation Output Types

- In combat simulation experiments, output metrics of interest may also be categorical and/or discrete variables, not just continuous.
- Meta-modelling approaches for continuous variables have been extensively covered (see e.g. Chen et al. [2009] for a review of these methods), although meta-modelling with binary, discrete, or count outputs has received very little attention.
  - Meckesheimer et al. [2001] tackle the problem of meta-modelling with piecewise-continuous responses, but their approach does not accommodate the meta-modelling of strictly binary, discrete, or count outputs.
- Due to the variety of output types, meta-modelling with OLS-based approaches [Kleijnen, 1992; Gill et al., 2018] are not appropriate.

Simulation Output Types

- In combat simulation experiments, output metrics of interest may also be categorical and/or discrete variables, not just continuous.
- Meta-modelling approaches for continuous variables have been extensively covered (see e.g. Chen et al. [2009] for a review of these methods), although meta-modelling with binary, discrete, or count outputs has received very little attention.
  - Meckesheimer et al. [2001] tackle the problem of meta-modelling with piecewise-continuous responses, but their approach does not accommodate the meta-modelling of strictly binary, discrete, or count outputs.
- Due to the variety of output types, meta-modelling with OLS-based approaches [Kleijnen, 1992; Gill et al., 2018] are not appropriate.

Simulation Output Types

- In combat simulation experiments, output metrics of interest may also be categorical and/or discrete variables, not just continuous.
- Meta-modelling approaches for continuous variables have been extensively covered (see e.g. Chen et al. [2009] for a review of these methods), although meta-modelling with binary, discrete, or count outputs has received very little attention.
  - Meckesheimer et al. [2001] tackle the problem of meta-modelling with piecewise-continuous responses, but their approach does not accommodate the meta-modelling of strictly binary, discrete, or count outputs.
- Due to the variety of output types, meta-modelling with OLS-based approaches [Kleijnen, 1992; Gill et al., 2018] are not appropriate.

Simulation Output Types

- In combat simulation experiments, output metrics of interest may also be categorical and/or discrete variables, not just continuous.
- Meta-modelling approaches for continuous variables have been extensively covered (see e.g. Chen et al. [2009] for a review of these methods), although meta-modelling with binary, discrete, or count outputs has received very little attention.
  - Meckesheimer et al. [2001] tackle the problem of meta-modelling with piecewise-continuous responses, but their approach does not accommodate the meta-modelling of strictly binary, discrete, or count outputs.
- Due to the variety of output types, meta-modelling with OLS-based approaches [Kleijnen, 1992; Gill et al., 2018] are not appropriate.

Main Contributions

- We discuss a framework for a regression-based meta-modelling of simulation experiments with continuous, binary, and count outputs.
- Specifically, we illustrate the use of **estimated generalized least squares (EGLS)** [Kleijnen, 1992], **finite mixture GLMs** [Wedel and DeSarbo, 1995], and **heteroskedastic binary regression** [Alvarez and Brehm, 1995].
- Our framework also accounts for possible *heterogeneity* and *correlation* induced by the use of CRNs.
- We focus on regression-based approaches as these structures are more interpretable compared to other methods. A regression-based approach also lends itself more easily to sensitivity analyses, design point comparison, and ranking of alternatives.

Main Contributions

- We discuss a framework for a regression-based meta-modelling of simulation experiments with continuous, binary, and count outputs.
- Specifically, we illustrate the use of **estimated generalized least squares (EGLS)** [Kleijnen, 1992], **finite mixture GLMs** [Wedel and DeSarbo, 1995], and **heteroskedastic binary regression** [Alvarez and Brehm, 1995].
- Our framework also accounts for possible *heterogeneity* and *correlation* induced by the use of CRNs.
- We focus on regression-based approaches as these structures are more interpretable compared to other methods. A regression-based approach also lends itself more easily to sensitivity analyses, design point comparison, and ranking of alternatives.

Main Contributions

- We discuss a framework for a regression-based meta-modelling of simulation experiments with continuous, binary, and count outputs.
- Specifically, we illustrate the use of **estimated generalized least squares (EGLS)** [Kleijnen, 1992], **finite mixture GLMs** [Wedel and DeSarbo, 1995], and **heteroskedastic binary regression** [Alvarez and Brehm, 1995].
- Our framework also accounts for possible *heterogeneity* and *correlation* induced by the use of CRNs.
- We focus on regression-based approaches as these structures are more interpretable compared to other methods. A regression-based approach also lends itself more easily to sensitivity analyses, design point comparison, and ranking of alternatives.

Main Contributions

- We discuss a framework for a regression-based meta-modelling of simulation experiments with continuous, binary, and count outputs.
- Specifically, we illustrate the use of **estimated generalized least squares (EGLS)** [Kleijnen, 1992], **finite mixture GLMs** [Wedel and DeSarbo, 1995], and **heteroskedastic binary regression** [Alvarez and Brehm, 1995].
- Our framework also accounts for possible *heterogeneity* and *correlation* induced by the use of CRNs.
- We focus on regression-based approaches as these structures are more interpretable compared to other methods. A regression-based approach also lends itself more easily to sensitivity analyses, design point comparison, and ranking of alternatives.

Presentation Outline

Notation

- $\mathbf{x}_i = (1, x_{i,1}, \ldots, x_{i,L})^\top$ denotes the regressors (incl. higher-order or interaction terms) for each design point $i = 1, \ldots, n$.
- $\mathbf{X}$ is the $n \times (L+1)$ matrix whose ith row is $\mathbf{x}_i^\top$.
- Let $\{w_{i;r}\}_{r=1}^m$ be the m realizations of the (continuous) output variable w_i.
- Let $\bar{\mathbf{w}} = (\bar{w}_1, \ldots, \bar{w}_n)^\top$, where $\bar{w}_i = \frac{1}{m}\sum_{r=1}^m w_{i;r}$, be the vector of simulation output averages.
- $\beta = (\beta_0, \beta_1, \ldots, \beta_L)^\top$ is the vector of unknown regression coefficients.
- $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^\top$ is the vector of error terms.

Notation

- $\mathbf{x}_i = (1, x_{i,1}, \ldots, x_{i,L})^\top$ denotes the regressors (incl. higher-order or interaction terms) for each design point $i = 1, \ldots, n$.
- $\mathbf{X}$ is the $n \times (L+1)$ matrix whose ith row is $\mathbf{x}_i^\top$.
- Let $\{w_{i;r}\}_{r=1}^{m}$ be the m realizations of the (continuous) output variable w_i.
- Let $\bar{\mathbf{w}} = (\bar{w}_1, \ldots, \bar{w}_n)^\top$, where $\bar{w}_i = \frac{1}{m}\sum_{r=1}^{m} w_{i;r}$, be the vector of simulation output averages.
- $\beta = (\beta_0, \beta_1, \ldots, \beta_L)^\top$ is the vector of unknown regression coefficients.
- $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^\top$ is the vector of error terms.

Notation

- $\mathbf{x}_i = (1, x_{i,1}, \ldots, x_{i,L})^\top$ denotes the regressors (incl. higher-order or interaction terms) for each design point $i = 1, \ldots, n$.
- $\mathbf{X}$ is the $n \times (L+1)$ matrix whose ith row is $\mathbf{x}_i^\top$.
- Let $\{w_{i;r}\}_{r=1}^m$ be the m realizations of the (continuous) output variable w_i.
- Let $\bar{\mathbf{w}} = (\bar{w}_1, \ldots, \bar{w}_n)^\top$, where $\bar{w}_i = \frac{1}{m}\sum_{r=1}^m w_{i;r}$, be the vector of simulation output averages.
- $\beta = (\beta_0, \beta_1, \ldots, \beta_L)^\top$ is the vector of unknown regression coefficients.
- $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^\top$ is the vector of error terms.

Notation

- $\mathbf{x}_i = (1, x_{i,1}, \ldots, x_{i,L})^\top$ denotes the regressors (incl. higher-order or interaction terms) for each design point $i = 1, \ldots, n$.
- $\mathbf{X}$ is the $n \times (L+1)$ matrix whose ith row is $\mathbf{x}_i^\top$.
- Let $\{w_{i;r}\}_{r=1}^m$ be the m realizations of the (continuous) output variable w_i.
- Let $\bar{\mathbf{w}} = (\bar{w}_1, \ldots, \bar{w}_n)^\top$, where $\bar{w}_i = \frac{1}{m} \sum_{r=1}^m w_{i;r}$, be the vector of simulation output averages.
- $\beta = (\beta_0, \beta_1, \ldots, \beta_L)^\top$ is the vector of unknown regression coefficients.
- $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^\top$ is the vector of error terms.

Notation

- $\mathbf{x}_i = (1, x_{i,1}, \ldots, x_{i,L})^\top$ denotes the regressors (incl. higher-order or interaction terms) for each design point $i = 1, \ldots, n$.
- $\mathbf{X}$ is the $n \times (L+1)$ matrix whose ith row is $\mathbf{x}_i^\top$.
- Let $\{w_{i;r}\}_{r=1}^m$ be the m realizations of the (continuous) output variable w_i.
- Let $\bar{\mathbf{w}} = (\bar{w}_1, \ldots, \bar{w}_n)^\top$, where $\bar{w}_i = \frac{1}{m}\sum_{r=1}^m w_{i;r}$, be the vector of simulation output averages.
- $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_L)^\top$ is the vector of unknown regression coefficients.
- $\epsilon = (\epsilon_1, \ldots, \epsilon_n)^\top$ is the vector of error terms.

Notation

- $\mathbf{x}_i = (1, x_{i,1}, \ldots, x_{i,L})^\top$ denotes the regressors (incl. higher-order or interaction terms) for each design point $i = 1, \ldots, n$.
- $\mathbf{X}$ is the $n \times (L+1)$ matrix whose ith row is $\mathbf{x}_i^\top$.
- Let $\{w_{i;r}\}_{r=1}^m$ be the m realizations of the (continuous) output variable w_i.
- Let $\bar{\mathbf{w}} = (\bar{w}_1, \ldots, \bar{w}_n)^\top$, where $\bar{w}_i = \frac{1}{m}\sum_{r=1}^m w_{i;r}$, be the vector of simulation output averages.
- $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_L)^\top$ is the vector of unknown regression coefficients.
- $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_n)^\top$ is the vector of error terms.

Generalized Least Squares (GLS)

- We assume the linear input-output relationship

$$w_i = \mathbf{x}_i^\top \beta + \epsilon_i, \qquad \text{where } \mathrm{Var}[\epsilon|X] = \Sigma$$

where $\Sigma = [\sigma_{i,j}]$ is an unknown positive semi-definite $n \times n$ matrix with possibly unequal diagonal entries and nonzero off-diagonal entries.

- The GLS estimator of β is

$$\hat{\beta}^{GLS} = (\mathbf{X}^\top \Sigma^{-1} \mathbf{X})^{-1} \mathbf{X}^\top \Sigma^{-1} \bar{\mathbf{w}},$$

which can be calculated if Σ is known.

- We recover the usual OLS estimator if $\Sigma = \sigma^2 \mathbf{I}$, for some constant $\sigma > 0$.

Generalized Least Squares (GLS)

- We assume the linear input-output relationship

$$w_i = \mathbf{x}_i^\top \beta + \epsilon_i, \qquad \text{where } \mathrm{Var}[\epsilon|X] = \Sigma$$

where $\Sigma = [\sigma_{i,j}]$ is an unknown positive semi-definite $n \times n$ matrix with possibly unequal diagonal entries and nonzero off-diagonal entries.

- The GLS estimator of β is

$$\hat{\beta}^{GLS} = (\mathbf{X}^\top \Sigma^{-1} \mathbf{X})^{-1} \mathbf{X}^\top \Sigma^{-1} \bar{\mathbf{w}},$$

which can be calculated if Σ is known.

- We recover the usual OLS estimator if $\Sigma = \sigma^2 \mathbf{I}$, for some constant $\sigma > 0$.

Generalized Least Squares (GLS)

- We assume the linear input-output relationship

$$w_i = \mathbf{x}_i^\top \beta + \epsilon_i, \qquad \text{where } \mathrm{Var}[\epsilon|X] = \Sigma$$

where $\Sigma = [\sigma_{i,j}]$ is an unknown positive semi-definite $n \times n$ matrix with possibly unequal diagonal entries and nonzero off-diagonal entries.

- The GLS estimator of β is

$$\hat{\beta}^{GLS} = (\mathbf{X}^\top \Sigma^{-1} \mathbf{X})^{-1} \mathbf{X}^\top \Sigma^{-1} \bar{\mathbf{w}},$$

which can be calculated if Σ is known.

- We recover the usual OLS estimator if $\Sigma = \sigma^2 \mathbf{I}$, for some constant $\sigma > 0$.

Estimated Generalized Least Squares (EGLS)

- Given repeated measurements of the output for each i, we can estimate Σ by the sample covariance matrix $\hat{\Sigma} = [\hat{\sigma}_{i,j}]$, where

$$\hat{\sigma}_{i,j} = \frac{1}{m-1}\sum_{r=1}^{m}(w_{i;r} - \bar{w}_i)(w_{j;r} - \bar{w}_j), \qquad i,j = 1,\ldots,n.$$

- Using $\hat{\Sigma}$ in the GLS estimator results to the *estimated* GLS (EGLS) estimator

$$\hat{\beta}^{EGLS} = (\mathbf{X}^\top \tilde{\Sigma}^{-1}\mathbf{X})^{-1}\mathbf{X}^\top \tilde{\Sigma}^{-1}\bar{\mathbf{w}}$$

where $\tilde{\Sigma} = \hat{\Sigma}/m$.

- This estimator requires $m > n$ since $\hat{\Sigma}$ is singular otherwise [Kleijnen, 2015].

Estimated Generalized Least Squares (EGLS)

- Given repeated measurements of the output for each i, we can estimate Σ by the sample covariance matrix $\hat{\Sigma} = [\hat{\sigma}_{i,j}]$, where

$$\hat{\sigma}_{i,j} = \frac{1}{m-1} \sum_{r=1}^{m} (w_{i;r} - \bar{w}_i)(w_{j;r} - \bar{w}_j), \qquad i, j = 1, \ldots, n.$$

- Using $\hat{\Sigma}$ in the GLS estimator results to the *estimated* GLS (EGLS) estimator

$$\hat{\beta}^{EGLS} = (\mathbf{X}^\top \tilde{\Sigma}^{-1} \mathbf{X})^{-1} \mathbf{X}^\top \tilde{\Sigma}^{-1} \bar{\mathbf{w}}$$

where $\tilde{\Sigma} = \hat{\Sigma}/m$.

- This estimator requires $m > n$ since $\hat{\Sigma}$ is singular otherwise [Kleijnen, 2015].

Estimated Generalized Least Squares (EGLS)

- Given repeated measurements of the output for each i, we can estimate Σ by the sample covariance matrix $\hat{\Sigma} = [\hat{\sigma}_{i,j}]$, where

$$\hat{\sigma}_{i,j} = \frac{1}{m-1}\sum_{r=1}^{m}(w_{i;r} - \bar{w}_i)(w_{j;r} - \bar{w}_j), \qquad i, j = 1, \ldots, n.$$

- Using $\hat{\Sigma}$ in the GLS estimator results to the *estimated* GLS (EGLS) estimator

$$\hat{\beta}^{EGLS} = (\mathbf{X}^\top \tilde{\Sigma}^{-1}\mathbf{X})^{-1}\mathbf{X}^\top \tilde{\Sigma}^{-1}\bar{\mathbf{w}}$$

  where $\tilde{\Sigma} = \hat{\Sigma}/m$.

- This estimator requires $m > n$ since $\hat{\Sigma}$ is singular otherwise [Kleijnen, 2015].

Tests of Significance via Jackknifing

- For each $r = 1, \ldots, m$, we calculate the EGLS estimate $\hat{\beta}^{J:(-r)}$ after removing the rth replication.

- For each $\ell = 1, \ldots, L$, calculate m pseudo-values

$$J_{\ell;r} = m\hat{\beta}_\ell^{EGLS} - (m-1)\hat{\beta}_\ell^{J:(-r)}, \qquad r = 1, \ldots, m$$

  and the average $\bar{J}_\ell = \frac{1}{m} J_{\ell;r}$.

- Assuming $J_{\ell,m}$ are i.i.d. normal, a $100(1-\alpha)\%$ CI for β_ℓ is

$$\bar{J}_\ell \pm t_{1-\frac{\alpha}{2},m-1} \times S_\ell^J, \qquad \text{where } S_\ell^J = \sqrt{\frac{\sum_{r=1}^m (J_{\ell;r} - \bar{J}_\ell)^2}{m-1}}.$$

- Jackknifing requires $m - 1 > n$ to ensure that $\hat{\beta}^{J:(-r)}$ is well-defined for each r.

Heterogeneity for Non-Continuous Outputs

- We define *heterogeneity* as the case where there may exist smaller *latent classes* of unique design points where the base and main effects (i.e. regression coefficients) may differ across latent classes.
- This heterogeneity can be modelled using a finite mixture of GLMs [Wedel and DeSarbo, 1995], which can accommodate continuous, binary, and count output types among others.

Heterogeneity for Non-Continuous Outputs

- We define *heterogeneity* as the case where there may exist smaller *latent classes* of unique design points where the base and main effects (i.e. regression coefficients) may differ across latent classes.
- This heterogeneity can be modelled using a finite mixture of GLMs [Wedel and DeSarbo, 1995], which can accommodate continuous, binary, and count output types among others.

Finite Mixture GLMs

- Let S be the (unknown) number of latent classes.
- For each i, define $\mathbf{u}_i = (u_{i,1}, \ldots, u_{i,S})$, where $u_{i,s} = 1$ if design point i belongs to class s.
  - $\mathbf{u}_i$ has a *categorical distribution* with probabilities $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_S)$.
  - Note that $\mathbf{u}_i$ is unobserved.
- Conditional on design point i belonging to class s, w_i has pdf/pmf $f_i^{(s)}(w; \beta_s)$ belonging to the exponential family.
  - $\beta_s = (\beta_{s,0}, \beta_{s,1}, \ldots, \beta_{s,L})^\top$ denotes the regression parameters for latent class s.
  - $\beta = (\beta_1^\top, \ldots, \beta_S^\top)$ is the collection of all regression parameters
- The conditional mean $\mu_i^{(s)} = \mathbb{E}[w_i | \mathbf{x}_i, u_{i,s}]$ is related to $\boldsymbol{\beta}_s$ via an appropriate link function $g(\mu_i^{(s)}) = \mathbf{x}_i^\top \boldsymbol{\beta}_s$.

Finite Mixture GLMs

- Estimates for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are obtained by maximizing the *complete log-likelihood*

$$L_c(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln f_i^{(s)}(w_i; \boldsymbol{\beta}_s) + \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln \alpha_s .$$

- Since the complete likelihood function involves unobserved data, maximum likelihood estimation is achieved using the EM algorithm [Dempster et al., 1977; Wedel and DeSarbo, 1995].

- S must also be estimated; we do so by calculating the log-likelihood, the AIC, and the BIC for a pre-specified set of values for S and select the value which maximizes the likelihood or minimizes the AIC/BIC.

- We use the **flexmix** package in R [Leisch, 2004] to estimate finite mixture GLMs.

Finite Mixture GLMs

- Estimates for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are obtained by maximizing the *complete log-likelihood*

$$L_c(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln f_i^{(s)}(w_i; \boldsymbol{\beta}_s) + \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln \alpha_s.$$

- Since the complete likelihood function involves unobserved data, maximum likelihood estimation is achieved using the EM algorithm [Dempster et al., 1977; Wedel and DeSarbo, 1995].

- S must also be estimated; we do so by calculating the log-likelihood, the AIC, and the BIC for a pre-specified set of values for S and select the value which maximizes the likelihood or minimizes the AIC/BIC.

- We use the **flexmix** package in R [Leisch, 2004] to estimate finite mixture GLMs.

Finite Mixture GLMs

- Estimates for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are obtained by maximizing the *complete log-likelihood*

$$L_c(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln f_i^{(s)}(w_i; \boldsymbol{\beta}_s) + \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln \alpha_s.$$

- Since the complete likelihood function involves unobserved data, maximum likelihood estimation is achieved using the EM algorithm [Dempster et al., 1977; Wedel and DeSarbo, 1995].
- S must also be estimated; we do so by calculating the log-likelihood, the AIC, and the BIC for a pre-specified set of values for S and select the value which maximizes the likelihood or minimizes the AIC/BIC.
- We use the **flexmix** package in R [Leisch, 2004] to estimate finite mixture GLMs.

Finite Mixture GLMs

- Estimates for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are obtained by maximizing the *complete log-likelihood*

$$L_c(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln f_i^{(s)}(w_i; \boldsymbol{\beta}_s) + \sum_{i=1}^{n} \sum_{s=1}^{S} u_{i,s} \ln \alpha_s.$$

- Since the complete likelihood function involves unobserved data, maximum likelihood estimation is achieved using the EM algorithm [Dempster et al., 1977; Wedel and DeSarbo, 1995].
- S must also be estimated; we do so by calculating the log-likelihood, the AIC, and the BIC for a pre-specified set of values for S and select the value which maximizes the likelihood or minimizes the AIC/BIC.
- We use the **flexmix** package in R [Leisch, 2004] to estimate finite mixture GLMs.

Illustration: Heteroskedastic Probit Regression

- Suppose we observe $w_i = \mathbf{1}(y_i^* > 0)$, where $\mathbf{1}(\cdot)$ is the indicator function and y_i^* is a latent variable of the form $y_i^* = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i$.

- Assuming $\epsilon_i \sim N(0, \sigma_i^2)$, reflecting heteroskedasticity in the latent error, the *heteroskedastic* probit model is specified as

$$\Phi^{-1}(p_i) = \frac{\mathbf{x}_i^\top \beta}{\sigma_i}, \qquad h(\sigma_i) = h(1) + \mathbf{z}_i^\top \gamma,$$

  where $p_i = \mathbb{P}(w_i = 1 | \mathbf{x}_i) = \mathbb{E}[w_i | \mathbf{x}_i]$.

- The scale model $h(\sigma_i) = h(1) + \mathbf{z}_i^\top \gamma$ consists of:
  1. A scale link function $h(\cdot)$ (usually the log, square root, or identity function)
  2. A vector of regressors $\mathbf{z}_i$ (not necessary equal to $\mathbf{x}_i$)
  3. A vector of coefficients γ.

Illustration: Heteroskedastic Probit Regression

- Suppose we observe $w_i = \mathbf{1}(y_i^* > 0)$, where $\mathbf{1}(\cdot)$ is the indicator function and y_i^* is a latent variable of the form $y_i^* = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i$.

- Assuming $\epsilon_i \sim N(0, \sigma_i^2)$, reflecting heteroskedasticity in the latent error, the *heteroskedastic* probit model is specified as

$$\Phi^{-1}(p_i) = \frac{\mathbf{x}_i^\top \boldsymbol{\beta}}{\sigma_i}, \qquad h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma},$$

where $p_i = \mathbb{P}(w_i = 1|\mathbf{x}_i) = \mathbb{E}[w_i|\mathbf{x}_i]$.

- The scale model $h(\sigma_i) = h(1) + \mathbf{z}_i^\top \gamma$ consists of:
  1. A scale link function $h(\cdot)$ (usually the log, square root, or identity function)
  2. A vector of regressors $\mathbf{z}_i$ (not necessary equal to $\mathbf{x}_i$)
  3. A vector of coefficients γ.

Illustration: Heteroskedastic Probit Regression

- Suppose we observe $w_i = \mathbf{1}(y_i^* > 0)$, where $\mathbf{1}(\cdot)$ is the indicator function and y_i^* is a latent variable of the form $y_i^* = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i$.

- Assuming $\epsilon_i \sim N(0, \sigma_i^2)$, reflecting heteroskedasticity in the latent error, the *heteroskedastic* probit model is specified as

$$\Phi^{-1}(p_i) = \frac{\mathbf{x}_i^\top \boldsymbol{\beta}}{\sigma_i}, \qquad h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma},$$

  where $p_i = \mathbb{P}(w_i = 1 | \mathbf{x}_i) = \mathbb{E}[w_i | \mathbf{x}_i]$.

- The scale model $h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma}$ consists of:
  1. A scale link function $h(\cdot)$ (usually the log, square root, or identity function)
  2. A vector of regressors $\mathbf{z}_i$ (not necessary equal to $\mathbf{x}_i$)
  3. A vector of coefficients γ.

Illustration: Heteroskedastic Probit Regression

- Suppose we observe $w_i = \mathbf{1}(y_i^* > 0)$, where $\mathbf{1}(\cdot)$ is the indicator function and y_i^* is a latent variable of the form $y_i^* = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i$.
- Assuming $\epsilon_i \sim N(0, \sigma_i^2)$, reflecting heteroskedasticity in the latent error, the *heteroskedastic* probit model is specified as

$$\Phi^{-1}(p_i) = \frac{\mathbf{x}_i^\top \boldsymbol{\beta}}{\sigma_i}, \qquad h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma},$$

where $p_i = \mathbb{P}(w_i = 1 | \mathbf{x}_i) = \mathbb{E}[w_i | \mathbf{x}_i]$.

- The scale model $h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma}$ consists of:
  1. A scale link function $h(\cdot)$ (usually the log, square root, or identity function)
  2. A vector of regressors $\mathbf{z}_i$ (not necessary equal to $\mathbf{x}_i$)
  3. A vector of coefficients γ.

Illustration: Heteroskedastic Probit Regression

- Suppose we observe $w_i = \mathbf{1}(y_i^* > 0)$, where $\mathbf{1}(\cdot)$ is the indicator function and y_i^* is a latent variable of the form $y_i^* = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i$.

- Assuming $\epsilon_i \sim N(0, \sigma_i^2)$, reflecting heteroskedasticity in the latent error, the *heteroskedastic* probit model is specified as

$$\Phi^{-1}(p_i) = \frac{\mathbf{x}_i^\top \boldsymbol{\beta}}{\sigma_i}, \qquad h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma},$$

  where $p_i = \mathbb{P}(w_i = 1 | \mathbf{x}_i) = \mathbb{E}[w_i | \mathbf{x}_i]$.

- The scale model $h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma}$ consists of:
  1. A scale link function $h(\cdot)$ (usually the log, square root, or identity function)
  2. A vector of regressors $\mathbf{z}_i$ (not necessary equal to $\mathbf{x}_i$)
  3. A vector of coefficients γ.

Illustration: Heteroskedastic Probit Regression

- Suppose we observe $w_i = \mathbf{1}(y_i^* > 0)$, where $\mathbf{1}(\cdot)$ is the indicator function and y_i^* is a latent variable of the form $y_i^* = \mathbf{x}_i^\top \boldsymbol{\beta} + \epsilon_i$.
- Assuming $\epsilon_i \sim N(0, \sigma_i^2)$, reflecting heteroskedasticity in the latent error, the *heteroskedastic* probit model is specified as

$$\Phi^{-1}(p_i) = \frac{\mathbf{x}_i^\top \boldsymbol{\beta}}{\sigma_i}, \qquad h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma},$$

  where $p_i = \mathbb{P}(w_i = 1 | \mathbf{x}_i) = \mathbb{E}[w_i | \mathbf{x}_i]$.
- The scale model $h(\sigma_i) = h(1) + \mathbf{z}_i^\top \boldsymbol{\gamma}$ consists of:
  1. A scale link function $h(\cdot)$ (usually the log, square root, or identity function)
  2. A vector of regressors $\mathbf{z}_i$ (not necessary equal to $\mathbf{x}_i$)
  3. A vector of coefficients $\boldsymbol{\gamma}$.

Heteroskedastic Binary Regression

- Aside from the probit link $g(p) = \Phi^{-1}(p)$, we can also use the logit link $g(p) = \ln(p/(1-p))$. See Koenker and Yoon [2009] for other appropriate link functions.
- The **glmx** package in R [Zeileis et al., 2015] provides a suite of functions for fitting heteroskedastic binary regression models.

Heteroskedastic Binary Regression

- Aside from the probit link $g(p) = \Phi^{-1}(p)$, we can also use the logit link $g(p) = \ln(p/(1-p))$. See Koenker and Yoon [2009] for other appropriate link functions.
- The **glmx** package in R [Zeileis et al., 2015] provides a suite of functions for fitting heteroskedastic binary regression models.

Presentation Outline

Introductory Notes and Assumptions

- We consider the main effects (intercept term plus first-order terms)
- We assume a 5% level of significance for all hypothesis tests/confidence intervals. We focus on individual tests of significance.
  - Variables/estimates marked with * are significant at $\alpha = 5\%$.
- All statistical analyses were performed using established utilities/packages in R. Computation times for all runs are negligible.

Introductory Notes and Assumptions

- We consider the main effects (intercept term plus first-order terms)
- We assume a 5% level of significance for all hypothesis tests/confidence intervals. We focus on individual tests of significance.
  - Variables/estimates marked with * are significant at $\alpha = 5\%$.
- All statistical analyses were performed using established utilities/packages in $\mathbb{R}$. Computation times for all runs are negligible.

Introductory Notes and Assumptions

- We consider the main effects (intercept term plus first-order terms)
- We assume a 5% level of significance for all hypothesis tests/confidence intervals. We focus on individual tests of significance.
  - Variables/estimates marked with * are significant at $\alpha = 5\%$.
- All statistical analyses were performed using established utilities/packages in $\mathbb{R}$. Computation times for all runs are negligible.

Introductory Notes and Assumptions

- We consider the main effects (intercept term plus first-order terms)
- We assume a 5% level of significance for all hypothesis tests/confidence intervals. We focus on individual tests of significance.
  - Variables/estimates marked with * are significant at $\alpha = 5\%$.
- All statistical analyses were performed using established utilities/packages in `R`. Computation times for all runs are negligible.

Data Description

- Outputs: *mission success* (binary), *number of red infantry defeated* (count), *number of red vehicles destroyed* (count)
- Inputs: *Option* ("A", "B", "C", or "D"), *F1* ("Direct" or "Indirect"), *F2* ("25" or "75"), and *F3* ("Low", "Medium", or "High")
- There are $n = 48$ distinct input combinations, each of which is repeated $m = 200$ times.
- For illustrative purposes we consider the *number of red infantry defeated* a continuous output rather than a count output since it has a relatively high average value, good dispersion, and an approximately symmetric distribution.

Data Description

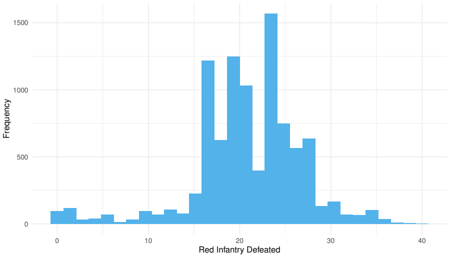
- Outputs: *mission success* (binary), *number of red infantry defeated* (count), *number of red vehicles destroyed* (count)
- Inputs: *Option* (“A”, “B”, “C”, or “D”), *F1* (“Direct” or “Indirect”), *F2* (“25” or “75”), and *F3* (“Low”, “Medium”, or “High”)
- There are $n = 48$ distinct input combinations, each of which is repeated $m = 200$ times.
- For illustrative purposes we consider the *number of red infantry defeated* a continuous output rather than a count output since it has a relatively high average value, good dispersion, and an approximately symmetric distribution.

Data Description

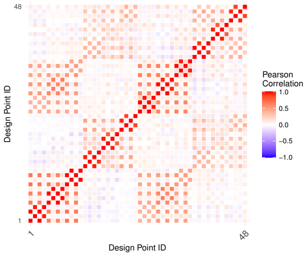
- Outputs: *mission success* (binary), *number of red infantry defeated* (count), *number of red vehicles destroyed* (count)
- Inputs: *Option* ("A", "B", "C", or "D"), *F1* ("Direct" or "Indirect"), *F2* ("25" or "75"), and *F3* ("Low", "Medium", or "High")
- There are $n = 48$ distinct input combinations, each of which is repeated $m = 200$ times.
- For illustrative purposes we consider the *number of red infantry defeated* a continuous output rather than a count output since it has a relatively high average value, good dispersion, and an approximately symmetric distribution.

Data Description

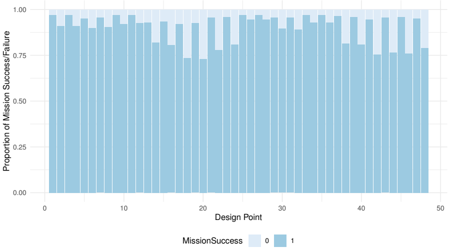
- Outputs: *mission success* (binary), *number of red infantry defeated* (count), *number of red vehicles destroyed* (count)
- Inputs: *Option* ("A", "B", "C", or "D"), *F1* ("Direct" or "Indirect"), *F2* ("25" or "75"), and *F3* ("Low", "Medium", or "High")
- There are $n = 48$ distinct input combinations, each of which is repeated $m = 200$ times.
- For illustrative purposes we consider the *number of red infantry defeated* a continuous output rather than a count output since it has a relatively high average value, good dispersion, and an approximately symmetric distribution.



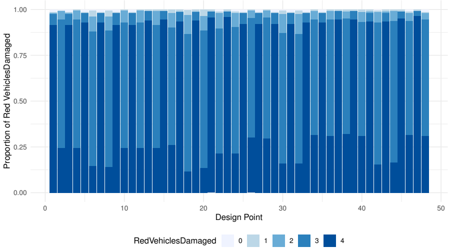
(a) Histogram of RID



(b) Correlation heat map for RID across design points



(a) MS by design point



(b) RVD by design point

Table: OLS, EGLS, and finite mixture Gaussian regression estimates for number of red infantry defeated.

	OLS (HC se)		EGLS Est.		EGLS 95% Jackknifed CI		FM Gaussian Regression Class 1		Class 2		Class 3	
Intercept	23.340	*	23.454	*	23.039	24.489	22.981	*	22.405	*	23.234	*
Option B	-1.029	*	-1.255	*	-2.338	-0.789	-2.607	*	0.385	*	-0.952	*
Option C	-0.105		-0.035		-0.795	0.278	0.177	*	-0.069		1.140	*
Option D	-1.119	*	-1.193	*	-2.263	-0.744	-1.833	*	-0.194	*	-1.066	*
F1 Indirect	-2.839	*	-2.933	*	-3.808	-2.426	-1.779	*	-1.891	*	-2.789	*
F2 75	0.009		0.007		-0.001	0.070	-0.064	*	0.007		-0.228	*
F3 Low	-0.816	*	-0.647	*	-1.044	-0.166	-1.408	*	-0.191	*	-1.158	*
F3 Medium	-0.280	*	-0.309	*	-0.820	-0.071	-0.349	*	0.075	*	-0.738	*
Log-Lik	-30433.70						54.68					
AIC	60885.41						-51.37					
BIC	60949.94						2.89					

Note: Breusch-Pagan test on OLS: p-value $< 2.2e^{-16}$; OLS $R^2 = 6.645\%$; Rao's lack-of-fit F test for ELGS: test statistic $= 1.69$, p-value $= 0.0041$.

Table: Homoskedastic and heteroskedastic binary regression (with probit and logit link functions) and finite mixture logistic regression results for mission success.

	Homoskedastic				Heteroskedastic				FM Logistic Regression			
	Probit		Logit		Probit		Logit		Class 1		Class 2	
Intercept	1.2041	*	2.0124	*	0.8818	*	1.4441	*	0.8785	*	1.0220	*
Option B	-0.8926	*	-1.4877	*	-0.4215		-0.7117		-0.2800	*	-0.4215	*
Option C	-0.1523		-0.2569		-0.0361		-0.0623		0.0320		-0.3944	*
Option D	-0.7843	*	-1.3804	*	-0.2456		-0.4194		-0.2879	*	-0.2895	
F1 Indirect	-0.8402	*	-1.3954	*	-0.7586	*	-1.2322	*	-0.2872	*	-0.3758	*
F2 75	-0.0772		-0.1370		-0.0441		-0.0787		-0.0199		-0.0321	
F3 Low	-0.2634	*	-0.4357	*	-0.0961		-0.1632		-0.0934	*	0.0280	
F3 Medium	0.0352		0.0616		0.0038		0.0054		0.0010		0.0877	
Log-Lik	-713.01		-712.38		-703.60		-703.60		-744.37			
AIC	1442.00		1440.80		1437.30		1437.27		1526.75			
BIC	1482.73		1481.47		1513.65		1513.62		1623.46			
LR Test p-val.					0.0091		0.0145					
AR	0.7114		0.7114		0.9024		0.9024					
Bal. AR	0.6716		0.6716		0.5000		0.5000					

Table: Results of the Poisson (P), quasi-Poisson (QP), Conway-Maxwell Poisson (CMP), negative binomial (NB), and Poisson-lognormal mixture (PLN) regression for number of red vehicles damaged.

	P		QP		CMP		NB		PLN	
Intercept	1.3621	*	1.3621	*	15.8125	*	1.3621	*	1.3723	*
Option B	0.0023		0.0023		0.0219		0.0023		-0.0053	
Option C	0.0084		0.0084		0.0813		0.0084		0.0081	
Option D	0.0184		0.0184	*	0.1765	*	0.0184		0.0111	
F1 Indirect	-0.2186	*	-0.2186	*	-2.0633	*	-0.2186	*	-0.2228	*
F2 75	0.0007		0.0007		0.0064		0.0007		0.0003	
F3 Low	-0.0288	*	-0.0288	*	-0.2760	*	-0.0288	*	-0.0274	*
F3 Medium	-0.0018		-0.0018		-0.0165		-0.0018		0.0005	
Log-lik	-15488.30				-8752.54		-15488.33		-15299.16	
AIC	30993.00				17523.08		30995.00			
BIC	31049.97				17587.60		31059.18		-15340.40	
Dispersion	0.0874		0.0875		0.0932					

Note: FM Poisson regression returns $S = 1$ as optimal $\Rightarrow$ usual Poisson regression

Presentation Outline

Concluding Remarks

- Our analysis using finite mixture GLMs has shown that there is significant heterogeneity in the base and main effects to the mean of the continuous and binary outputs.
- The results arising from the finite mixture GLMs, however, must be appraised via a qualitative assessment of the design points that are clustered together in latent classes to determine why such a clustering was derived from the data.
- As it is likely that there is some degree of under- or overdispersion in the output metrics, especially for count data, simulation meta-modelling via a joint modelling of the mean and dispersion [see e.g. Smyth, 1989]

Concluding Remarks

- Our analysis using finite mixture GLMs has shown that there is significant heterogeneity in the base and main effects to the mean of the continuous and binary outputs.
- The results arising from the finite mixture GLMs, however, must be appraised via a qualitative assessment of the design points that are clustered together in latent classes to determine why such a clustering was derived from the data.
- As it is likely that there is some degree of under- or overdispersion in the output metrics, especially for count data, simulation meta-modelling via a joint modelling of the mean and dispersion [see e.g. Smyth, 1989]

Concluding Remarks

- Our analysis using finite mixture GLMs has shown that there is significant heterogeneity in the base and main effects to the mean of the continuous and binary outputs.
- The results arising from the finite mixture GLMs, however, must be appraised via a qualitative assessment of the design points that are clustered together in latent classes to determine why such a clustering was derived from the data.
- As it is likely that there is some degree of under- or overdispersion in the output metrics, especially for count data, simulation meta-modelling via a joint modelling of the mean and dispersion [see e.g. Smyth, 1989]

Concluding Remarks

- For count data with a known upper limit (as is the case for most defence-related applications), approaches related to truncated or censored count data regression may be considered [see e.g. Cameron and Trivedi, 2013, Sections 4.3-4.4].
- Furthermore, additional work must be done to properly characterize the extent to which CRNs induce “correlation” in binary and discrete output metrics.

Concluding Remarks

- For count data with a known upper limit (as is the case for most defence-related applications), approaches related to truncated or censored count data regression may be considered [see e.g. Cameron and Trivedi, 2013, Sections 4.3-4.4].
- Furthermore, additional work must be done to properly characterize the extent to which CRNs induce "correlation" in binary and discrete output metrics.

Acknowledgments

- The Commonwealth of Australia (represented by the Defence Science and Technology Group) supported this research through a Defence Science Partnership agreement.
- The authors also thank Dr. Andrew Gill for fruitful discussions about the data set and the statistical methodology employed in this paper.

Acknowledgments

- The Commonwealth of Australia (represented by the Defence Science and Technology Group) supported this research through a Defence Science Partnership agreement.
- The authors also thank Dr. Andrew Gill for fruitful discussions about the data set and the statistical methodology employed in this paper.

Thank you very much for your attention!

References

Alvarez, R. M. and Brehm, J. (1995). American ambivalence towars abortion policy: development of a heteroskedastic probit model of competing values. *American Journal of Political Science*, 39(4):1055–1082.

Cameron, A. C. and Trivedi, P. K. (2013). *Regression Analysis of Count Data*. Cambridge University Press, New York, 2nd edition.

Chen, V. C. P., Tsui, K.-L., Barton, R. R., and Meckesheimer, M. (2009). A review on design, modeling and applications of computer experiments. *IIE Transactions*, 38(4):273–291.

Dempster, A. P., Laird, N. M., and Rubin, R. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, B*, 39:1–38.

Gill, A., Grieger, D., Wong, M., and Chau, W. (2018). Combat simulation analytics: Regression analysis, multiple comparisons and ranking sensitivity. In Rabe, M., Juan, A. A., Mustafee, N., Skoogh, A., Jain, S., and Johansson, B., editors, *Proceedings of the 2018 Winter Simulation Conference*, page 3789–3800. IEEE Press.

References (contd.)

Kleijnen, J. P. C. (1992). Regression metamodels for simulation with common random numbers: comparison of validation tests and confidence intervals. *Management Science*, 38(8):1164–1185.

Kleijnen, J. P. C. (2015). *Design and Analysis of Simulation Experiments*. Springer International Publishing, Switzerland, 2nd edition.

Koenker, R. and Yoon, J. (2009). Parametric links for binary choice models: a Fisherian-Bayesian colloquy. *Journal of Econometrics*, 152:120–130.

Leisch, F. (2004). FlexMix: A general framework for finite mixture models and latent class regression in R. *Journal of Statistical Software*, 11(8):1–18.

Meckesheimer, M., Barton, R. R., Simpson, T. W., Limayem, F., and Yannou, B. (2001). Metamodeling of combined discrete/continuous responses. *AIAA Journal*, 39:1950–1959.

Smyth, G. K. (1989). Generalized linear models with varying dispersion. *Journal of the Royal Statistical Society, Series B (Methodological)*, 51(1):47–60.

References (contd.)

Wedel, M. and DeSarbo, W. S. (1995). A mixture likelihood approach for generalized linear models. *Journal of Classification*, 12:21–55.

Zeileis, A., Koenker, R., and Doebler, P. (2015). *glmx: Generalized Linear Models Extended*. R package version 0.1-1.