STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **x**, Month 202x, pp 1–28.
Published online in International Academic Press (www.IAPress.org)

Development of Penalized Estimation for Conway–Maxwell–Poisson Regression under Multicollinearity and High Dimensionality

Ibrahim M. Taha [1,2,*], Mohamed R. Abonazel [1], Amany M. Mousa [1]

[1] *Department of Applied Statistics and Econometrics, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza 12613, Egypt*
[2] *Department of Mathematics, Statistics and Insurance, Faculty of Management Sciences, Sadat Academy for Management Sciences, Corniche El-Nil, Maadi, Cairo 2222, Egypt*

Abstract Modern count data applications are increasingly characterized by dispersion, high dimensionality, and strong predictor dependence, conditions under which maximum likelihood estimation for Conway–Maxwell–Poisson (COM-Poisson) regression becomes unstable and prone to severe overfitting. The purpose of this study is to develop an estimation framework for COM-Poisson regression that remains stable when the predictor dimension is large relative to the sample size and when predictors are strongly collinear, and that selects predictors rather than shrinking all of them. Such a framework is needed because the shrinkage estimators currently available for this model retain the full predictor set, so that a sparse solution cannot be obtained precisely in the settings where sparsity is most valuable. We propose an L_1-regularized framework for COM-Poisson regression, to our knowledge the first of its kind, by introducing COMP-LASSO and COMP-ALASSO estimators based on the penalized COM-Poisson likelihood. The proposed estimators are computed using an iteratively reweighted coordinate descent algorithm that explicitly incorporates the dispersion structure of the COM-Poisson model. Simulation experiments across a wide range of dimensionality, multicollinearity, and dispersion settings show that penalized estimation consistently outperforms maximum likelihood in both prediction accuracy and variable selection, with the largest gains occurring in severely ill-conditioned problems. Empirical applications in clinical research, consumer finance, and community ecology demonstrate that the proposed methods produce substantially sparser and more accurate models while avoiding the instability and overfitting behaviour of maximum likelihood estimation. These results establish sparse penalized estimation as an effective and practical framework for high-dimensional dispersed count regression.

Keywords Conway–Maxwell–Poisson regression, LASSO, penalized likelihood, coordinate descent, multicollinearity, variable selection, high-dimensional inference.

AMS 2010 subject classifications 62J07, 62J12

DOI: 10.19139/soic-2310-5070-4220

1. Introduction

Count data have been observed in almost every field of empirical science (disease incidence, traffic accidents, genetic mutations and consumer transactions, among others) and, for a long time, the Poisson regression model has been the standard one for count responses [12]. Its main drawback is that it assumes that the conditional variance equals the conditional mean, a condition that is often not satisfied by real data. Overdispersion is almost ubiquitous in ecological, epidemiological and economic applications, and, less discussed, underdispersion occurs where regularity mechanisms limit variability [23]. The negative binomial model is able to capture overdispersion but not underdispersion, and neither model covers both regimes in one coherent model.

*Correspondence to: Ibrahim M. Taha (Email: ibrahimaboalazm@gmail.com), Department of Applied Statistics and Econometrics, Faculty of Graduate Studies for Statistical Research, Cairo University, Giza 12613, Egypt.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

The Conway–Maxwell–Poisson distribution overcomes this by generalizing the Poisson distribution with two parameters, embedding the Poisson, geometric and Bernoulli as special cases [14, 36]. Its dispersion parameter controls continuously the variance-to-mean relationship, such that equidispersion, overdispersion and underdispersion can all be reached. Sellers and Shmueli [35] formalized the regression extension and derived its asymptotic properties, and since then, there has been a growing literature in which it has been applied to road accident modelling [28], marketing analytics [34] and biomedical research [25]. Estimation, however, has developed unevenly. Considerable effort has gone into making count regression robust or efficient under other departures from ideal conditions, for instance the robust M-estimation approach of Youssef, Abonazel and Ahmed [41] for Poisson panel data with fixed effects, or the comparison of robust and non-robust count estimators under overdispersion, zero inflation and outliers by Abonazel, El-sayed and Saber [3], but for the COM-Poisson model itself estimation has remained almost entirely confined to maximum likelihood, which is adequate only under favourable design conditions. When the predictor dimension approaches the sample size, as in genomics, environmental monitoring and high-frequency marketing data, the information matrix approaches singularity and coefficient estimates exhibit high variability and substantial overfitting [10, 22]. Multicollinearity aggravates this: strong linear dependence among predictors inflates variances and destabilizes inference even at moderate dimensionality [16], an effect shown by Taha [37] to be acute in generalized linear models.

The principled treatment is penalized estimation. Hoerl and Kennard [24] introduced ridge regression to shrink the coefficients toward zero and stabilize high-variance estimates. For COMP regression, Abonazel et al. [5] proposed a two-parameter Liu estimator combining ridge and Liu shrinkage, with uniformly improved MSE over maximum likelihood, and Abonazel [1] extended this with modified Liu estimators having better finite-sample properties. Ridge estimation for the same model has been developed in parallel by Abonazel et al. [4], who construct ridge parameters for COM-Poisson under highly correlated predictors and report gains over maximum likelihood in both dispersion regimes. Parallel Liu-type developments exist for neighbouring count models, such as the estimator of Alrweili [8] for Poisson-inverse Gaussian regression. These are valuable contributions, but they shrink without selecting: ridge-type estimators retain the full predictor set and therefore cannot deliver a sparse model in high-dimensional settings.

The LASSO [38] fills this gap precisely, estimating and selecting simultaneously by setting irrelevant coefficients exactly to zero, and its coordinate descent implementation for generalized linear models [19] makes it tractable in high dimensions. The Adaptive LASSO [43] imposes coefficient-specific penalties that improve selection consistency under multicollinearity [42, 9]. The gap this paper addresses is therefore narrow and specific. Penalized estimation for count responses has been developed only for the standard Poisson framework, through adaptive elastic net variants [6] and penalized spline estimators [2], neither of which can represent underdispersion. Variable selection for count models has also been pursued outside the penalized-likelihood framework, for example by metaheuristic search [7], but again within the Poisson family. For the COM-Poisson model itself, penalized estimation exists only in ridge and Liu form [1, 5, 4], which shrinks without selecting, and the same is true of the corresponding developments for the Poisson model [30]. Neither line yields a sparse estimator for a model that accommodates both dispersion regimes. ElSheikh, Abonazel, and Ali [15] survey high-dimensional penalized regression and identify dispersed count models as an open problem. L_1-penalized estimation for COM-Poisson regression appears not to have been investigated previously.

L_1 is not the only penalty available, and the scope adopted here should be stated. The elastic net and the nonconvex families such as SCAD and MCP are well established for linear and generalized linear models [21, 22], and adaptive elastic net variants have been applied to Poisson regression [6]. They are not developed here, for two reasons, of which the first is one of order: the L_1 and adaptive L_1 cases are the canonical entry point to sparse estimation, and the behaviour of the COM-Poisson likelihood under a single convex penalty must be understood before mixed or nonconvex penalties can be assessed against it. The second is computational, since every fit already nests a coordinate descent within an iteratively reweighted loop within a profile search over ν; a second mixing parameter, or a penalty whose stationary points are not unique, multiplies that cost and raises optimization questions distinct from the statistical ones addressed here. Elastic-net mixing is identified as future work in Section 6, and the framework of Section 3 admits substitution of the penalty term without alteration of the surrounding algorithm.

The research method proceeds as follows. We attach an L_1 penalty directly to the COM-Poisson log-likelihood rather than to a Poisson surrogate, and maximize the resulting objective by a two-level scheme: an outer iteratively reweighted least squares loop that recomputes the COMP conditional mean and variance at the current iterate and forms a working response, and an inner cyclic coordinate descent that updates each coefficient by soft-thresholding the weighted partial residual. Because the working weights are the COMP variances rather than Poisson ones, the dispersion structure of the model is carried through every estimation step. The dispersion parameter itself is left unpenalized and updated by profile likelihood at each outer iteration, and the penalty strength is chosen by five-fold cross-validation along a warm-started grid. The adaptive variant replaces the uniform threshold by coefficient-specific weights derived from a preliminary ridge fit. Evidence is then assembled from two sources: a factorial Monte Carlo experiment crossing sample size, predictor dimension, correlation and dispersion, evaluated on an independent test set; and three empirical applications chosen to span severe collinearity at low dimension, large samples with moderate collinearity, and genuine high dimensionality with $p \gg n$.

The paper makes four contributions. It introduces COMP-LASSO and COMP-ALASSO, to our knowledge the first L_1-penalized estimators formulated directly from the COM-Poisson likelihood, and therefore the first estimators for this model that perform variable selection rather than shrinkage alone. It develops an iteratively reweighted coordinate descent algorithm whose working weights are the COM-Poisson variances, so that the coefficients and the dispersion parameter are estimated jointly rather than treating dispersion as fixed or Poisson-like. It quantifies, across a factorial simulation design, how the benefit of penalization varies with dimensionality, collinearity and the dispersion regime, and identifies the conditions under which the adaptive variant improves upon the uniform penalty. Finally, it shows empirically that unpenalized COM-Poisson estimation can misstate not only which predictors matter but the dispersion of the response itself, and that the consequences differ according to whether the difficulty arises from collinearity at low dimension or from a predictor set comparable in size to the sample.

The rest of the paper is organized as follows. The theoretical framework is developed in Section 2. The proposed estimators and algorithms are presented in Section 3. In Section 4 simulation results are reported. Empirical applications are in Section 5. Section 6 offers the conclusion.

2. Theoretical Framework

2.1. Conway–Maxwell–Poisson Distribution

A discrete random variable Y is said to be distributed according to the Conway–Maxwell–Poisson distribution [14, 36] if its probability mass function has the form

$$P(Y = y \mid \lambda, \nu) = \frac{\lambda^y}{(y!)^\nu \, Z(\lambda, \nu)}, \qquad y = 0, 1, 2, \ldots, \tag{1}$$

where the rate parameter is $\lambda > 0$, the dispersion parameter is $\nu \geq 0$, and $Z(\lambda, \nu) = \sum_{j=0}^{\infty} \lambda^j / (j!)^\nu$ is the normalizing constant. The behaviour of ν is the defining behaviour of the model: $\nu = 1$ gives the standard Poisson; $\nu < 1$ gives overdispersion; and $\nu > 1$ gives underdispersion. It is a single parameter that continuously varies between the three cases: geometric $\nu \to 0$, Bernoulli $\nu \to \infty$, with all three nested in between, as stated by Sellers and Shmueli [35]. Moments are not analytically available due to the absence of a closed form of $Z(\lambda, \nu)$. Derivatives of the normalizing function are used to compute the conditional mean and variance,

$$E(Y) = \lambda \frac{\partial \log Z(\lambda, \nu)}{\partial \lambda}, \qquad \mathrm{Var}(Y) = \lambda \frac{\partial E(Y)}{\partial \lambda}. \tag{2}$$

Sellers and Shmueli [35] give that the following are asymptotic approximations for practical computation,

$$E(Y) \approx \lambda^{1/\nu} - \frac{\nu - 1}{2\nu}, \qquad \mathrm{Var}(Y) \approx \frac{\lambda^{1/\nu}}{\nu}, \tag{3}$$

which are accurate if $\lambda^{1/\nu}$ is not too small. These approximations reduce the cost of evaluating the conditional moments, but they do not remove the need to evaluate $Z(\lambda, \nu)$ itself, which enters the log-likelihood directly and

admits no closed form. Every likelihood evaluation therefore requires the series to be truncated. The truncation rule, the regime in which (3) is used in preference to direct summation, and the associated error control are set out in Section 3.6.

2.2. COMP Regression Formulation

Suppose that there are n independent observations (y_i, x_i), with $x_i = (x_{i1}, \ldots, x_{ip})^\top$ a p-dimensional covariate vector; then the COMP regression model is defined as

$$y_i \mid x_i \sim \mathrm{COMP}(\lambda_i, \nu), \tag{4}$$

with a log-linear relation between the rate parameter and the covariates,

$$\log(\lambda_i) = x_i^\top \beta = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}, \tag{5}$$

so that $\lambda_i = \exp(x_i^\top \beta)$. This formulation, due to Sellers and Shmueli [35], ensures that predicted rates are positive for all values of the coefficients, and that multiplicative interpretations are possible: a unit increase in x_j corresponds to a multiplicative change of $\exp(\beta_j)$ in λ. Following the tradition of flexibility versus tractability, the dispersion parameter ν is estimated globally as a scalar nuisance. The factorial term in the denominator carries the exponent ν, which is often taken to place the model outside the exponential family. It does not: as shown in Section 3.4, (1) is a two-parameter canonical exponential family, and for fixed ν a one-parameter family in $\eta = \log \lambda$. What distinguishes the COM-Poisson case is that its cumulant function $\log Z(\lambda, \nu)$ has no closed form, so the moments and the likelihood must be evaluated numerically. This complicates theory and computation, but it is also the source of the dispersion flexibility of the model.

2.3. Maximum Likelihood Estimation

The log-likelihood for the whole sample is

$$\ell(\beta, \nu) = \sum_{i=1}^{n} \Big[y_i x_i^\top \beta - \nu \log(y_i!) - \log Z\big(\exp(x_i^\top \beta), \nu\big) \Big]. \tag{6}$$

The maximum likelihood estimates $(\hat\beta, \hat\nu)$ are such that

$$(\hat\beta, \hat\nu) = \arg\max_{\beta,\, \nu} \ \ell(\beta, \nu). \tag{7}$$

The score with respect to β has a familiar residual form,

$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^{n} x_i \, (y_i - \mu_i), \tag{8}$$

where $\mu_i = E(y_i \mid x_i)$ is the COMP conditional mean. This is the same as the GLM scoring structure described by Wedderburn [40] and suggests the use of iteratively reweighted least squares as an outer optimization loop. Numerical differentiation of the normalizing constant is needed for the score with respect to ν,

$$\frac{\partial \ell}{\partial \nu} = -\sum_{i=1}^{n} \Big[\log(y_i!) + \frac{\partial \log Z(\lambda_i, \nu)}{\partial \nu} \Big]. \tag{9}$$

Given regularity conditions, $\hat\beta$ is consistent and asymptotically normal,

$$\sqrt{n}\,(\hat\beta - \beta^*) \xrightarrow{d} N\big(0,\, I^{-1}(\theta_0)\big), \tag{10}$$

where $I(\theta)$ is the Fisher information matrix and θ_0 is the true value of the parameter vector [39]. But these asymptotic guarantees are for fixed p as $n \to \infty$ and hence do not hold exactly in the setting that is of interest in this paper.

2.4. Breakdown Under High Dimensionality and Multicollinearity

As p gets close to or larger than n, the Fisher information matrix $I(\theta) \propto X^\top W X$, with W the diagonal matrix of COMP working weights, becomes rank deficient in the same way as the design matrix. Eigenvalues tend to bunch up around zero, condition numbers tend to blow up, and the coefficient estimates from the matrix inversion have variances orders of magnitude larger than in well-conditioned problems [21]. As a consequence, the resulting estimates may become highly sensitive to small perturbations in the data.

The same pathology occurs with multicollinearity, but through a different mechanism. As demonstrated by Farrar and Glauber [16], even moderate relationships between the predictors tend to cause the design matrix to become singular. The practical consequence is inflated estimator variance and severe overfitting, as confirmed in generalized linear model settings by Taha [37]. This is precisely what we see in our simulation results: under maximum likelihood, all $p - k$ noise predictors are selected in every replicate of every scenario, with the false positive frequency matching the number of true zeros. These failures motivate penalized estimation. The penalized framework optimizes the function $\ell(\beta, \nu)$ subject to constraints,

$$Q(\beta, \nu; \tau) = \ell(\beta, \nu) - \tau \sum_{j=1}^{p} P_j(\beta_j), \tag{11}$$

where $P_j(\cdot)$ is a penalty on the jth coefficient and $\tau \geq 0$ is a regularization strength parameter. Depending on the choice of P_j, the estimator can either shrink without selecting (ridge, Liu) or shrink and select simultaneously (LASSO, Adaptive LASSO), which is the difference from all previously proposed COMP regularization methods.

3. Proposed Penalized COMP Estimators

3.1. Penalized Likelihood Framework

The overall framework of the proposed methodology is summarized in Figure 1. The penalized COM-Poisson problem, introduced in Section 2.4, maximizes $Q(\beta, \nu; \tau)$ over β for a given penalty function $P_j(\cdot)$ and regularization level $\tau \geq 0$. Two specific choices of P_j define the estimators proposed in this paper, developed in Sections 3.2 and 3.3 respectively. In both cases, following the convention of Tibshirani [38], β_0 is not penalized, which does not affect the fit of the model but ensures that predictions are centered on the intercept scale. The dispersion parameter ν is not penalized, and it is estimated jointly at each outer iteration using profile likelihood, as outlined in Section 3.4.

3.2. COMP LASSO

By letting $P_j(\beta_j) = |\beta_j|$ we obtain the L_1-penalized COMP objective,

$$\hat{\beta}_{\text{COMP-LASSO}} = \arg\max_{\beta} \left\{ \sum_{i=1}^{n} \Big[y_i \, x_i^\top \beta \; - \; \nu \log(y_i!) \; - \; \log Z\big(e^{x_i^\top \beta}, \, \nu\big) \Big] \right. \\ \left. - \; \tau \sum_{j=1}^{p} \big|\beta_j\big| \right\}. \tag{12}$$

The L_1 constraint region is a cross-polytope with corners on the coordinate axes. This is the space where likelihood contours often touch at vertices where some coordinates are exactly zero, and this is why the COMP-LASSO is a selector rather than a shrinker alone, a property not shared by ridge-type shrinkage estimators. This geometry yields sparse solutions, which become sparser as τ gets larger, and which remove less influential predictors first, as demonstrated by Tibshirani [38] for linear models and Park and Hastie [33] for generalized linear models.

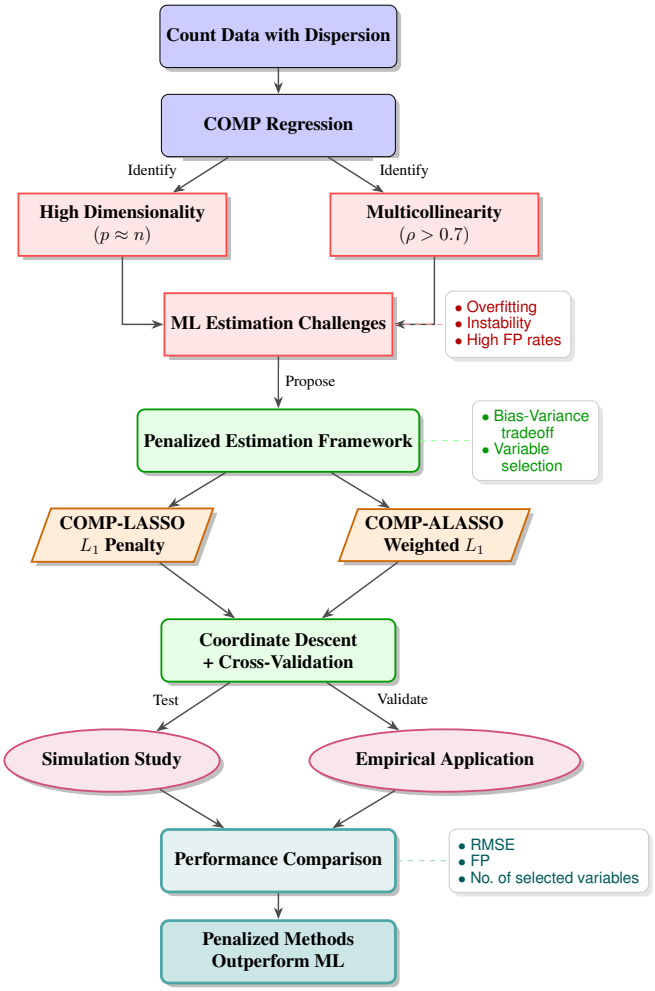

Figure 1. Research Framework.

3.3. COMP Adaptive LASSO

The standard COMP-LASSO is unable to handle multicollinearity, as it uniformly penalizes all the coefficients: if the noise predictors are correlated, they are selected in addition to the true effects, since the cost of the penalty is the same for both the noise and true effects. Zou [43] solved this by using coefficient-specific weights inversely proportional to a preliminary estimator $\tilde{\beta}$,

$$w_j = \frac{1}{|\tilde{\beta}_j|^\gamma + \varepsilon}, \qquad j = 1, \ldots, p, \tag{13}$$

where $\gamma > 0$ controls how aggressively small preliminary estimates are up-weighted relative to large ones and ε is a small stabilization constant. Large weights are assigned to coefficients with small preliminary estimates (inducing heavy shrinkage), while small weights are assigned to coefficients with large preliminary estimates (allowing them

to be retained). The COMP-ALASSO estimator has the following form:

$$\hat{\beta}_{\text{COMP-ALASSO}} = \arg\max_{\beta} \left\{ \sum_{i=1}^{n} \left[y_i \, x_i^{\top}\beta \; - \; \nu \log(y_i!) \; - \; \log Z\big(e^{x_i^{\top}\beta}, \, \nu\big) \right] \right. \\ \left. - \; \tau \sum_{j=1}^{p} w_j \, |\beta_j| \right\}. \tag{14}$$

The preliminary estimate $\tilde{\beta}$ is obtained from Poisson ridge regression with penalty parameter selected by 5-fold cross-validation. Using a Poisson ridge here rather than a COMP ridge is computationally expedient, but the two models are not on the same scale and the difference deserves comment. The Poisson working model posits $\log \mu = x^{\top}\beta$, whereas under the COM-Poisson model the mean satisfies $\log \mu \approx x^{\top}\beta/\nu$ by (2), so the preliminary estimator targets β_0/ν_0 rather than β_0. The distortion is a single positive multiplicative factor common to all coordinates. Since w_j is a decreasing function of $|\tilde{\beta}_j|$ and the weights are subsequently normalized by their median, a common rescaling of $\tilde{\beta}$ leaves the ordering of the weights unchanged and shifts their overall level only in a way the normalization absorbs; which coefficients are penalized most heavily relative to the others is therefore unaffected. Invariance of the ranking is nevertheless weaker than the $\sqrt{n}$-consistency for β_0 that the formal oracle argument of Zou [43] and Zhang and Lu [42] requires of an initial estimator. That requirement, together with the joint estimation of ν and the regime $p \gtrsim n$ in which restricted eigenvalue conditions [9, 10] would replace the usual fixed-p design condition, means that no formal oracle property is claimed for COMP-ALASSO here; the simulation and application results of Sections 4 and 5 are offered as finite-sample evidence rather than as verification of an asymptotic theory, and a proof under primitive conditions is left to separate work. The exponent γ is fixed at 1, following Zou [43]. The stabilization constant is set to $\varepsilon_w = \sqrt{\log(p+1)/n}$, which adapts to the dimensionality of the problem, with weights subsequently normalized by their median and clipped to the interval $[0.02, 50]$ to prevent extreme down- or up-weighting of individual coefficients.

3.4. Optimization: Iteratively Reweighted Coordinate Descent

Neither objective admits a closed-form solution. Both are non-differentiable at zero because of the L_1 term, and the COMP log-likelihood adds a non-linear normalizing constant at each evaluation. We address both difficulties using a hybrid algorithm combining the iteratively reweighted least squares approach of McCullagh and Nelder [31] and the coordinate descent approach of Friedman, Hastie, and Tibshirani [19].

At the current iterate $\beta^{(t)}$, compute for each observation the COMP conditional mean μ_i and variance σ_i^2 by direct summation when $\lambda_i^{1/\nu} \leq 5$ and otherwise by the asymptotic approximations (3) of Sellers and Shmueli [35], as detailed in Section 3.6. Form the linear predictor $\eta_i = x_i^{\top}\beta^{(t)}$ and the working response

$$z_i = \eta_i + \frac{y_i - \mu_i}{\sigma_i^2}, \tag{15}$$

with working weight $v_i = \sigma_i^2$.

The form of this weight follows from a single identity. Because the link is $\eta = \log \lambda$, we have $\mathrm{d}\lambda/\mathrm{d}\eta = \lambda$, and combining this with the variance expression in (2) gives

$$\frac{\partial \mu}{\partial \eta} \;=\; \frac{\partial \mu}{\partial \lambda}\,\frac{\mathrm{d}\lambda}{\mathrm{d}\eta} \;=\; \lambda\,\frac{\partial \mu}{\partial \lambda} \;=\; \mathrm{Var}(Y). \tag{16}$$

The generic iteratively reweighted least squares weight $v = (\partial\mu/\partial\eta)^2/\,\mathrm{Var}(Y)$ therefore collapses to $v = \mathrm{Var}(Y)$, so that $v_i = \sigma_i^2$ exactly, with no approximation beyond that already used for μ_i and σ_i^2 themselves. This identity is not accidental, since writing (1) as $\exp\{\eta y - \nu \log(y!) - \log Z(\lambda, \nu)\}$ exhibits the COM-Poisson distribution as a two-parameter canonical exponential family with natural parameters $(\log \lambda, -\nu)$, sufficient statistics $(y, \log(y!))$ and cumulant function $\log Z$; for fixed ν it reduces to a one-parameter family in $\eta = \log \lambda$. Hence $\mu = \partial \log Z/\partial\eta$ and $\mathrm{Var}(Y) = \partial^2 \log Z/\partial\eta^2 = \partial\mu/\partial\eta$, so that (16) is the canonical-link simplification familiar from generalized

linear models [31]. The COM-Poisson case differs not in lacking this structure but in lacking a closed form for $\log Z$, so its cumulants must be evaluated numerically or through (2) rather than read off analytically.

This reduces the penalized COMP problem to a penalized least squares subproblem at each outer iteration, with weights that reflect the true dispersion structure of the model and not Poisson surrogates.

For the weighted subproblem, repeat the process above for $j = 1, \ldots, p$ and update each coefficient using soft-thresholding. Calculate the partial residual excluding predictor j,

$$r_{ij} = z_i - \beta_0 - \sum_{k \neq j} x_{ik}\, \beta_k^{(t)}. \tag{17}$$

The COMP-LASSO coordinate update is then

$$\beta_j^{(t+1)} = \frac{S\big(\sum_{i=1}^n \sigma_i^2\, x_{ij}\, r_{ij},\ \tau\big)}{\sum_{i=1}^n \sigma_i^2\, x_{ij}^2 + \delta}, \tag{18}$$

with a small ridge stabilization term $\delta > 0$ and the soft-thresholding operator $S(\cdot, \tau)$,

$$S(z, \tau) = \operatorname{sign}(z)\, \max\big(|z| - \tau,\ 0\big). \tag{19}$$

For COMP-ALASSO, the only difference is that τ is replaced by the coefficient-specific threshold τw_j, and the computational structure is the same. The inner loop is $O(np)$, with the residuals being updated incrementally after each coordinate step.

Profile likelihood is used to update ν after each outer iteration by maximizing $\ell(\beta^{(t+1)}, \nu)$ for a fixed value of β, which is done by a grid search over a set of candidate values. A coarse grid is used followed by a fine local search, which involves $O(G)$ evaluations of $\log Z$ per outer iteration, where G is the grid size. The outer loop is stopped as soon as the largest absolute change in (β_0, β) falls below $\epsilon_{\text{out}} = 10^{-3}$, or when $T_{\max} = 20$ outer iterations are reached. The entire process is summarized in Algorithm 1, and the numerical constants, series truncation and safeguards it relies on are set out in Section 3.6.

3.5. Tuning Parameter Selection

The penalty strength τ is chosen by 5-fold cross-validation based on held-out folds of the data, by minimizing the mean squared prediction error,

$$\hat{\tau} = \arg\min_{\tau \in \mathcal{T}}\ \frac{1}{5} \sum_{f=1}^{5} \frac{1}{n_f} \sum_{i \in C_f} \big(y_i - \hat{\mu}_i^{-f}(\tau)\big)^2, \tag{20}$$

where C_f indexes the fold f for which the prediction is made, n_f is the size of fold f, and $\hat{\mu}_i^{-f}(\tau)$ is the mean predicted for observation i by the model trained on the other four folds. The candidate grid $\mathcal{T}$ consists of 20 points spaced logarithmically from $\tau_{\max} = 1.2 \cdot \max_j |\nabla_j \ell(0)|$, a margin above the smallest penalty at which all coefficients are driven to zero, down to 0.1% of $\tau_{\max}$. This warm-starting along the path, with each solution starting from the previous τ value, significantly cuts down on total computation [18].

The cost of the procedure can be stated explicitly. One full cycle of coordinate descent over all p coordinates costs $O(np)$ operations when residuals are updated incrementally rather than recomputed, so a single fit costs $O(T_{\max} T_{\text{in}} np + T_{\max} G n M)$, the second term being the profile search over ν with G grid points and M series terms per evaluation of $\log Z$. The complete tuning procedure multiplies the first component by the number of folds and the size of $\mathcal{T}$, which is why ν is held fixed within the folds as described in Section 3.6. The cost grows roughly linearly in p at fixed n, as the $O(np)$ term dominates. For reference, the corresponding Poisson LASSO fitted by coordinate descent is faster by orders of magnitude, since it requires neither the series evaluation nor the profile search; the additional expense is the price of estimating the dispersion rather than assuming it.

Algorithm 1 COMP Regression IRLS + Coordinate Descent (COMP-LASSO / COMP-ALASSO)

Input: $(\mathbf{y}, \mathbf{X})$, penalty type, grid $\mathcal{T}$, $\epsilon_{\text{out}} = 10^{-3}$, $\epsilon_{\text{in}} = 10^{-4}$, $T_{\max} = 20$, $T_{\text{in}} = 50$, $\delta = 10^{-4}$
Output: $\hat{\boldsymbol{\beta}},\ \hat{\nu},\ \hat{S} = \{j : \hat{\beta}_j \neq 0\}$

Inner routine FIT$(\mathbf{y}, \mathbf{X}, \tau, \mathbf{w}, \nu;\ \textit{profile})$ (repeat for $t = 0, \ldots, T_{\max}$)

Step 1 — Working quantities: for each i, compute μ_i and σ_i^2 by direct summation if $\lambda_i^{1/\nu} \leq 5$, else by

$$\mu_i \approx \lambda_i^{1/\nu} - \tfrac{\nu-1}{2\nu}, \quad \sigma_i^2 \approx \tfrac{\lambda_i^{1/\nu}}{\nu}, \quad \text{then } z_i \leftarrow \eta_i + \tfrac{y_i - \mu_i}{\sigma_i^2}$$

Step 2 — Coordinate descent: cycle over $j = 1, \ldots, p$ (at most T_{in} cycles, tolerance ϵ_{in})

$$\beta_j^{(t+1)} \leftarrow \frac{S\big(\sum_i \sigma_i^2\, x_{ij}\, r_{ij},\ \tau w_j\big)}{\sum_i \sigma_i^2\, x_{ij}^2 + \delta}, \quad S(z, \lambda) = \operatorname{sign}(z)\max(|z| - \lambda,\ 0)$$

where $r_{ij} = z_i - \beta_0 - \sum_{k \neq j} x_{ik}\beta_k^{(t)}$

Step 3 — Dispersion update: if *profile* is true, $\nu^{(t+1)} \leftarrow \arg\max_\nu\ \ell\big(\boldsymbol{\beta}^{(t+1)}, \nu\big)$ via profile likelihood grid search; otherwise ν is held at its input value

Step 4 — Convergence: stop if $\max\big\{\|\boldsymbol{\beta}^{(t+1)} - \boldsymbol{\beta}^{(t)}\|_\infty, |\beta_0^{(t+1)} - \beta_0^{(t)}|\big\} < \epsilon_{\text{out}}$

return $(\boldsymbol{\beta}, \nu)$

Stage 1 — Initialization

Warm-start $\boldsymbol{\beta}^{(0)}$ from Poisson GLM
if COMP-ALASSO **then**
    $w_j \leftarrow \big(|\tilde{\beta}_j|^\gamma + \varepsilon_w\big)^{-1}$ using ridge estimate $\tilde{\boldsymbol{\beta}}$
else $w_j \leftarrow 1$
end if
Profile ν once from $\boldsymbol{\beta}^{(0)}$ to obtain $\hat{\nu}^{(0)}$

Stage 2 — Tuning (five-fold cross-validation, ν held fixed at $\hat{\nu}^{(0)}$)

for each $\tau \in \mathcal{T}$, in decreasing order with warm starts **do**
    for $f = 1, \ldots, 5$ **do**
        $(\boldsymbol{\beta}^{-f}, \cdot) \leftarrow$ FIT$(\mathbf{y}_{-f}, \mathbf{X}_{-f}, \tau, \mathbf{w}, \hat{\nu}^{(0)};\ \textit{profile} = \text{false})$; predict on fold f
    end for
end for
$\hat{\tau} \leftarrow \arg\min_{\tau \in \mathcal{T}}$ cross-validated MSE, as in (20)

Stage 3 — Final fit (complete dataset, ν re-profiled)

$(\hat{\boldsymbol{\beta}}, \hat{\nu}) \leftarrow$ FIT$(\mathbf{y}, \mathbf{X}, \hat{\tau}, \mathbf{w}, \hat{\nu}^{(0)};\ \textit{profile} = \text{true})$
if $|\hat{\nu} - \hat{\nu}^{(0)}| > 0.1$ **then**
    repeat Stage 3 with $\hat{\nu}^{(0)} \leftarrow \hat{\nu}$
end if

return $\hat{\boldsymbol{\beta}},\ \hat{\nu},\ \hat{S} = \{j : \hat{\beta}_j \neq 0\}$

3.6. Numerical Implementation

Since $\log Z(\lambda, \nu)$ admits no closed form, the estimator definitions above leave several numerical choices unspecified, and these affect reproducibility. The series $Z(\lambda, \nu)$ defined beneath (1) is evaluated by fixed truncation at $M = 150$ terms,

$$\log Z(\lambda, \nu)\ \approx\ \log Z_M(\lambda, \nu)\ =\ \log \sum_{j=0}^{M} \exp\big\{j \log \lambda - \nu \log \Gamma(j+1)\big\}, \tag{21}$$

computed through the log-sum-exp identity so that no individual term overflows. The value $M = 150$ was chosen because the linear predictor is confined to $[-5, 5]$ during estimation, bounding $\lambda \leq e^5 \approx 148$, and over the range of (λ, ν) encountered the discarded tail is negligible relative to the retained mass. A fixed-M rule with an empirically adequate margin does not, however, certify a bound on the remainder $R_M = \sum_{j>M} \lambda^j/(j!)^\nu$ uniformly over the parameter space, and the margin narrows when ν approaches the lower end of the search range with λ near its bound; should (21) return a non-finite value, the routine reverts to the Poisson value $\log Z = \lambda$. Re-evaluating the profile likelihood for ν in the application of Section 5.4 with M raised successively to 300, 600, 1200 and 3000 left the maximizing value unchanged, indicating that the truncation error at $M = 150$ is already negligible relative to the resolution of the search grid. The conditional moments are obtained by direct summation when $\lambda^{1/\nu} \leq 5$ and by the approximations (3) otherwise, confining the latter to the regime in which Sellers and Shmueli [35] establish their accuracy.

Three clipping operations are applied during estimation, distinct from the $[-3, 3]$ clipping used in data generation in Section 4.1. The linear predictor is confined to $[-5, 5]$ whenever the likelihood or the working quantities are evaluated, which bounds λ and prevents overflow in (21); the intercept and coefficients are confined to $[-5, 5]$ after each coordinate descent cycle, which prevents a diverging iterate from propagating; and at prediction the linear predictor is confined to $[-4, 4]$ with the fitted mean bounded above by $\max\{100, 3\max_i y_i\}$ over the training response. The preliminary ridge estimate entering (13) is confined to $[-3, 3]$ before the adaptive weights are formed. These are numerical safeguards rather than statistical constraints and bind rarely for the penalized estimators, but iterates obtained without them will differ. The ridge stabilization constant in the coordinate descent denominator is $\delta = 10^{-4}$ throughout for both penalized estimators, the inner coordinate descent uses tolerance $\epsilon_{\text{in}} = 10^{-4}$ and at most $T_{\text{in}} = 50$ cycles, and the dispersion parameter is updated by a two-stage grid search, a coarse grid $\{0.3, 0.5, \ldots, 2.5\}$ followed by a local refinement of width ± 0.25 in steps of 0.05 truncated to $[0.2, 4.0]$, giving $G \approx 23$ evaluations of the profile likelihood per outer iteration.

Re-profiling ν inside every cross-validation fold would be prohibitively expensive, so the procedure is executed in three stages. Stage 1 obtains a warm-start coefficient vector from a Poisson LASSO fit and profiles ν once from it, giving $\hat{\nu}^{(0)}$. Stage 2 carries out the five-fold cross-validation over the grid $\mathcal{T}$ with ν held fixed at $\hat{\nu}^{(0)}$, so that each fold runs only the iteratively reweighted loop and the coordinate descent, and selects $\hat{\tau}$ as the minimizer of (20). Stage 3 refits the model on the full dataset at $\hat{\tau}$ with ν re-profiled from the resulting coefficients, repeating the fit at the updated value if the re-profiled ν differs from $\hat{\nu}^{(0)}$ by more than 0.1. Holding ν fixed within the folds is an approximation adopted for tractability, and Algorithm 1 sets out the three stages explicitly, with the estimation loop written as an inner routine that Stage 2 calls once per fold per candidate τ and Stage 3 calls once on the complete dataset.

All computations were carried out in R version 4.2.2. The Poisson warm start and the preliminary ridge estimate are obtained with `glmnet`, while the truncated evaluation of $\log Z$, the iteratively reweighted loop, the coordinate descent and the profile search over ν are implemented directly. The CrohnD data are distributed with the `robustbase` package [27] and the CreditCard data with the `AER` package [20], so both applications can be reproduced from publicly available sources without additional data access.

4. Simulation Study

4.1. Experimental Design

The simulation evaluates the three estimators COMP-ML, COMP-LASSO, and COMP-ALASSO across 81 scenarios constructed by fully crossing sample size $n \in \{70, 100, 200\}$, predictor dimension $p \in \{30, 50, 100\}$, correlation $\rho \in \{0.7, 0.9, 0.95\}$, and dispersion $\nu \in \{0.7, 1.0, 1.5\}$. The sparsity is set to 20%, meaning that $k = \lfloor 0.2p \rfloor$ coefficients of β are non-zero, randomly selected from the interval $U[0.5, 1.5]$. The intercept is $\beta_0 = 0.5$. To avoid numerically pathological counts, linear predictors are first clipped to the interval $[-3, 3]$ and then exponentiated, similar to the regularization strategy proposed by Bühlmann and Van De Geer [10] for high-dimensional GLMs. The number of Monte Carlo replications is 50 for each scenario, a figure dictated by computational cost, since each penalized fit nests a coordinate descent within an iteratively reweighted loop within

a profile search over ν, every likelihood evaluation of which requires a truncated series, and the whole is repeated across a five-fold cross-validation over the τ grid for each of the 81 scenarios. A replication count of this order is common practice in simulation studies of penalized estimation where each fit is itself expensive: Algamal and Lee [6] use 50 replications in their study of the adaptive modified elastic net for penalized Poisson regression, Lu, Zhu and Lian [29] likewise use 50 in their simulations for high-dimensional quantile tensor regression, and the same count is adopted in the comparative study of generalized LASSO regularization for regression models by Flexeder [17]. Every entry in Tables 1–6 carries its Monte Carlo standard deviation, so that the precision of each comparison can be assessed directly; Section 4.4 examines these. The tuning parameters of COMP-LASSO and COMP-ALASSO are chosen by 5-fold cross-validation through minimization of the mean squared prediction error (MSPE), similar to the tuning procedure proposed by Friedman, Hastie and Tibshirani [19]. Performance is assessed on a fixed external test set of $n_{\text{test}} = 200$ observations from the same data-generating process, completely separate from the training data, so that the test metrics are not contaminated by in-sample fit. The ratios of sample size to predictor dimension thus vary widely across the design.

4.2. The COMP-ML Benchmark

Since the exact maximum likelihood estimator does not exist in the cells with $p \geq n$, the benchmark against which the penalized estimators are compared requires a description as explicit as Algorithm 1.

COMP-ML is computed by the same iteratively reweighted scheme as the penalized estimators with the penalty set to zero. A general-purpose numerical optimizer is not used, nor is a Moore–Penrose generalized inverse, nor is any column of X removed automatically. Instead the coefficient update solves the weighted least squares subproblem coordinatewise, with the same ridge term δ_{ML} in the denominator that appears in (18), so that the working cross-product remains invertible when $X^\top W X$ is singular. The iterate is initialized from a Poisson ridge fit, ν is profiled on the coarse grid $\{0.4, 0.8, \ldots, 2.4\}$ with a local refinement of width ± 0.3 in steps of 0.1, the linear predictor and coefficients are subject to the same $[-5, 5]$ safeguards described in Section 3.6, and the outer loop runs to the same $T_{\max} = 20$ with tolerance $\epsilon_{\text{out}} = 10^{-3}$. All three estimators therefore see identical data, identical safeguards and identical stopping rules within a scenario; what differs is only the penalty.

The stabilization constant is not held fixed across the design. It is set to

$$\delta_{\text{ML}} = \begin{cases} 10^{-2}, & p \geq n, \\ 10^{-3}, & n/2 \leq p < n, \\ 10^{-4}, & p < n/2, \end{cases} \tag{22}$$

the more severely rank-deficient designs requiring more stabilization to return a finite iterate. This has a consequence that bears on how the results below may be read. Within a scenario the comparison is unaffected: δ_{ML} is a constant of that cell, all three estimators are fitted to the same replicates, and the differences in Tables 1–6 are differences between estimators under identical conditions. Across scenarios, however, δ_{ML} changes discontinuously with p/n, so comparing COMP-ML at one (n, p) with COMP-ML at another confounds the behaviour of the estimator with the stabilization applied to it. Trends in the COMP-ML column as a function of n at fixed p therefore carry no interpretation as evidence about the response of maximum likelihood to sample size, and none is placed on them in Section 4.4. A design holding δ_{ML} constant throughout would admit that reading and is the appropriate form for future work.

Convergence was monitored across all 81 scenarios and recorded at 100% for every estimator in every cell. The figure requires qualification: a fit is recorded as converged when it completes $T_{\max}$ outer iterations with a finite log-likelihood, so the statistic certifies non-divergence rather than attainment of ϵ_{out}. No replicate was discarded, and every tabulated cell rests on the full 50 replications.

4.3. Performance Metrics

The evaluation is based on three primary metrics, supplemented by the selection measures reported in Section 4.5. Prediction accuracy is measured on the test set through the root mean squared error,

$$\mathrm{RMSE} = \sqrt{\frac{1}{n_{\mathrm{test}}} \sum_{i=1}^{n_{\mathrm{test}}} \left(y_i - \hat{\mu}_i\right)^2}, \tag{23}$$

where $\hat{\mu}_i$ is the fitted COMP conditional mean. RMSE is the main accuracy criterion, since Cameron and Trivedi [12] argue that squared-error loss is well suited to count-regression comparisons because it gives greater weight to large prediction errors, which arise here when COMP-ML makes extreme predictions under almost singular designs. Variable selection is evaluated through the number of false positives,

$$\mathrm{FP} = \#\{\, j : \hat{\beta}_j \neq 0,\ \beta_j = 0 \,\}, \tag{24}$$

where $\#\{\cdot\}$ denotes the cardinality of a set, that is, the number of elements it contains. FP is therefore the number of noise predictors that are falsely retained. This directly measures the Type I selection error, as in the original LASSO framework of Tibshirani [38] and the adaptive LASSO framework of Zou [43]. Lower is better: the theoretical minimum of COMP-ML is zero, but its practical value is always $p - k$, as it never zeroes a coefficient. The third metric, the number of selected variables,

$$n_{\mathrm{selected}} = \#\{\, j : \hat{\beta}_j \neq 0 \,\}, \tag{25}$$

complements FP by giving an overall picture of model complexity. A method could attain a low FP while rejecting many true signals, appearing favourable on FP alone while yielding $n_{\mathrm{selected}} \ll k$. Reporting both together, along with the true nonzero count k, makes the precision–recall trade-off transparent, as suggested by Bickel, Ritov and Tsybakov [9] for the honest evaluation of high-dimensional selectors. Each mean is reported with its Monte Carlo standard deviation over the 50 replicates, indicating the variability of the estimate and not just its central tendency.

4.4. RMSE Results

The consistent result across Tables 1–3 and in Figure 2 is that penalized estimation outperforms ML in 78 of the 81 scenarios, and the size of the difference is directly tied to the severity of the conditioning of the design. In the overdispersed case ($\nu = 0.7$; Table 1), ML estimation shows substantial instability. At $n = 200, p = 100, \rho = 0.7$ it reaches 100.677 (worse than an intercept-only model), while COMP-LASSO and COMP-ALASSO remain comparatively stable across scenarios, between 17 and 27. The best ML value in Table 1 is 27.158; COMP-ALASSO in that same cell is 17.024, 37% lower. At $\rho = 0.95$ in that cell the reduction is 37%. Penalized estimators achieve lower RMSE in all 27 scenarios, and COMP-ALASSO beats COMP-LASSO in 17 of them, especially where the dimensionality is not too high, indicating that the ridge-based preliminary weights separate signal from noise effectively.

For the equidispersed case ($\nu = 1$, Table 2), ML stabilizes; the Poisson warm start here is essentially exact, yet penalized estimators still achieve lower RMSE in all scenarios. The largest gap is at $n = 200, p = 100, \rho = 0.7$: ML $= 28.197$ versus COMP-LASSO $= 7.391$, a 74% reduction. Even under the mildest conditions in this design, $n = 70, p = 30, \rho = 0.7$, COMP-LASSO reduces ML's error by about 50%. COMP-LASSO achieves the lowest RMSE in 26 of the 27 scenarios, and COMP-ALASSO dominates in the last cell at the least-determined corner, where $p/n \approx 100/70$.

The picture changes in the underdispersed case ($\nu = 1.5$; Table 3). Under underdispersion, all three methods converge towards one another, and ML attains the lower mean RMSE in three scenarios, all at $n = 70$, where strong penalization may occasionally introduce additional shrinkage bias. These three cells do not constitute evidence in favour of ML. The largest margin is 0.284 RMSE points, and relative to Monte Carlo error the differences are slight: measured against the better of the two penalized estimators, and expressed in units of the Monte Carlo standard error of the difference, the margins are 0.29 at $(p, \rho) = (50, 0.9)$, 1.62 at $(100, 0.7)$ and 2.18 at $(100, 0.9)$. Because

Table 1. RMSE across scenarios ($\nu = 0.7$).

n	p	ρ	True nonzero	COMP-ML	COMP-LASSO	COMP-ALASSO
70	30	0.7	6	57.751 (17.487)	18.700 (2.480)	**18.395 (2.560)**
100	30	0.7	6	49.331 (12.102)	18.639 (2.250)	**18.130 (2.106)**
200	30	0.7	6	27.158 (6.311)	17.443 (2.351)	**17.024 (2.394)**
70	50	0.7	10	34.083 (16.722)	22.504 (2.248)	**22.407 (2.331)**
100	50	0.7	10	78.645 (22.698)	21.417 (2.068)	**21.268 (2.041)**
200	50	0.7	10	45.846 (10.022)	20.303 (1.458)	**20.088 (1.563)**
70	100	0.7	20	33.491 (2.550)	27.269 (2.267)	**27.192 (2.200)**
100	100	0.7	20	33.295 (1.799)	**25.626 (1.842)**	25.684 (1.970)
200	100	0.7	20	100.677 (17.162)	23.915 (1.699)	**23.692 (1.803)**
70	30	0.9	6	65.250 (17.653)	**20.529 (2.627)**	20.565 (2.650)
100	30	0.9	6	49.444 (13.483)	20.181 (2.371)	**20.043 (2.359)**
200	30	0.9	6	31.684 (8.072)	19.700 (1.863)	**19.672 (1.940)**
70	50	0.9	10	43.405 (28.158)	22.657 (1.998)	**22.556 (1.929)**
100	50	0.9	10	86.871 (14.497)	22.377 (1.813)	**22.340 (1.831)**
200	50	0.9	10	45.286 (12.938)	21.941 (1.312)	**21.833 (1.420)**
70	100	0.9	20	34.670 (2.453)	**25.537 (1.807)**	25.541 (1.932)
100	100	0.9	20	34.952 (2.386)	**24.772 (1.708)**	24.875 (1.820)
200	100	0.9	20	95.223 (13.399)	**23.583 (1.279)**	23.631 (1.355)
70	30	0.95	6	68.730 (18.353)	20.560 (2.017)	**20.534 (2.061)**
100	30	0.95	6	51.909 (19.497)	20.578 (2.447)	**20.543 (2.331)**
200	30	0.95	6	32.201 (8.516)	20.192 (1.559)	**20.186 (1.585)**
70	50	0.95	10	52.635 (31.444)	**22.489 (2.062)**	22.602 (1.884)
100	50	0.95	10	87.435 (13.945)	**22.470 (1.634)**	22.535 (1.842)
200	50	0.95	10	37.086 (12.382)	22.390 (1.157)	**22.271 (1.178)**
70	100	0.95	20	35.246 (2.173)	**24.652 (1.974)**	24.694 (1.911)
100	100	0.95	20	35.311 (2.174)	**24.314 (1.509)**	24.427 (1.524)
200	100	0.95	20	97.646 (15.270)	**23.984 (1.389)**	23.994 (1.384)

Note. Mean test-set RMSE over 50 replications, Monte Carlo standard deviation in parentheses; the Monte Carlo standard error is that value divided by $\sqrt{50}$. Bold: lowest in row.

the estimators are fitted to common replicates, these figures treat as independent quantities that are positively correlated, and are therefore conservative. The three estimators are best described as indistinguishable in these cells. No exceptions occur from $n = 100$ onwards, where COMP-LASSO prevails.

The reliability of these comparisons can be assessed directly from the dispersion figures already reported in the tables, since the Monte Carlo standard error of each entry is its standard deviation divided by $\sqrt{50}$. Across all three RMSE tables the Monte Carlo standard error never exceeds 0.38 RMSE points for either penalized estimator and 4.45 for COMP-ML, against differences between the methods that routinely run to tens of RMSE points. The penalized advantage is 15.6 standard errors wide at $(\nu, n, p, \rho) = (0.7, 70, 30, 0.7)$, 31.5 at $(0.7, 200, 100, 0.7)$ and 23.3 at $(1, 200, 100, 0.7)$. Doubling the replication count would reduce these standard errors by a factor of $\sqrt{2}$ and leave the ordering of the estimators unchanged.

Figure 2 reveals two features that the tables cannot. The COMP-ML curve is non-monotone in n in the $p = 100$ columns for two values of ν, rising sharply between $n = 100$ and $n = 200$. We draw no statistical conclusion from this. As set out in Section 4.2, the stabilization constant δ_{ML} in (22) falls from 10^{-2} to 10^{-3} at exactly that transition, since $p = 100$ crosses from $p \geq n$ into $n/2 \leq p < n$. The estimated dispersion moves in step with it: for $\nu = 0.7$ it stands at $\hat{\nu} \approx 1.10$ for $n \leq 100$, almost without variation across ρ, and falls to $\hat{\nu} \approx 0.51$ at $n = 200$. The pattern is therefore attributable to the stabilization schedule rather than to the response of maximum likelihood to sample size, and it is reported here for transparency rather than as a finding. The COMP-LASSO and COMP-ALASSO curves decrease gradually throughout. The correlation effect is equally clear, since the three line styles for the penalized methods are nearly indistinguishable across ρ, while the spread of COMP-ML widens with p, indicating that L_1 penalization is substantially less sensitive to strong multicollinearity than the unpenalized fit.

Table 2. RMSE across scenarios ($\nu = 1$).

n	p	ρ	True nonzero	COMP-ML	COMP-LASSO	COMP-ALASSO
70	30	0.7	6	13.574 (7.435)	**6.981 (1.520)**	7.100 (1.453)
100	30	0.7	6	13.911 (4.997)	**6.478 (1.101)**	6.873 (1.268)
200	30	0.7	6	12.678 (2.876)	**6.019 (1.019)**	6.475 (1.199)
70	50	0.7	10	8.528 (0.831)	**7.590 (1.306)**	7.808 (1.396)
100	50	0.7	10	17.721 (8.518)	**7.214 (1.135)**	7.523 (1.305)
200	50	0.7	10	16.160 (3.278)	**6.719 (0.766)**	7.119 (1.034)
70	100	0.7	20	9.230 (0.711)	8.901 (1.520)	**8.508 (0.994)**
100	100	0.7	20	9.213 (0.651)	**7.888 (0.789)**	7.983 (0.870)
200	100	0.7	20	28.197 (6.271)	**7.391 (0.683)**	7.649 (0.722)
70	30	0.9	6	18.477 (7.980)	**7.740 (1.317)**	7.928 (1.380)
100	30	0.9	6	18.049 (3.526)	**7.522 (1.299)**	7.815 (1.428)
200	30	0.9	6	13.412 (3.169)	**6.991 (0.915)**	7.206 (0.963)
70	50	0.9	10	9.391 (3.054)	**7.973 (1.063)**	8.253 (1.198)
100	50	0.9	10	23.264 (7.425)	**7.800 (1.019)**	7.965 (0.925)
200	50	0.9	10	18.598 (3.876)	**7.663 (0.923)**	7.940 (1.084)
70	100	0.9	20	9.308 (0.874)	**8.696 (1.139)**	8.751 (1.205)
100	100	0.9	20	9.554 (0.750)	**8.203 (0.800)**	8.366 (1.029)
200	100	0.9	20	29.389 (3.694)	**7.778 (0.815)**	7.891 (0.835)
70	30	0.95	6	20.196 (6.681)	**7.936 (1.203)**	8.091 (1.228)
100	30	0.95	6	19.107 (3.476)	**7.422 (1.004)**	7.503 (1.063)
200	30	0.95	6	14.287 (3.077)	**7.429 (1.054)**	7.504 (1.167)
70	50	0.95	10	11.187 (5.298)	**8.671 (2.051)**	8.689 (1.905)
100	50	0.95	10	24.584 (7.484)	**8.364 (1.377)**	8.527 (1.456)
200	50	0.95	10	18.396 (3.096)	**7.727 (0.994)**	7.817 (1.002)
70	100	0.95	20	9.384 (0.903)	**8.648 (1.038)**	8.799 (1.225)
100	100	0.95	20	9.671 (0.857)	**8.333 (0.951)**	8.423 (1.054)
200	100	0.95	20	29.720 (4.745)	**8.094 (0.898)**	8.186 (0.960)

Note. Mean test-set RMSE over 50 replications, Monte Carlo standard deviation in parentheses; the Monte Carlo standard error is that value divided by $\sqrt{50}$. Bold: lowest in row.

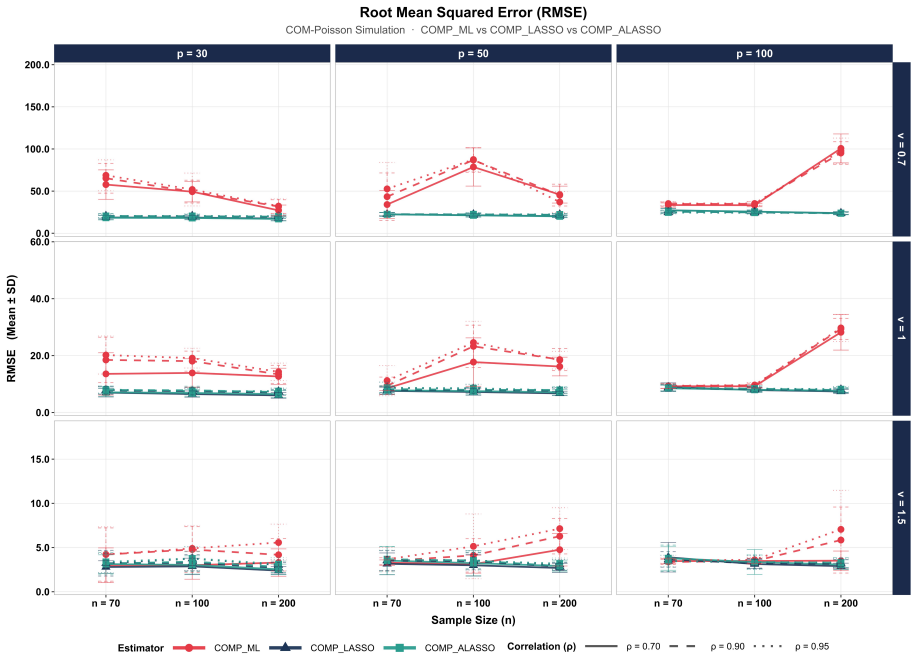

Figure 2. RMSE across all scenarios.

Table 3. RMSE across scenarios ($\nu = 1.5$).

n	p	ρ	True nonzero	COMP-ML	COMP-LASSO	COMP-ALASSO
70	30	0.7	6	3.002 (1.957)	**2.798 (0.711)**	3.133 (1.051)
100	30	0.7	6	3.000 (1.587)	**2.868 (0.927)**	3.031 (1.036)
200	30	0.7	6	3.300 (1.551)	**2.389 (0.456)**	2.537 (0.585)
70	50	0.7	10	3.286 (0.317)	**3.157 (1.231)**	3.524 (1.592)
100	50	0.7	10	3.115 (1.014)	**2.990 (1.215)**	3.233 (1.431)
200	50	0.7	10	4.755 (1.840)	**2.687 (0.507)**	2.916 (0.659)
70	100	0.7	20	**3.467 (0.291)**	3.898 (1.667)	3.786 (1.358)
100	100	0.7	20	3.478 (0.299)	**3.133 (0.514)**	3.369 (1.411)
200	100	0.7	20	3.515 (1.088)	**2.898 (0.331)**	3.067 (0.550)
70	30	0.9	6	4.210 (3.011)	3.230 (1.455)	**3.182 (1.265)**
100	30	0.9	6	4.776 (2.590)	**3.270 (0.924)**	3.412 (1.118)
200	30	0.9	6	4.179 (1.836)	**2.711 (0.593)**	2.817 (0.647)
70	50	0.9	10	**3.471 (0.378)**	3.520 (1.124)	3.681 (1.422)
100	50	0.9	10	4.114 (1.888)	**3.447 (0.960)**	3.541 (0.930)
200	50	0.9	10	6.281 (2.003)	**2.852 (0.456)**	2.903 (0.464)
70	100	0.9	20	**3.372 (0.287)**	3.656 (0.876)	3.791 (1.489)
100	100	0.9	20	3.470 (0.322)	**3.200 (0.528)**	3.355 (0.799)
200	100	0.9	20	5.848 (3.734)	**3.159 (0.467)**	3.297 (0.555)
70	30	0.95	6	4.218 (3.121)	**3.234 (1.081)**	3.376 (1.240)
100	30	0.95	6	4.905 (2.557)	**3.746 (1.348)**	3.770 (1.349)
200	30	0.95	6	5.569 (2.080)	**3.019 (0.651)**	3.129 (0.749)
70	50	0.95	10	3.665 (1.348)	3.559 (1.142)	**3.525 (1.136)**
100	50	0.95	10	5.139 (3.663)	3.557 (1.104)	**3.498 (1.019)**
200	50	0.95	10	7.139 (2.365)	**3.078 (0.582)**	3.182 (0.633)
70	100	0.95	20	3.536 (0.302)	**3.470 (0.658)**	3.472 (0.657)
100	100	0.95	20	3.611 (0.273)	**3.312 (0.762)**	3.374 (0.806)
200	100	0.95	20	7.055 (4.417)	**3.263 (0.504)**	3.309 (0.541)

Note. Mean test-set RMSE over 50 replications, Monte Carlo standard deviation in parentheses; the Monte Carlo standard error is that value divided by $\sqrt{50}$. Bold: lowest in row.

4.5. False Positive Results

Variable selection is addressed directly in Tables 4–6, which report both the false-positive counts and the number of selected variables for every scenario, and in Figures 3–4 (the false-positive blocks are discussed here, the selection blocks in Section 4.6). Throughout Tables 4–6, COMP-ML, COMP-LASSO and COMP-ALASSO are abbreviated ML, LASSO and ALASSO, k is the number of true nonzero coefficients, and the lowest (best) value in each metric block is set in bold. The first point to note is that the ML standard deviation is 0.00 in every row of all three tables: it always selects exactly $p - k$ noise predictors, namely 24 for $p = 30$, 40 for $p = 50$, and 80 for $p = 100$, with no variation and no exceptions. It is not that ML is a poor selector; it is that it cannot select at all, and the zero standard deviation makes that clear.

The zero standard deviation reported for COMP-ML throughout these tables has a structural rather than a numerical origin. COMP-ML maximizes a smooth objective with no non-differentiable term at the origin, so no mechanism sets a coefficient exactly to zero and the event $\hat{\beta}_j = 0$ occurs with probability zero. Consequently $\mathbf{1}\{\hat{\beta}_j \neq 0\} = 1$ for every j in every replicate, and both FP and n_{selected} are degenerate random variables taking the constant values $p - k$ and p by construction. The zero standard deviation is therefore a property of the estimator's definition rather than evidence about the numerical behaviour of the fit, and the same pattern would be observed for any non-thresholding estimator, including the ridge and Liu estimators proposed for this model [1, 5]. The underlying fits do vary across replicates, as the COMP-ML standard deviations in Tables 1–3 show, ranging from roughly 0.3 to 31.4; the selection metrics cannot register that variation, since they depend not on the magnitudes of the estimates but only on whether those magnitudes are exactly zero. The penalized methods cut the false-positive count to between 3 and 13 depending on the scenario, an average reduction of 85% across scenarios.

In the overdispersed case ($\nu = 0.7$, Table 4) ALASSO prevails in most cells. It achieves 3.02 false positives at $n = 200, p = 30, \rho = 0.7$, small relative to the maximum of 24. The ALASSO advantage is clearest at low p and

Table 4. False positives (FP) and number of selected variables ($\nu = 0.7$).

				False positives (FP)			No. selected variables		
n	p	ρ	k	ML	LASSO	ALASSO	ML	LASSO	ALASSO
70	30	0.7	6	24.00 (0.00)	**5.32 (3.09)**	5.32 (3.16)	30.00 (0.00)	10.48 (3.60)	**10.48 (3.52)**
100	30	0.7	6	24.00 (0.00)	4.72 (3.96)	**4.14 (2.76)**	30.00 (0.00)	10.24 (4.10)	**9.62 (2.85)**
200	30	0.7	6	24.00 (0.00)	3.46 (2.70)	**3.02 (3.32)**	30.00 (0.00)	9.34 (2.68)	**8.86 (3.34)**
70	50	0.7	10	40.00 (0.00)	6.84 (3.33)	**6.32 (3.33)**	50.00 (0.00)	13.80 (3.86)	**13.30 (3.92)**
100	50	0.7	10	40.00 (0.00)	6.66 (3.52)	**5.98 (3.20)**	50.00 (0.00)	14.54 (3.89)	**13.70 (3.69)**
200	50	0.7	10	40.00 (0.00)	6.52 (3.84)	**5.22 (3.06)**	50.00 (0.00)	15.68 (4.09)	**14.18 (3.24)**
70	100	0.7	20	80.00 (0.00)	11.42 (5.17)	**10.96 (5.25)**	100.00 (0.00)	21.42 (5.96)	**20.90 (5.93)**
100	100	0.7	20	80.00 (0.00)	11.60 (4.81)	**10.48 (4.99)**	100.00 (0.00)	23.42 (6.23)	**21.92 (6.56)**
200	100	0.7	20	80.00 (0.00)	12.26 (5.33)	**11.64 (5.13)**	100.00 (0.00)	27.32 (6.12)	**26.60 (5.59)**
70	30	0.9	6	24.00 (0.00)	4.60 (2.16)	**4.02 (2.02)**	30.00 (0.00)	8.06 (2.44)	**7.40 (2.18)**
100	30	0.9	6	24.00 (0.00)	4.82 (2.15)	**4.54 (2.55)**	30.00 (0.00)	8.70 (2.33)	**8.22 (2.77)**
200	30	0.9	6	24.00 (0.00)	4.66 (1.69)	**3.86 (1.59)**	30.00 (0.00)	9.30 (1.83)	**8.14 (1.81)**
70	50	0.9	10	40.00 (0.00)	6.38 (2.21)	**5.80 (2.01)**	50.00 (0.00)	10.88 (2.54)	**10.14 (2.25)**
100	50	0.9	10	40.00 (0.00)	7.28 (2.34)	**6.32 (2.46)**	50.00 (0.00)	12.30 (2.58)	**10.86 (2.47)**
200	50	0.9	10	40.00 (0.00)	7.94 (2.71)	**6.58 (2.70)**	50.00 (0.00)	14.22 (2.68)	**12.42 (2.58)**
70	100	0.9	20	80.00 (0.00)	10.60 (3.46)	**10.06 (3.01)**	100.00 (0.00)	16.10 (3.48)	**15.48 (3.20)**
100	100	0.9	20	80.00 (0.00)	10.82 (3.05)	**10.18 (2.57)**	100.00 (0.00)	17.48 (3.48)	**16.72 (3.18)**
200	100	0.9	20	80.00 (0.00)	12.96 (4.49)	**11.42 (3.47)**	100.00 (0.00)	21.74 (4.31)	**19.54 (3.36)**
70	30	0.95	6	24.00 (0.00)	4.84 (3.11)	**4.16 (2.57)**	30.00 (0.00)	7.42 (3.57)	**6.52 (2.77)**
100	30	0.95	6	24.00 (0.00)	4.28 (1.65)	**3.74 (1.85)**	30.00 (0.00)	7.30 (1.96)	**6.52 (2.20)**
200	30	0.95	6	24.00 (0.00)	4.10 (1.68)	**3.54 (1.89)**	30.00 (0.00)	7.72 (1.55)	**6.82 (1.96)**
70	50	0.95	10	40.00 (0.00)	6.16 (2.06)	**5.78 (2.01)**	50.00 (0.00)	8.94 (1.82)	**8.52 (1.81)**
100	50	0.95	10	40.00 (0.00)	**5.74 (1.93)**	5.90 (2.48)	50.00 (0.00)	**9.06 (1.79)**	9.10 (2.97)
200	50	0.95	10	40.00 (0.00)	7.18 (2.40)	**5.98 (2.25)**	50.00 (0.00)	11.52 (2.30)	**9.86 (2.29)**
70	100	0.95	20	80.00 (0.00)	9.28 (2.91)	**8.76 (2.98)**	100.00 (0.00)	13.06 (3.33)	**12.30 (2.88)**
100	100	0.95	20	80.00 (0.00)	9.82 (2.82)	**9.22 (2.55)**	100.00 (0.00)	14.26 (2.85)	**13.48 (2.72)**
200	100	0.95	20	80.00 (0.00)	10.64 (2.95)	**9.80 (3.29)**	100.00 (0.00)	16.48 (3.05)	**15.36 (3.32)**

Note. Mean over 50 replications, Monte Carlo standard deviation in parentheses. Bold: lowest in each block. ML, LASSO, ALASSO abbreviate COMP-ML, COMP-LASSO, COMP-ALASSO. The zero standard deviation for ML is structural, not numerical (Section 4.5).

moderate correlation, where the ridge weights can sharply separate large from small coefficients. The two methods converge at $p = 100$, where the differences are sometimes below 1 FP on average. FP for the penalized methods increases with ρ at $p = 50$: at $n = 200, \rho = 0.7$, ALASSO gives 5.22, whereas at $\rho = 0.9$ it gives 6.58. This is the familiar phenomenon that the adaptive weights correct only partially, as correlated noise predictors look more like true signals to the penalty when correlation rises.

The equidispersed case ($\nu = 1$, Table 5) shows the same pattern. ALASSO leads in most cells, with LASSO taking a few at the underdetermined corner ($p = 100, n = 70$). The theoretical maximum of ML is unchanged. The penalized FP values are close to those in Table 4; dispersion has little influence on selection behaviour, as expected, since the L_1 penalty acts on the coefficient scale, with no reference to the working weights.

No qualitative change appears in the underdispersed case ($\nu = 1.5$, Table 6). ALASSO attains the lower count in most cells and LASSO in a minority at high p, while ML again offers no basis for comparison. The FP magnitudes match Tables 4 and 5, reinforcing that ν is essentially irrelevant to false-positive behaviour.

The difference is apparent in Figure 3 (3D surface). As both p and ρ rise, the red ML surface climbs steeply, reaching 80 at the far corner, the combined effect of dimensionality and multicollinearity. The LASSO and ALASSO surfaces stay almost completely flat over the entire p–ρ grid, essentially independent of both dimensions, and there is no overlap between the two surfaces.

The within-penalized comparison appears in Figure 4, where the two penalized methods are compared against ML, shown only as a benchmark label at the top of each panel (24, 40, 80). Both methods stay low and flat (FP $\sim$ 3–5) at $p = 30$, with ALASSO marginally lower than LASSO at all three values of ν. The values rise to 5–8 at $p = 50$, where the two methods are broadly similar with a small gap at lower correlation. At $p = 100$ the error bars

Table 5. False positives (FP) and number of selected variables ($\nu = 1$).

				False positives (FP)			No. selected variables		
n	p	ρ	k	ML	LASSO	ALASSO	ML	LASSO	ALASSO
70	30	0.7	6	24.00 (0.00)	4.30 (3.35)	**2.98 (1.85)**	30.00 (0.00)	9.50 (3.64)	**8.06 (2.07)**
100	30	0.7	6	24.00 (0.00)	4.38 (2.72)	**3.60 (2.44)**	30.00 (0.00)	9.86 (2.93)	**8.90 (2.70)**
200	30	0.7	6	24.00 (0.00)	4.12 (3.12)	**3.04 (2.04)**	30.00 (0.00)	10.02 (3.10)	**8.84 (2.05)**
70	50	0.7	10	40.00 (0.00)	7.36 (4.27)	**6.26 (3.37)**	50.00 (0.00)	14.70 (5.10)	**13.32 (4.09)**
100	50	0.7	10	40.00 (0.00)	6.98 (3.55)	**6.22 (3.20)**	50.00 (0.00)	14.92 (4.06)	**13.98 (3.78)**
200	50	0.7	10	40.00 (0.00)	5.76 (2.85)	**4.74 (2.45)**	50.00 (0.00)	14.98 (2.90)	**13.76 (2.61)**
70	100	0.7	20	80.00 (0.00)	12.46 (4.47)	**11.24 (3.90)**	100.00 (0.00)	22.28 (6.11)	**20.66 (4.83)**
100	100	0.7	20	80.00 (0.00)	11.86 (4.38)	**11.74 (4.44)**	100.00 (0.00)	22.86 (5.30)	**22.84 (5.43)**
200	100	0.7	20	80.00 (0.00)	12.30 (4.13)	**10.94 (3.62)**	100.00 (0.00)	27.38 (4.79)	**25.84 (4.04)**
70	30	0.9	6	24.00 (0.00)	4.90 (1.88)	**4.68 (1.95)**	30.00 (0.00)	8.20 (2.23)	**7.78 (2.33)**
100	30	0.9	6	24.00 (0.00)	4.94 (1.73)	**4.64 (2.59)**	30.00 (0.00)	8.96 (1.71)	**8.42 (2.68)**
200	30	0.9	6	24.00 (0.00)	5.20 (2.05)	**4.70 (2.79)**	30.00 (0.00)	9.78 (2.17)	**8.84 (2.97)**
70	50	0.9	10	40.00 (0.00)	5.96 (2.14)	**5.86 (2.19)**	50.00 (0.00)	10.08 (2.68)	**9.88 (2.69)**
100	50	0.9	10	40.00 (0.00)	7.44 (3.33)	**6.60 (1.94)**	50.00 (0.00)	12.40 (3.69)	**11.26 (2.32)**
200	50	0.9	10	40.00 (0.00)	8.02 (2.70)	**7.02 (2.20)**	50.00 (0.00)	13.90 (2.79)	**12.52 (2.40)**
70	100	0.9	20	80.00 (0.00)	10.68 (2.82)	**10.34 (2.85)**	100.00 (0.00)	15.84 (3.20)	**15.52 (3.39)**
100	100	0.9	20	80.00 (0.00)	10.84 (3.42)	**10.52 (3.30)**	100.00 (0.00)	17.40 (3.76)	**16.94 (3.66)**
200	100	0.9	20	80.00 (0.00)	12.52 (2.97)	**11.30 (2.94)**	100.00 (0.00)	22.02 (3.04)	**20.36 (3.10)**
70	30	0.95	6	24.00 (0.00)	4.46 (1.84)	**4.20 (1.94)**	30.00 (0.00)	7.10 (1.99)	**6.58 (2.13)**
100	30	0.95	6	24.00 (0.00)	5.04 (2.13)	**4.40 (2.16)**	30.00 (0.00)	8.06 (2.27)	**7.12 (2.38)**
200	30	0.95	6	24.00 (0.00)	5.54 (1.99)	**4.28 (2.01)**	30.00 (0.00)	8.90 (2.00)	**7.36 (2.22)**
70	50	0.95	10	40.00 (0.00)	6.04 (2.42)	**5.42 (2.12)**	50.00 (0.00)	9.00 (2.35)	**8.20 (2.27)**
100	50	0.95	10	40.00 (0.00)	6.56 (2.05)	**5.96 (1.83)**	50.00 (0.00)	9.40 (2.14)	**8.48 (1.79)**
200	50	0.95	10	40.00 (0.00)	6.74 (2.46)	**5.64 (1.90)**	50.00 (0.00)	10.84 (2.67)	**9.32 (2.10)**
70	100	0.95	20	80.00 (0.00)	**7.66 (3.09)**	7.84 (3.84)	100.00 (0.00)	**11.70 (2.88)**	11.82 (4.13)
100	100	0.95	20	80.00 (0.00)	8.96 (3.01)	**8.94 (2.61)**	100.00 (0.00)	13.16 (2.80)	**13.16 (2.74)**
200	100	0.95	20	80.00 (0.00)	10.76 (3.07)	**9.70 (2.67)**	100.00 (0.00)	16.52 (3.01)	**15.08 (2.78)**

Note. Mean over 50 replications, Monte Carlo standard deviation in parentheses. Bold: lowest in each block. ML, LASSO, ALASSO abbreviate COMP-ML, COMP-LASSO, COMP-ALASSO. The zero standard deviation for ML is structural, not numerical (Section 4.5).

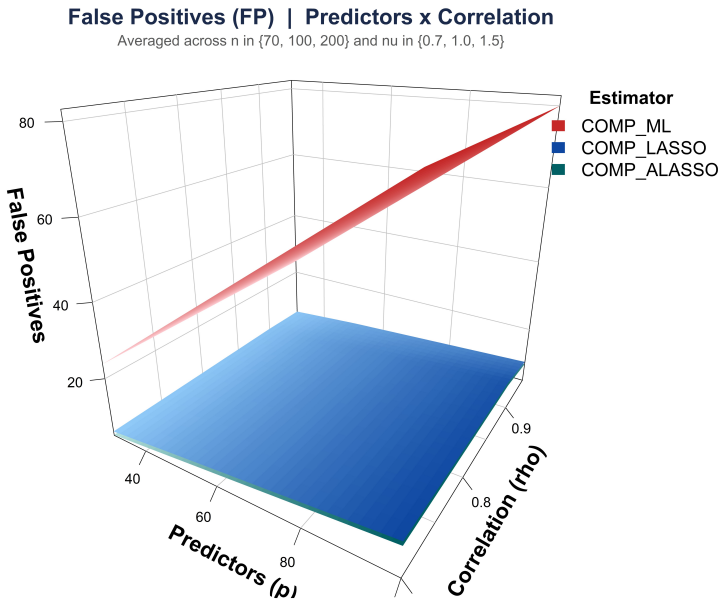

Figure 3. False positives across p and ρ (3D).

Table 6. False positives (FP) and number of selected variables ($\nu = 1.5$).

				False positives (FP)			No. selected variables		
n	p	ρ	k	ML	LASSO	ALASSO	ML	LASSO	ALASSO
70	30	0.7	6	24.00 (0.00)	4.64 (2.95)	**4.18 (3.05)**	30.00 (0.00)	9.82 (3.14)	**9.32 (3.28)**
100	30	0.7	6	24.00 (0.00)	4.96 (3.41)	**4.24 (3.11)**	30.00 (0.00)	10.36 (3.71)	**9.60 (3.37)**
200	30	0.7	6	24.00 (0.00)	4.70 (3.27)	**3.64 (2.60)**	30.00 (0.00)	10.58 (3.25)	**9.44 (2.63)**
70	50	0.7	10	40.00 (0.00)	7.38 (3.37)	**6.86 (3.65)**	50.00 (0.00)	14.50 (4.04)	**13.92 (4.16)**
100	50	0.7	10	40.00 (0.00)	6.54 (2.97)	**6.02 (3.05)**	50.00 (0.00)	14.28 (3.63)	**13.72 (3.57)**
200	50	0.7	10	40.00 (0.00)	7.12 (3.45)	**6.52 (3.04)**	50.00 (0.00)	16.32 (3.64)	**15.58 (3.26)**
70	100	0.7	20	80.00 (0.00)	13.24 (5.58)	**12.26 (5.76)**	100.00 (0.00)	23.30 (6.94)	**21.82 (6.76)**
100	100	0.7	20	80.00 (0.00)	**12.18 (4.83)**	12.88 (5.67)	100.00 (0.00)	**23.48 (6.38)**	24.18 (6.75)
200	100	0.7	20	80.00 (0.00)	12.90 (3.82)	**11.52 (3.46)**	100.00 (0.00)	28.12 (3.90)	**26.54 (3.24)**
70	30	0.9	6	24.00 (0.00)	5.12 (3.64)	**3.82 (1.79)**	30.00 (0.00)	8.64 (4.19)	**7.02 (2.26)**
100	30	0.9	6	24.00 (0.00)	5.94 (3.21)	**5.24 (2.33)**	30.00 (0.00)	9.74 (3.28)	**8.80 (2.26)**
200	30	0.9	6	24.00 (0.00)	4.92 (2.33)	**4.36 (2.34)**	30.00 (0.00)	9.44 (2.41)	**8.70 (2.53)**
70	50	0.9	10	40.00 (0.00)	6.54 (2.65)	**5.84 (2.44)**	50.00 (0.00)	10.60 (2.76)	**9.66 (2.62)**
100	50	0.9	10	40.00 (0.00)	6.80 (2.20)	**6.24 (2.21)**	50.00 (0.00)	11.58 (2.61)	**10.72 (2.61)**
200	50	0.9	10	40.00 (0.00)	8.14 (3.94)	**6.98 (2.67)**	50.00 (0.00)	13.94 (4.46)	**12.26 (2.77)**
70	100	0.9	20	80.00 (0.00)	10.30 (3.14)	**9.80 (3.16)**	100.00 (0.00)	16.10 (3.32)	**15.58 (3.25)**
100	100	0.9	20	80.00 (0.00)	11.58 (3.06)	**11.44 (3.82)**	100.00 (0.00)	17.58 (3.50)	**17.42 (4.31)**
200	100	0.9	20	80.00 (0.00)	12.10 (3.10)	**11.56 (2.74)**	100.00 (0.00)	20.44 (3.64)	**19.56 (3.31)**
70	30	0.95	6	24.00 (0.00)	4.34 (2.11)	**4.24 (2.45)**	30.00 (0.00)	6.72 (2.30)	**6.50 (2.75)**
100	30	0.95	6	24.00 (0.00)	5.18 (2.24)	**4.72 (2.56)**	30.00 (0.00)	8.12 (2.51)	**7.52 (2.79)**
200	30	0.95	6	24.00 (0.00)	5.42 (2.44)	**4.46 (2.62)**	30.00 (0.00)	9.08 (2.73)	**7.74 (2.84)**
70	50	0.95	10	40.00 (0.00)	5.90 (2.05)	**5.52 (2.14)**	50.00 (0.00)	8.38 (2.16)	**7.82 (2.29)**
100	50	0.95	10	40.00 (0.00)	6.64 (3.10)	**5.60 (2.25)**	50.00 (0.00)	10.04 (3.57)	**8.62 (2.52)**
200	50	0.95	10	40.00 (0.00)	7.02 (2.77)	**6.74 (3.24)**	50.00 (0.00)	11.34 (2.81)	**10.88 (3.60)**
70	100	0.95	20	80.00 (0.00)	**7.80 (2.54)**	7.90 (2.70)	100.00 (0.00)	11.30 (2.76)	**11.30 (2.76)**
100	100	0.95	20	80.00 (0.00)	8.78 (2.51)	**8.50 (2.57)**	100.00 (0.00)	12.70 (2.92)	**12.28 (2.81)**
200	100	0.95	20	80.00 (0.00)	10.68 (2.99)	**9.86 (3.18)**	100.00 (0.00)	16.10 (2.96)	**14.98 (3.12)**

Note. Mean over 50 replications, Monte Carlo standard deviation in parentheses. Bold: lowest in each block. ML, LASSO, ALASSO abbreviate COMP-ML, COMP-LASSO, COMP-ALASSO. The zero standard deviation for ML is structural, not numerical (Section 4.5).

are considerably larger, the two estimators are almost identical, and a reversal occurs: FP increases slightly with n rather than decreasing. Larger samples let cross-validation select a less aggressive τ and admit marginally more predictors, a well-known property of the CV-tuned LASSO that becomes visible only when p is large relative to the signal strength.

False positives alone give a one-sided picture, since they can be driven down by shrinking harder, so the simulation also records true positives, false negatives, sensitivity (the proportion of truly nonzero coefficients that are retained) and the F_1 score (the harmonic mean of precision and sensitivity). The sparsity of the penalized estimators proves to be bought at a real cost in recall: averaged over the 81 scenarios, sensitivity is 0.546 for COMP-LASSO and 0.524 for COMP-ALASSO, against 1.000 for COMP-ML, which attains perfect recall trivially by retaining every predictor. The loss is concentrated where the design is hardest, with sensitivity falling from 0.777 and 0.766 at $\rho = 0.70$ to 0.357 and 0.331 at $\rho = 0.95$; under the most severe collinearity considered, roughly two thirds of the truly nonzero coefficients are missed. Once precision and recall are combined, however, the two penalized estimators are indistinguishable. The mean F_1 is 0.478 for both, against 0.333 for COMP-ML, and although COMP-ALASSO attains the lower false-positive count in 76 of the 81 scenarios it attains the higher F_1 in only 34. The adaptive weights therefore purchase sparsity rather than accuracy of selection: both penalized estimators outperform COMP-ML substantially on selection, and differ little from one another.

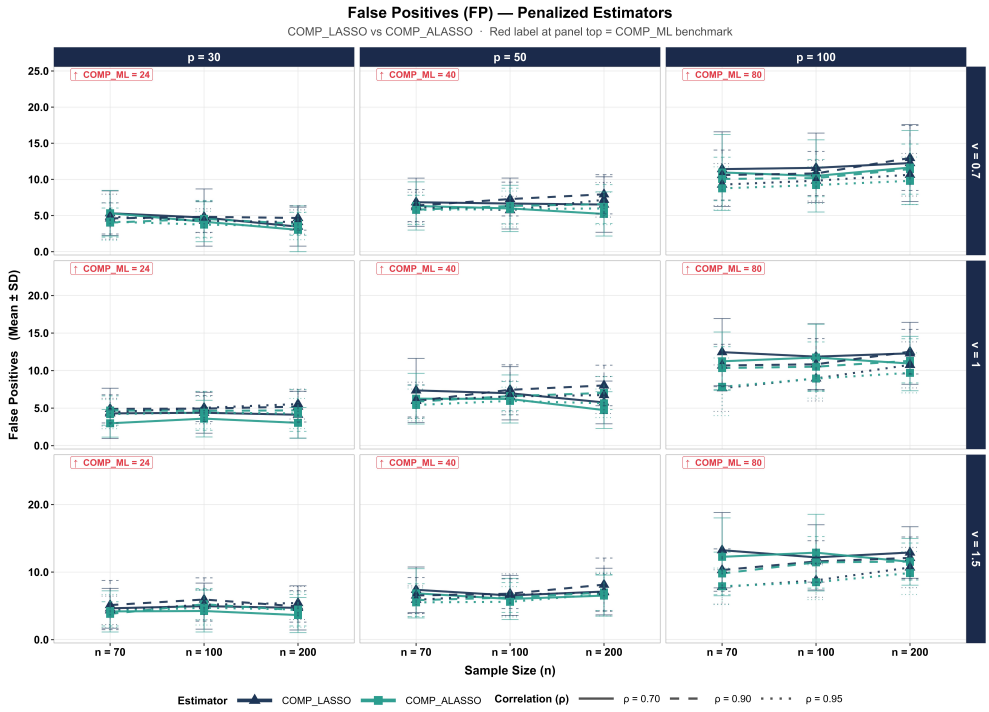

Figure 4. False positives across all scenarios.

4.6. Number of Selected Variables

The number of selected variables is reported in the right-hand block of Tables 4–6, the same tables that carry the false positives. The ML column reports 30.00, 50.00 or 100.00 in every row, with SD $= 0.00$ throughout. ML keeps all p predictors regardless of sample size, correlation, or dispersion. That zero standard deviation does not signal stability; it signals that no selection is taking place at all.

Both penalized methods select far fewer predictors than ML; they overselect relative to the oracle value k, the true number of nonzero coefficients, but only mildly. At $\nu = 0.7$ (Table 4), COMP-ALASSO is the more parsimonious in most cells. The clearest example is the high-correlation block, where ALASSO achieves nearly exact recovery against an oracle of 6 variables, while ML is still at 30 variables at $n = 100, p = 30, \rho = 0.95$. As ρ increases, n_{selected} decreases for both penalized methods across all three p columns: stronger correlation causes the penalty to concentrate on fewer representatives of each correlated cluster, reducing overall selection.

The two penalized methods agree at $p = 100$, where the differences are usually around 1–2 variables. This convergence carries over to Tables 5 and 6 ($\nu = 1$ and $\nu = 1.5$): ALASSO dominates in most cells, LASSO in a few at large p and small n, and the ordering is unaffected by the dispersion parameter. The absolute values in Table 6 are comparable to those in Tables 4 and 5, suggesting that ν is largely irrelevant to selection behaviour.

Figure 5 summarizes the complete picture. The red ML circles never shift with n; they sit at a fixed height above the oracle, at 24 for $p = 30$, 40 for $p = 50$, and 80 for $p = 100$. The LASSO and ALASSO lollipops sit barely above 0 at $p = 30$ and $p = 50$, with ALASSO slightly closer to the oracle. The $p = 100$ column is the most informative, where both penalized methods underselect below zero, into the teal zone. This is not a deficiency: removing several noise predictors will on occasion set a true signal to zero as well, and the RMSE results show that this trade is worthwhile at the most aggressive penalization. The underselection is small and shrinks to zero as n grows. This is the direction the selection-consistency theory of the adaptive LASSO would predict, although for the reasons given in Section 3.3 that theory is not established for the present estimators and the pattern is reported as a finite-sample observation only.

Figure 5. Variable selection accuracy.

5. Empirical Applications

5.1. *Datasets and Setup*

The proposed estimators are tested on three real datasets from the existing literature, chosen to span distinct regimes: severe collinearity at low p, a large n with moderate collinearity, and genuine high dimensionality with $p > n$. Their characteristics are summarized in Table 7. For each dataset the table reports the sample mean, the sample variance and their ratio, alongside the range of the response and the collinearity diagnostics. These quantities characterize the dispersion before any model is fitted, and provide the benchmark against which the estimated dispersion parameters of Sections 5.2–5.4 should be read. All three outcomes are count variables with variance-to-mean ratios from 3.66 to 7.49, far above 1 in every case, making a COM-Poisson model a natural choice and a standard Poisson model a poor one. The raw distributions are plotted in Figure 6. CreditCard is strongly zero-heavy, with 1060 zeros in 1319 observations; CrohnD has 50 zeros in 117; and BCI has none, its response being an abundance count that is bounded away from zero in every plot.

The CrohnD data record 117 patients enrolled in a clinical trial for Crohn's disease [27]. The response `nrAdvE` is the number of adverse events experienced by a patient over the course of the trial, and the substantive question is whether treatment assignment affects the adverse-event rate once body composition and demographics are accounted for. The eight predictors are BMI, height, weight, age, sex, country, and two indicator variables coding the three-level treatment factor (placebo, drug 1, drug 2). The severity of the collinearity here is not an accident of sampling but a definitional consequence: BMI is constructed as weight in kilograms divided by the square of height in metres, so BMI, weight and height are functionally dependent by construction. This is why `weight` carries a variance inflation factor of 106.18. Deleting one of the three is unattractive on clinical grounds, since BMI is the standard summary of body composition while weight and height separately carry information about frame

and dosing; a method that can shrink and select among mutually redundant but individually meaningful predictors is preferable to one that forces the analyst to discard a variable a priori.

The CreditCard data contain 1319 credit card applicants [20]. The response `reports` is the number of major derogatory reports on an applicant's credit history, and the question of interest is which applicant characteristics predict a record of delinquency. The eleven predictors are age, yearly income, average monthly credit card expenditure, the expenditure-to-income share, number of dependents, months at the current address, number of major cards held, number of active credit accounts, home ownership, self-employment status, and whether the application was accepted. As in CrohnD, the strongest dependence is definitional rather than incidental: `share` is by construction monthly expenditure divided by yearly income, so the three variables satisfy an exact relationship, which is why `expenditure` carries the largest variance inflation factor (5.33) and the `share`–`expenditure` pair is the most strongly correlated. The collinearity is real but far milder than in CrohnD, which is what makes this dataset a useful intermediate case.

The BCI data are the Barro Colorado Island tree census of Condit et al. [13], distributed with the `vegan` package [32]. It records the number of stems of each of 225 species in 50 contiguous one-hectare plots of tropical forest in Panama. The response is the abundance of *Faramea occidentalis*, the most abundant species in the plot, and the 224 predictors are the abundances of the remaining species. The ecological question is which co-occurring species predict the abundance of a focal species, the basic quantity in joint species distribution modelling. The difficulty is of a different kind from the other two applications: there is no pathological pair of definitionally related variables, but rather $p/n = 4.48$, compounded by extreme sparsity in the design, since 40 of the 224 species occur in two plots or fewer while only six occur in all fifty. Perfectly correlated columns arise as a by-product, since *Cupania cinerea* and *Miconia elata* each appear exactly once, in the same plot, so that their sample correlation is unity, and no unpenalized fit can distinguish between them. A method that reduces 224 candidate species to an interpretable subset is what the problem requires, and what maximum likelihood cannot supply.

All 50 plots are of equal area, so total stem count is an ecological quantity rather than an artifact of sampling effort. It varies only 1.8-fold across plots and correlates with the response at -0.257, and a model using log total abundance as its sole predictor attains a cross-validated RMSE of 15.89 against 16.04 for an intercept-only model. There is therefore no aggregate nuisance variable driving the results, and no offset is required. Whatever predictive gain the penalized estimators achieve must come from the identities of the individual species rather than from overall plot richness.

Table 7. Empirical dataset characteristics.

Characteristic	Application 1: Crohn's disease	Application 2: Credit card	Application 3: BCI forest census
Dependent variable (Y)	nrAdvE (no. of adverse events)	reports (no. of major derogatory reports)	Stems of *Faramea occidentalis*
Context / field	Clinical trial (Crohn's disease)	Consumer credit scoring	Tropical forest community ecology
Sample size (n)	117	1319	50
No. of predictors (p)	8	11	224
DV mean	2.034	0.456	34.340
DV variance	7.447	1.810	257.290
Variance / mean ratio	3.661	3.965	7.492
DV range (min–max)	0–12	0–14	7–80
Multicollinearity: max $\lvert r\rvert$	0.822	0.839	1.000
Multicollinearity: max VIF	106.18 (weight)	5.33 (expenditure)	n/a ($p > n$)
Collinearity source	Strong pairwise (weight–BMI)	Strong pairwise (share–expenditure)	High dimensionality and design sparsity
Source / origin	`robustbase` (R)	`AER` (Greene)	`vegan` (R)
Prior count-model literature	[27, 26]	[20, 11]	[13, 32]

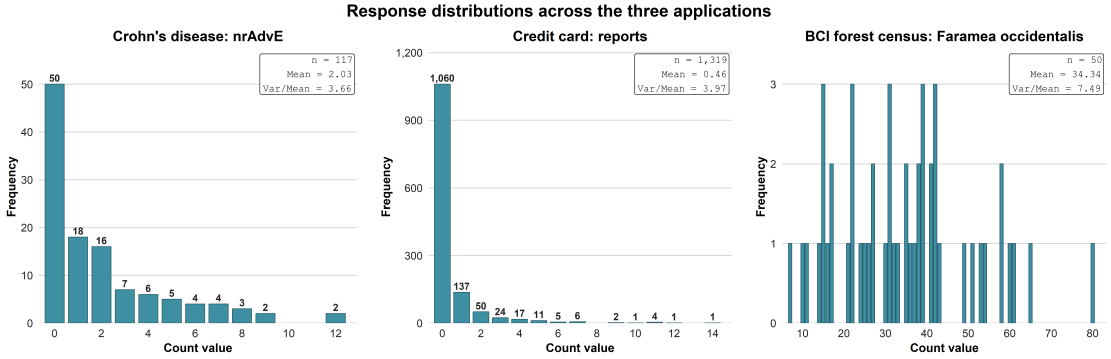

Figure 6. Response distributions across the three applications. Frequencies are printed above each bar where the support is sparse enough to carry them; the BCI response takes 34 distinct values across 50 plots and is shown without labels.

Each dataset is evaluated by 100 independent repetitions of 5-fold cross-validation, rather than a single partition, so that the reported figures are not sensitive to one particular assignment of observations to folds. Table 8 reports the mean RMSE and mean number of retained predictors over those 100 repetitions, with the corresponding standard deviations, together with the dispersion parameter ν estimated from the fit to the complete dataset. Predictors are standardized to zero mean and unit variance before fitting, so that the common penalty threshold applies comparably across coordinates; the intercept and ν are left unpenalized.

Table 8. Empirical performance across the applications.

Dataset	Method	RMSE (SD)	No. selected (SD)	Total p	Dispersion ν
CrohnD (n=117, p=8)	ML	5.0173 (7.2277)	8 (0)	8	0.20
	LASSO	2.9958 (2.6077)	6.3 (1.06)		0.75
	ALASSO	**2.6482 (0.7514)**	6.1 (1.21)		0.75
CreditCard (n=1319, p=11)	ML	2.6715 (3.0365)	11 (0)	11	0.20
	LASSO	**1.1887 (0.3252)**	2.0 (0)		0.70
	ALASSO	1.3076 (0.5896)	2.41 (1.38)		0.75
BCI (n=50, p=224)	ML	14.5941 (5.7712)	221.74 (1.74)	224	1.15
	LASSO	12.2133 (5.2277)	28.39 (24.17)		1.00
	ALASSO	**12.0984 (5.2535)**	28.61 (28.10)		1.00

Note. Mean over 100 repetitions of 5-fold cross-validation, standard deviation in parentheses. Unlike Tables 1–6, where the parenthetical value is a Monte Carlo standard deviation over 50 replications. ν is from the fit to the complete dataset. Bold: lowest RMSE per dataset.

5.2. Application 1: Crohn's Disease (CrohnD)

This application isolates the effect of severe collinearity. The problem is low-dimensional, with $p = 8$ predictors and $n = 117$ observations, but the weight variable has a VIF of 106.18, and the highest pairwise correlation is 0.822. ML's RMSE is 5.017 with an SD of 7.277. The magnitude of that standard deviation is itself informative, reflecting instability in the coefficient estimates rather than sampling variation alone.

COMP-ALASSO achieves a 47% reduction (RMSE $= 2.648$) and COMP-LASSO a 40% reduction (RMSE $= 2.996$), and both settle on a reduced set of about 6 predictors rather than all 8.

The identity of the discarded predictors is informative. On the fit to the complete dataset both penalized estimators retain BMI, height, country, sex, age and the indicator for the first treatment arm, and both set the coefficients of `weight` and of the second treatment indicator exactly to zero. That `weight` is the variable dropped is precisely what the collinearity structure described in Section 5.1 would lead one to expect: since BMI is weight divided by the square of height, retaining BMI and height renders weight redundant, and the penalty identifies this

without the analyst having to impose it. Clinically, the more consequential point is that the two treatment arms are not treated alike, since the first survives selection with a negative coefficient, indicating fewer adverse events relative to placebo, while the second is shrunk to zero, indicating no detectable effect on the adverse-event rate once body composition and demographics are accounted for. The unpenalized fit, which retains both indicators, offers no basis for that distinction. The dispersion estimate warrants closer attention. ML returns $\nu = 0.20$, whereas both penalized methods recover $\nu = 0.75$, a credible moderate overdispersion given the raw variance-to-mean ratio of 3.66. The ML value is not an interior maximum. The profile likelihood for ν increases monotonically towards the lower end of the search range, and the reported 0.20 is the boundary of that range, attained in every one of the 100 cross-validation repetitions without variation. The unpenalized fit therefore reports not an estimate of the dispersion but the absence of one: once the coefficients are destabilized by collinearity, ν is not identified and the likelihood offers no interior optimum. The penalized fits return 0.75, which lies well inside the search range and is consistent with the observed variance-to-mean ratio. A dispersion estimate lying exactly on a search boundary is thus better read as a diagnostic of failure than as a measurement. When the information matrix is distorted by collinearity, the damage falls not only on β but also on ν.

5.3. Application 2: Credit Card (CreditCard)

This is the largest dataset, and the one on which LASSO performs best of the three. With $n = 1319$ and $p = 11$, ML stabilizes somewhat, with RMSE dropping to 2.672. LASSO more than halves it to 1.189, and ALASSO follows at 1.308. The more surprising result is the selection: LASSO retains exactly 2 of the 11 predictors, reducing the model to 18.2% of its original size at substantially lower prediction error, while ALASSO averages 2.41.

The two predictors COMP-LASSO retains are the number of active credit accounts and the indicator for whether the application was accepted. Neither `share` nor `expenditure` survives, so the penalty resolves the definitional dependence among income, expenditure and their ratio by discarding all three rather than by arbitrating between them, a reasonable outcome given that none of them carries information about delinquency that the account variables do not already supply. One caveat should be attached to the second retained predictor: acceptance of a credit card application is plausibly determined in part by the applicant's record of derogatory reports, so its inclusion is defensible for prediction, which is the criterion used here, but it should not be read causally. Both recover $\nu \approx 0.70$–0.75, in agreement with the raw variance-to-mean ratio of 3.97.

Notably, LASSO outperforms ALASSO here. With only moderate collinearity (max $|r| = 0.839$, max VIF $= 5.33$) and a large sample, the adaptive re-weighting has little additional effect: the uniform penalty is already sufficient to isolate the two strong predictors, so the additional weighting introduces bias without a corresponding gain in selection.

5.4. Application 3: Barro Colorado Island Forest Census (BCI)

This is the high-dimensional case, with $p = 224$, $n = 50$ and $p/n = 4.48$. COMP-ML cannot perform selection and retains 221.74 predictors on average, returning RMSE $= 14.594$. COMP-LASSO reduces this to 12.213 using 28.39 species and COMP-ALASSO to 12.098 using 28.61, reductions of 16.3% and 17.1% respectively. Because all three estimators are evaluated on identical folds, the appropriate comparison is paired: over the 100 fold-level evaluations the mean advantage of COMP-LASSO over COMP-ML is 2.381 with a standard error of 0.407, and that of COMP-ALASSO is 2.496 with a standard error of 0.397, so both advantages are approximately six standard errors wide. The difference between the two penalized estimators is 0.115 with a standard error of 0.138; they are not distinguishable, and we do not claim a ranking between them.

The comparison that matters most for this application is not against COMP-ML but against a model that ignores species identity altogether. An intercept-only model attains a cross-validated RMSE of 16.04, and adding log total plot abundance as a single predictor improves this only to 15.89. The penalized estimators reach 12.10–12.21, so roughly 24% of the predictive error is removed relative to the null and almost all of that gain survives conditioning on overall plot abundance. The information is therefore carried by which species are present, not by how many stems a plot contains, an interpretation available here precisely because the plots are of equal area.

Selection is markedly less stable than in the two low-dimensional applications. The number of retained species has a standard deviation of 24.17 about a mean of 28.39 for COMP-LASSO, and across individual folds the

count ranges from 6 to 215. The fit to the complete dataset retains 27 species for COMP-LASSO and 46 for COMP-ALASSO. This behaviour is characteristic of $p > n$ designs in which many columns carry nearly the same information: 40 of the 224 species occur in at most two plots, several pairs of singleton species are perfectly correlated, and the penalty has little basis for preferring one member of such a pair to the other. We report the instability rather than suppress it, since a single selected set from a single fit would convey more confidence than the design supports; selection frequencies across resamples are the appropriate summary, and the aggregate model size is far more reproducible than its composition.

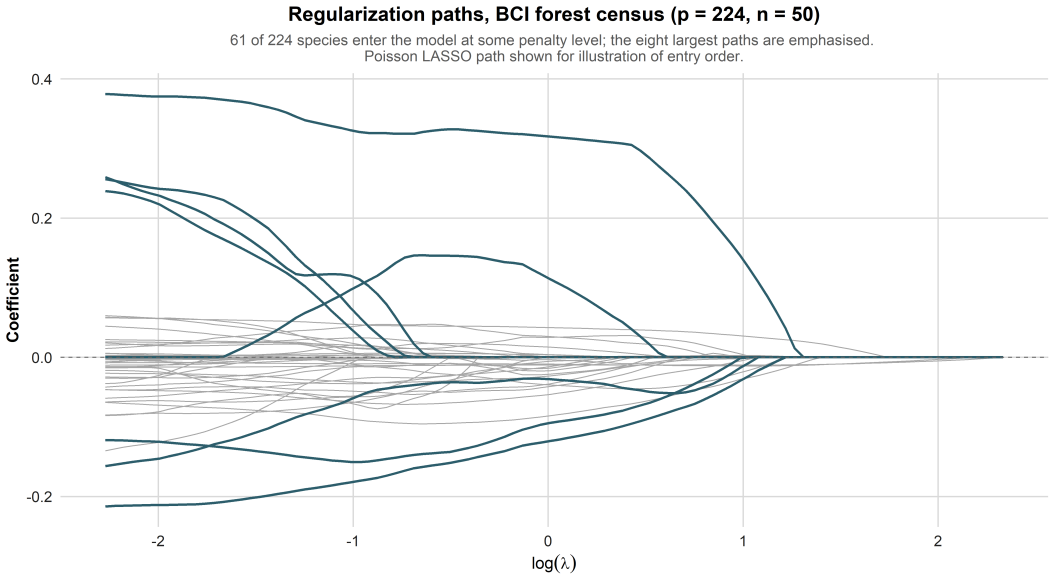

Figure 7. Regularization paths for the BCI forest census. The eight paths of largest amplitude are emphasised; species that never enter the model at any penalty level are shown in light grey. A Poisson LASSO path is displayed to illustrate the order and amplitude with which predictors enter, and is not the fitted COM-Poisson model of Table 8.

The regularization paths in Figure 7 show how this sparsity arises. Of the 224 species, 61 enter the model at some point along the penalty path while 163 are never selected at any penalty level. Under a strong penalty a single species carries almost all of the signal, its coefficient near 0.30 while every other path is at or near zero; as the penalty relaxes, three further species rise to comparable amplitude and a group of negative coefficients separates from the mass. The profile is one of graded rather than sharp sparsity: no small set of species dominates uniformly across the path, which is consistent with the instability of the selected set reported above and with the ecological expectation that many co-occurring species carry weakly overlapping information about a focal species.

The dispersion estimates require comment, because they behave differently from the two applications above. The response is strongly overdispersed marginally, with a variance-to-mean ratio of 7.49, yet both penalized estimators return $\hat{\nu} = 1.00$ and COMP-ML returns 1.15. This is not a failure of the profile search: the reported values are genuine maxima of the profile likelihood, and re-evaluating them with the truncation point M raised from 150 to 3000 leaves them unchanged. The explanation is that ν measures conditional rather than marginal dispersion. With 224 available predictors and 50 observations the fitted linear predictor varies enough in sample to absorb the marginal overdispersion, leaving a conditional distribution close to Poisson. The estimate is thus reporting a property of the fitted model rather than of the raw response, and the gap between $\hat{\nu}$ and the variance-to-mean ratio should be read as a signature of a rich predictor set rather than as a calibration error. Section 5.5 returns to the practical consequences of this distinction.

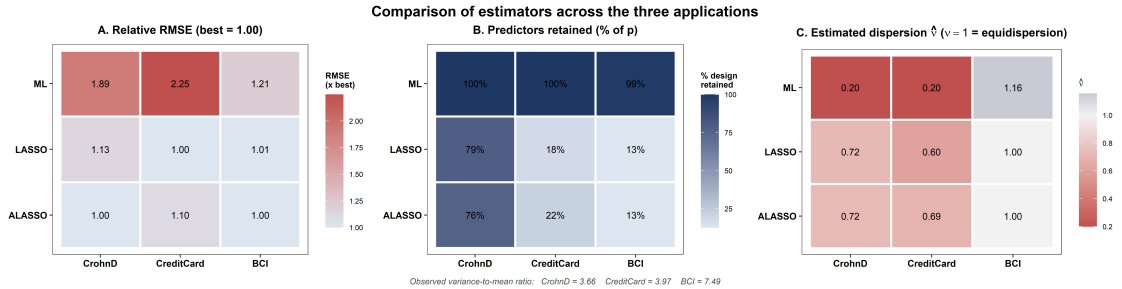

Figure 8. Estimator comparison across the applications. Panel A: RMSE relative to the best method within each dataset. Panel B: percentage of the design retained. Panel C: estimated dispersion $\hat{\nu}$, with the observed variance-to-mean ratio of each response printed beneath for comparison.

5.5. Cross-Application Comparison and Practical Considerations

The three datasets are compared in Figure 8, in which Panel A shows RMSE relative to the best method within each dataset: ML is the least accurate in all three, most markedly on CreditCard, where its error is 2.25 times that of the best. Panel B shows the percentage of predictors retained: effectively the whole design for ML, which has no selection capacity, against 76–79% on CrohnD, 18–22% on CreditCard and 12.7–12.8% on BCI for the penalized methods.[†] Panel C reports the estimated dispersion alongside the observed variance-to-mean ratio of each response, and shows the pattern discussed below: the penalized estimates lie between the COMP-ML value and unity on the two low-dimensional applications, whereas on BCI all three estimators sit at or near equidispersion despite the largest observed ratio of the three.

The three applications span genuinely different regimes, namely small n with severe collinearity, large n with moderate collinearity, and high dimensionality with $p > n$, yet the conclusion is the same in each. Penalized estimation reduces prediction error by 17–56% and selects far sparser models. The dispersion estimates require a more qualified statement. On the two low-dimensional applications the penalized estimators recover values consistent with the observed variance-to-mean ratio while COMP-ML does not; on BCI all three estimators report near-equidispersion despite marked marginal overdispersion, for the reasons set out in Section 5.4. What the applications establish uniformly is the advantage in prediction and in model size; the behaviour of $\hat{\nu}$ depends on the dimension of the design, as set out below. In these two respects the simulation findings are borne out in practice.

Fitting these datasets also exposed four difficulties that a simulation study does not reproduce. First, collinearity in real data is frequently definitional rather than incidental: in both CrohnD and CreditCard the worst-conditioned variables stand in an exact algebraic relationship to others in the design (BMI to weight and height, share to expenditure and income). Collinearity induced through a correlation matrix implies that some rotation or deletion would resolve it, whereas definitional dependence cannot be resolved without discarding a variable the subject-matter audience expects to see. This is an argument for penalization over pre-screening that the simulation results alone do not make. Second, predictor scaling is not cosmetic: the CreditCard predictors span income in units of ten thousand dollars, expenditure in dollars and durations in months, and since the L_1 penalty applies a common threshold across coordinates, an unstandardized design penalizes small-scale variables more heavily without this being apparent from the output. Third, the selected set is unstable in the highest-dimensional application in size as well as composition: across folds the number of retained predictors has a standard deviation of 24.17 against a mean of 28.39, with individual folds ranging from 6 to 215, so selection frequencies across resamples are a more informative summary than a single fitted set. Fourth, the computational burden is substantial, since each fit nests

[†]On BCI the mean retention for COMP-ML is 98.99% rather than exactly 100%. A coefficient is recorded as selected when its magnitude exceeds the numerical tolerance 10^{-8}, and in a $p > n$ design a small number of coefficients fall below that threshold in some folds without being set to zero by any thresholding operator. The structural argument of Section 4.5 is unaffected.

coordinate descent inside an iteratively reweighted loop inside a profile search over ν; warm-starting along the τ path is a practical necessity rather than an optimization nicety.

The interpretation of $\hat{\nu}$ requires a rule conditional on the design. When p is small relative to n, the fitted linear predictor absorbs little of the marginal variation and $\hat{\nu}$ should track the observed variance-to-mean ratio; a value that does not, or that sits on the boundary of the search range as COMP-ML does on CrohnD, is a diagnostic of a failed fit. When p approaches or exceeds n this reasoning no longer applies, since a sufficiently rich predictor can drive the conditional dispersion towards unity whatever the marginal dispersion, as it does on BCI. Compare $\hat{\nu}$ against the variance-to-mean ratio in low-dimensional problems and treat disagreement as a warning; in high-dimensional problems read $\hat{\nu}$ as a property of the fitted model rather than of the response. Treating a boundary solution as uninformative is safe in both regimes.

6. Conclusions

In this paper, we proposed two new L_1-penalized estimators for COM-Poisson regression, COMP-LASSO and COMP-ALASSO, developed directly from the penalized COM-Poisson log-likelihood and optimized with an iteratively reweighted coordinate descent algorithm that preserves the true dispersion structure at each iteration. Across 81 simulation scenarios, the penalized methods outperformed ML in prediction in 78 cases. Averaged across scenarios, prediction error falls by 39% and false-positive selections by 85%; pooling across the design before taking the ratio, which weights the severely ill-conditioned cells more heavily, gives 55% and 86% respectively. In the three remaining scenarios, all at $n = 70$ under underdispersion, the estimators are statistically indistinguishable once Monte Carlo error is taken into account rather than genuinely favourable to ML. The gains are most pronounced exactly where they are needed most, under high dimensionality and strong multicollinearity, where ML deteriorates most rapidly. These improvements were replicated in three empirical applications: a clinical trial of Crohn's disease, a consumer-credit dataset, and a tropical forest census with $p = 224$ and $n = 50$, where RMSE was reduced by 17–56% and the retained model was between one eighth and four fifths of the original size.

The behaviour of the dispersion estimate deserves particular emphasis, because it is not uniform across the three applications and the difference is instructive. On CrohnD, where collinearity is severe but p is small, the unpenalized profile likelihood has no interior maximum in ν at all: it runs to the lower boundary of the search range in every cross-validation repetition, reporting not an estimate but the absence of one, while the penalized fits return 0.75 in agreement with the observed variance-to-mean ratio. On BCI, where $p > n$, all three estimators report near-equidispersion despite a marginal variance-to-mean ratio of 7.49; these are genuine profile maxima, invariant to the series truncation, and they reflect a fitted linear predictor rich enough to absorb the marginal overdispersion. The two findings are complementary rather than contradictory. Collinearity at low dimension degrades the dispersion estimate to the point of non-identification, which penalization repairs; high dimensionality shifts what the parameter is measuring, from a property of the response to a property of the fit, which penalization does not repair and should not be expected to. Reporting $\hat{\nu}$ without reference to p/n is therefore unsafe, and unpenalized COM-Poisson regression in high dimensions can mislead not only about which predictors matter but about the distributional character of the outcome. Future work will extend this framework to zero-inflated COM-Poisson models, introduce elastic-net mixing for cases where pure sparsity is overly aggressive, and establish the oracle properties noted in Section 3.3 under primitive conditions, in particular by constructing an initial estimator whose consistency for β does not depend on the Poisson approximation.

REFERENCES

1. M. R. Abonazel, *New modified two-parameter Liu estimator for the Conway–Maxwell Poisson regression model*, Journal of Statistical Computation and Simulation, vol. 93, no. 12, pp. 1976–1996, 2023.
2. M. R. Abonazel, S. M. El-Sayed, and M. M. Seliem, *Developing two penalized spline estimators for semiparametric Poisson and zero-inflated Poisson models: Simulation and application*, Calcutta Statistical Association Bulletin, 2025.
3. M. R. Abonazel, S. M. El-sayed, and O. M. Saber, *Performance of robust count regression estimators in the case of overdispersion, zero inflated, and outliers: Simulation study and application to German health data*, Communications in Mathematical Biology and Neuroscience, vol. 2021, Article ID 55, 2021.

4. M. R. Abonazel, E. E. M. Ebrahim, A. A. Saber, and A. R. Azazy, *On new ridge estimators of the Conway–Maxwell Poisson model in the case of highly correlated predictor variables: Application to plywood quality data*, International Journal of Innovative Research and Scientific Studies, vol. 8, no. 5, pp. 603–614, 2025.
5. M. R. Abonazel, F. A. Awwad, E. Tag Eldin, B. G. Kibria, and I. G. Khattab, *Developing a two-parameter Liu estimator for the COM–Poisson regression model: Application and simulation*, Frontiers in Applied Mathematics and Statistics, vol. 9, 956963, 2023.
6. Z. Y. Algamal, and M. H. Lee, *Penalized Poisson regression model using adaptive modified elastic net penalty*, Electronic Journal of Applied Statistical Analysis, vol. 8, no. 2, pp. 236–245, 2015.
7. Z. Y. Algamal, *Variable selection in count data regression model based on firefly algorithm*, Statistics, Optimization & Information Computing, vol. 7, no. 2, pp. 520–529, 2019.
8. H. Alrweili, *Liu-type estimator for the Poisson-inverse Gaussian regression model: simulation and practical applications*, Statistics, Optimization & Information Computing, vol. 12, no. 4, pp. 982–1003, 2024.
9. P. J. Bickel, Y. Ritov, and A. B. Tsybakov, *Simultaneous analysis of Lasso and Dantzig selector*, Annals of Statistics, vol. 37, no. 4, pp. 1705–1732, 2009.
10. P. Bühlmann, and S. van de Geer, *Statistics for High-Dimensional Data: Methods, Theory and Applications*, Springer, 2011.
11. A. C. Cameron, and P. K. Trivedi, *Count data models for financial data*, in Handbook of Statistics, G. S. Maddala and C. R. Rao (Eds.), vol. 14, pp. 363–391, Elsevier, 1996.
12. A. C. Cameron, and P. K. Trivedi, *Regression Analysis of Count Data*, 2nd ed., Cambridge University Press, 2013.
13. R. Condit, N. Pitman, E. G. Leigh, J. Chave, J. Terborgh, R. B. Foster, P. Nunez, S. Aguilar, R. Valencia, G. Villa, H. C. Muller-Landau, E. Losos, and S. P. Hubbell, *Beta-diversity in tropical forest trees*, Science, vol. 295, no. 5555, pp. 666–669, 2002.
14. R. W. Conway, and W. L. Maxwell, *A queuing model with state dependent service rates*, Journal of Industrial Engineering, vol. 12, no. 2, pp. 132–136, 1962.
15. A. ElSheikh, M. R. Abonazel, and M. C. Ali, *A review of penalized regression and machine learning methods in high-dimensional data*, The Egyptian Statistical Journal, vol. 69, no. 1, pp. 250–261, 2025.
16. D. E. Farrar, and R. R. Glauber, *Multicollinearity in regression analysis: The problem revisited*, Review of Economics and Statistics, vol. 49, no. 1, pp. 92–107, 1967.
17. C. Flexeder, *Generalized Lasso regularization for regression models*, Diploma Thesis, Department of Statistics, Ludwig-Maximilians-Universität München, 2010.
18. J. Friedman, T. Hastie, and R. Tibshirani, *Pathwise coordinate optimization*, Annals of Applied Statistics, vol. 1, no. 2, pp. 302–332, 2007.
19. J. Friedman, T. Hastie, and R. Tibshirani, *Regularization paths for generalized linear models via coordinate descent*, Journal of Statistical Software, vol. 33, no. 1, pp. 1–22, 2010.
20. W. H. Greene, *Accounting for excess zeros and sample selection in Poisson and negative binomial regression models*, Working Paper EC-94-10, Stern School of Business, New York University, 1994.
21. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed., Springer, 2009.
22. T. Hastie, R. Tibshirani, and M. J. Wainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*, CRC Press, 2015.
23. J. M. Hilbe, *Modeling Count Data*, Cambridge University Press, 2014.
24. A. E. Hoerl, and R. W. Kennard, *Ridge regression: Biased estimation for nonorthogonal problems*, Technometrics, vol. 12, no. 1, pp. 55–67, 1970.
25. A. Huang, *Mean-parametrized Conway–Maxwell–Poisson regression models for dispersed counts*, Statistical Modelling, vol. 17, no. 6, pp. 359–380, 2017.
26. M. Jaenada, and L. Pardo, *Robust statistical inference in generalized linear models based on minimum Rényi pseudodistance estimators*, Entropy, vol. 24, no. 1, 123, 2022.
27. S. N. Lô, and E. Ronchetti, *Robust and accurate inference for generalized linear models*, Journal of Multivariate Analysis, vol. 100, no. 9, pp. 2126–2136, 2009.
28. D. Lord, and S. R. Geedipally, *The Conway–Maxwell–Poisson distribution: Fitting statistical traffic accident data*, Accident Analysis & Prevention, vol. 43, no. 6, pp. 2074–2082, 2011.
29. W. Lu, Z. Zhu, and H. Lian, *High-dimensional quantile tensor regression*, Journal of Machine Learning Research, vol. 21, no. 250, pp. 1–31, 2020.
30. A. F. Lukman, B. Aladeitan, K. Ayinde, and M. R. Abonazel, *Modified ridge-type for the Poisson regression model: Simulation and application*, Journal of Applied Statistics, vol. 49, no. 8, pp. 2124–2136, 2022.
31. P. McCullagh, and J. A. Nelder, *Generalized Linear Models*, 2nd ed., Chapman & Hall, 1989.
32. J. Oksanen, G. L. Simpson, F. G. Blanchet, R. Kindt, P. Legendre, P. R. Minchin, R. B. O'Hara, P. Solymos, M. H. H. Stevens, E. Szoecs, and H. Wagner, *vegan: community ecology package*, R package, version 2.7-5, 2025.
33. M. Y. Park, and T. Hastie, *L_1-regularization path algorithm for generalized linear models*, Journal of the Royal Statistical Society: Series B, vol. 69, no. 4, pp. 659–677, 2007.
34. K. F. Sellers, S. Borle, and G. Shmueli, *The COM-Poisson model for count data: A survey of methods and applications*, Applied Stochastic Models in Business and Industry, vol. 28, no. 2, pp. 104–116, 2012.
35. K. F. Sellers, and G. Shmueli, *A flexible regression model for count data*, Annals of Applied Statistics, vol. 4, no. 2, pp. 943–961, 2010.
36. G. Shmueli, T. P. Minka, J. B. Kadane, S. Borle, and P. Boatwright, *A useful distribution for fitting discrete data: Revival of the Conway–Maxwell–Poisson distribution*, Journal of the Royal Statistical Society: Series C, vol. 54, no. 1, pp. 127–142, 2005.
37. I. M. Taha, *Handling Multicollinearity Problem in Generalised Linear Models*, Unpublished Master's Thesis, Cairo University, 2019.
38. R. Tibshirani, *Regression shrinkage and selection via the lasso*, Journal of the Royal Statistical Society: Series B, vol. 58, no. 1, pp. 267–288, 1996.

39. A. W. van der Vaart, *Asymptotic Statistics*, Cambridge University Press, 1998.
40. R. W. M. Wedderburn, *Quasi-likelihood functions, generalized linear models, and the Gauss–Newton method*, Biometrika, vol. 61, no. 3, pp. 439–447, 1974.
41. A. H. Youssef, M. R. Abonazel, and E. G. Ahmed, *Robust M estimation for Poisson panel data model with fixed effects: method, algorithm, simulation, and application*, Statistics, Optimization & Information Computing, vol. 12, no. 5, pp. 1292–1305, 2024.
42. H. H. Zhang, and W. Lu, *Adaptive Lasso for Cox's proportional hazards model*, Biometrika, vol. 94, no. 3, pp. 691–703, 2007.
43. H. Zou, *The adaptive lasso and its oracle properties*, Journal of the American Statistical Association, vol. 101, no. 476, pp. 1418–1429, 2006.