Pak.j.stat.oper.res. Vol. 22 No. 3 2026 pp 317-324 DOI: http://dx.doi.org/10.18187/pjsor.v22i3.4821

Pakistan Journal of Statistics and Operation Research

A comparison of the discrimination performance of lasso and maximum likelihood estimation in logistic regression models

Gilberto P. Alcântara Junior[1], Gustavo H. A. Pereira[1]

1. Department of Statistics, Federal University of São Carlos, São Carlos, Brazil

Abstract

Logistic regression is widely used in many areas of knowledge. Several works compare the performance of lasso and maximum likelihood estimation in logistic regression. However, part of these works do not perform simulation studies and the remaining ones do not consider scenarios in which the ratio of the number of covariates to sample size is high. In this work, we compare the discrimination performance of lasso and maximum likelihood estimation in logistic regression using simulation studies and applications. Variable selection is done both by lasso and by stepwise when maximum likelihood estimation is used. We consider a wide range of values for the ratio of the number of covariates to sample size. The main conclusion of the work is that lasso has a better discrimination performance than maximum likelihood estimation when the ratio of the number of covariates to sample size is high.

Key Words: lasso; logistic regression; maximum likelihood estimation; stepwise.

1. Introduction

Logistic regression is a widely used model for binary responses. Part of its success lies in the fact that it has interpretable parameters. The parameters in logistic regression are usually estimated by the maximum likelihood method (Hosmer Jr et al., 2013). Stepwise is commonly used for variable selection (Gaebel et al., 2016; Müller et al., 2020).

Lasso (Tibshirani, 1996) is an estimation method that can be used in many regression models. It is often used when prediction is the main purpose of model development, because it usually produces better predictions than traditional methods (Hastie et al., 2019). In addition, unlike the maximum likelihood method, it can be used when the number of covariates is greater than the number of observations. Another interesting feature of lasso is that it also performs variable selection, because the estimates of several parameters are usually zero.

Several works compare lasso, the maximum likelihood method and other methods using real datasets with binary response (see for example Steyerberg et al. (2000), Greenwood et al. (2020) and Liu et al. (2021)). In general, they concluded that lasso has a better predictive performance than maximum likelihood estimation, especially when the sample size is small compared to the number of covariates.

Few works compare lasso and the maximum likelihood method using extensive simulation studies in logistic regression. Van Calster et al. (2020), Riley et al. (2021) and Martin et al. (2021) compared these methods and others. They concluded that lasso and other shrinkage methods do not guarantee improved performance. However, they did not consider scenarios in which the ratio of the number of covariates to sample size is greater than 0.2. In addition, they focused on the estimation of the probability of the response assuming each outcome instead of studying the discrimination performance of the methods. In many real problems, the goal is the latter and not the former. Finally, these works did not evaluate the maximum likelihood method associated with a variable selection method as it is usually used in practice. Pavlou et al. (2016) and Leeuwenberg et al. (2022) also compared lasso, the maximum likelihood estimation and other methods in logistic regression using simulation studies. However, the authors considered only

settings with low dimension.
In this work, we perform extensive simulation studies to compare the discrimination performance of lasso and the combination of stepwise with maximum likelihood estimation in logistic regression models. We also include in the analyses the combination of lasso and maximum likelihood method, in which the latter is used for parameter estimation and the former for variable selection. In the simulation studies, we vary the ratio of number of covariates to sample size from 0.01 to 0.5. The performance of these methods is also compared using nine real datasets.
The remainder of this paper is organized as follows. Section 2 presents the model and the methods considered in this work. The discrimination performance of the methods are compared using Monte Carlo simulation studies and nine real databases in Section 3 and 4, respectively. Concluding remarks are provided in Section 5.

2. Methodology

Let $y_1, y_2, \ldots, y_n$ be independent random variables, where each y_i, $i = 1, 2, \ldots, n$, is Bernoulli distributed with probability of success μ_i. The logistic regression model for binary responses is defined as

$$g(\mu_i) = \log\left(\frac{\mu_i}{1-\mu_i}\right) = \alpha + x_i^\top \beta, \tag{1}$$

where $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^\top$ is a vector of known covariates, α is an unknown intercept parameter and $\beta = (\beta_1, \beta_2, \ldots, \beta_p)^\top$ is a vector of unknown parameters ($\beta \in R^p$). Logistic regression is a particular case of generalized linear models in which the response variable is Bernoulli distributed and $g(.)$ is the logit link function (McCullagh and Nelder, 1989).
The parameters of the logistic regression model can be estimated by maximum likelihood, in which estimators of the parameters are obtained maximizing the log-likelihood of the model (1). This maximization uses a nonlinear optimization algorithm, such as the iterative weighted least squares (McCullagh and Nelder, 1989, Section 2.5). The log-likelihood of the model (1) is given by

$$l(\alpha, \beta) = \sum_{i=1}^{n} [y_i \log(\mu_i) + (1 - y_i) \log(1 - \mu_i)] \tag{2}$$

where $\mu_i = e^{\alpha + x_i^\top \beta}/(1 + e^{\alpha + x_i^\top \beta})$. In the first method considered here, stepwise is used for variable selection and maximum likelihood is used for parameter estimation. We denote it by StepML.
In the lasso method, the parameters are estimated by minimizing the following function:

$$-l(\alpha, \beta) + \lambda \sum_{j=1}^{p} |\beta_j|, \tag{3}$$

where λ is a tuning parameter. In this work, we chose the tuning parameter using ten-fold cross validation (Fischetti et al., 2017, page 614). The chosen λ was the one that maximized the average of the area under the ROC curve (AUC) in the validation datasets.
As lasso also selects covariates, it is reasonable to use it for variable selection and another method for parameter estimation. However, to the best of our knowledge, there is no work that compares the combination of lasso and a parameter estimation method with other techniques in logistic regression. Here, we consider the combination of lasso for variable selection and maximum likelihood for parameter estimation. We denote this combination as LassoML.

3. Simulation studies

To compare the discrimination performance of the methods described in Section 2 in the logistic regression, we performed a Monte Carlo simulation study. Following Van Calster et al. (2020), we considered a full factorial simulation setup varying the following factors: number of covariates ($p = 10, 30$ and 50), outcome rate event or percentage of successes (20% and 50%) and the correlation between predictors. The covariates were generated as random draws from the multivariate normal distribution with mean vector composed of zeros, variance vector composed of ones and correlations given by $(-\rho)^{|i-j|}$ (Hastie et al., 2020), where i and j are the ith and jth covariates, respectively, and $\rho = 0.5$ in the half of the scenarios and $\rho = 0.9$ in the remaining ones.

For each scenario, we considered that the 80% first covariates were noise ones (associated parameters equal to zero) and the value of the parameter for the $(0.8p + i)th$ covariate $(i = 1, 2, \ldots, p - 0.8p)$ was given by $(-1)^{(i+1)}(0.5)$. The following sample sizes were considered: $n = 100, 200, 500$ and 1000. Each sample was split in training dataset (70%) and test dataset (30%) and 500 Monte Carlo replications were considered in each scenario and sample size.

We used the Gini coefficient (GC) (Thomas et al., 2017) as the measure of the discrimination performance. This measure is a transformation of the area under the ROC curve given by GC $= 2(\text{AUC} - 0.5)$. The GC is usually used to evaluate credit scoring models (Silva et al., 2022) instead of AUC, because it takes values in the interval $(0, 1)$.

Simulation studies and applications were performed using R version 4.0.2. The analyses were conducted primarily using the `glm` function and the `glmnet` package (version 4.0-2). The scripts used in this work are available at `https://github.com/GilAlcantara/Study_Lasso_Article`.

Figure 1 presents the results of the simulations for the scenarios in which the outcome rate event (ORE) is equal to 0.5. When $p = 50$ and $n = 100$, the average GC is much higher for Lasso than for StepML (0.65 versus 0.30 when $\rho = 0.5$). The average GC for LassoML is between the other two methods but closer to Lasso (0.55 when $\rho = 0.5$). The difference in the discrimination performance of the methods is related especially to the value of the ratio p/n. In general, the greater this ratio, the greater the difference in the average GC between the three methods and Lasso is the method of best discrimination performance followed by LassoML. When p/n is less than 0.05, the three methods present similar performance. The difference in the average GC between the methods is slightly higher when ρ is changed from 0.5 to 0.9. As the differences are small when this change is made, the level of correlations between the covariates does not considerably affect the relative discrimination performance of the methods considered here.

Due to the high computational cost, we did not consider scenarios with $p/n > 0.5$ in the simulation studies. We investigated high-dimensional settings ($p/n > 2$) through the two real-data applications presented in Section 4.

The results of the simulations for the scenarios in which the ORE is equal to 0.2 are shown in Figure 2. When we change the ORE from 0.5 to 0.2, the conclusions are similar in most of the cases. However, for fixed p, n and ρ, in some case, the difference between the methods is considerably greater for ORE equals to 0.2 than when it is 0.5. For example, When $p = 50$, $n = 500$ and $\rho = 0.9$, the three methods have similar discrimination performance when the ORE is equal to 0.5. However, for the same p, n and ρ and ORE equals 0.2, the average GC is considerably greater for Lasso (0.91) than for StepML (0.75).

In addition to the average GC, Tables 3 and 4 in the Appendix also report the standard deviations (SD) of GC. Note that, in general, the standard deviations are small. Considering this information and the fact that, in this case, the standard errors are given by $\text{SD}/\sqrt{500}$, Lasso is statistically significantly better than the other methods in settings in which it presents considerably higher average GC.

4. Applications

We also compared the methods considered here using nine applications. The datasets were obtained from different areas and have different values of n, p/n and ORE. Table 1 presents the characteristics of these datasets. Note that p/n varies from 0.001 to 4.783.

Table 1: Features of the used datasets.

Dataset	n	p	p/n	ORE $(\%)$	Reference
1	30000	23	0.001	22	Yeh and Lien (2009)
2	3656	15	0.004	15	Detrano et al. (1989)
3	392	8	0.020	33	Ramana et al. (2011)
4	123	6	0.049	50	Thrun et al. (1991)
5	569	30	0.053	37	Street et al. (1993)
6	351	34	0.097	36	Sigillito et al. (1989)
7	195	22	0.113	75	Little et al. (2007)
8	70	205	2.929	41	Zarchi et al. (2018)
9	115	550	4.783	33	Sørlie et al. (2003)

For each application, we split the dataset in training dataset (70%) and test dataset (30%) and repeated this procedure 100 times. Table 2 presents the average and standard deviation of the GC for the three methods in the test datasets.

Although the simulation study did not include high-dimensional settings ($p/n > 0.5$), two of the nine real-data

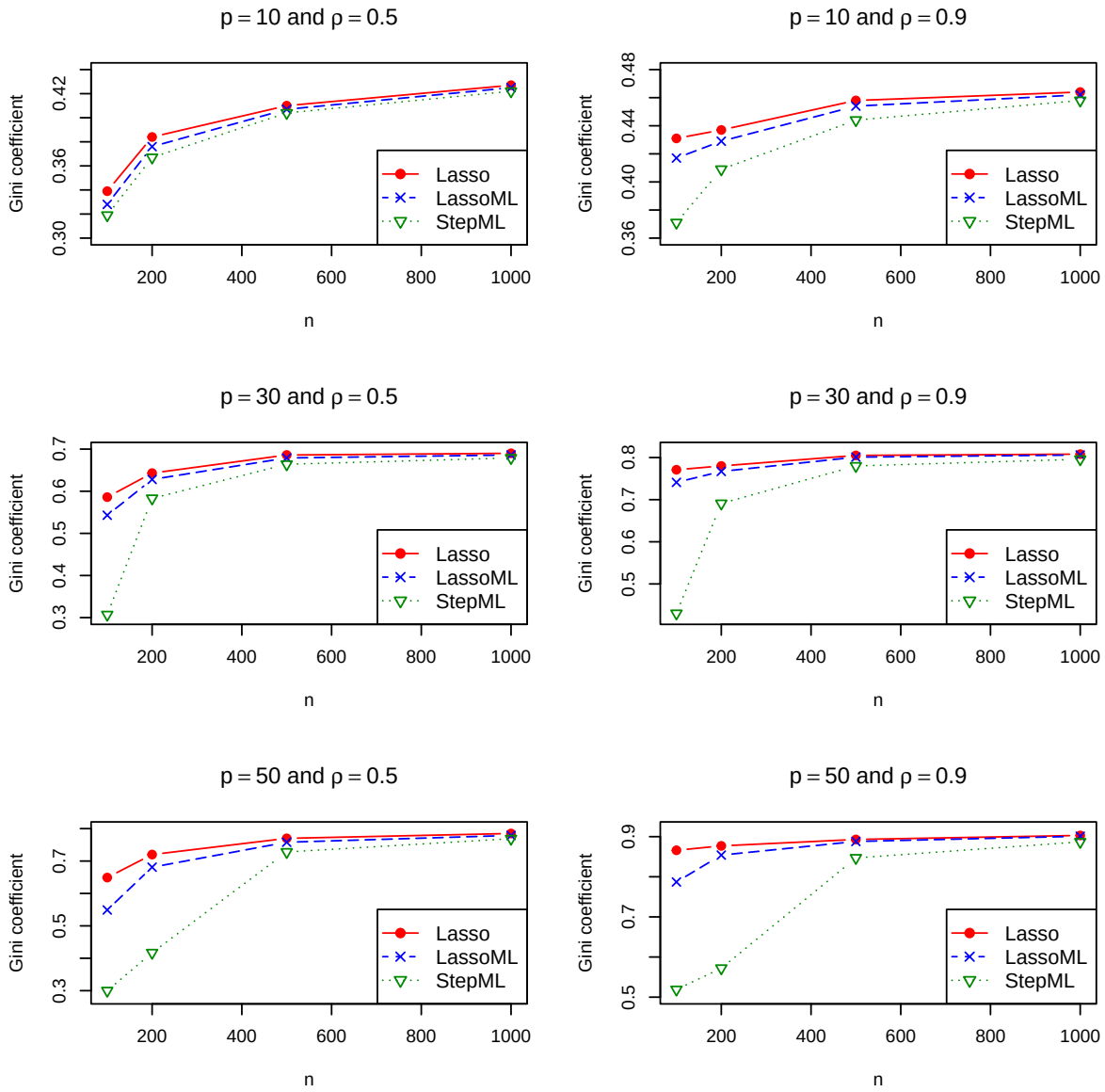

Figure 1: Average Gini coefficient for scenarios with outcome rate event equals to 50%

applications correspond to such scenarios. Nevertheless, considering all nine applications, the results were generally consistent with the patterns observed in the simulations. In particular, the differences in discrimination performance among the methods were mainly related to the value of the ratio p/n. For the the two datasets in which $p > n$, the average GC is much higher for Lasso than for StepML and LassoML has also an average GC much higher than StepML and lower than Lasso. On the other hand, in the four datasets in which p/n is lower than 0.05, the discrimination performance of the three methods is similar. The other three datasets have p/n between 0.05 and 0.12. In two of them, the average GC follows the same order across the methods noted in the datasets in which $p > n$. However, the differences in the average GC are lower for these datasets because p/n is not so high. The last dataset (number 7) has an unexpected results. It presents p/n greater than 0.1, but the average GC is similar for the three methods. For the datasets in which Lasso has a higher average GC than the other methods, it also presents a lower standard deviation of the GC.

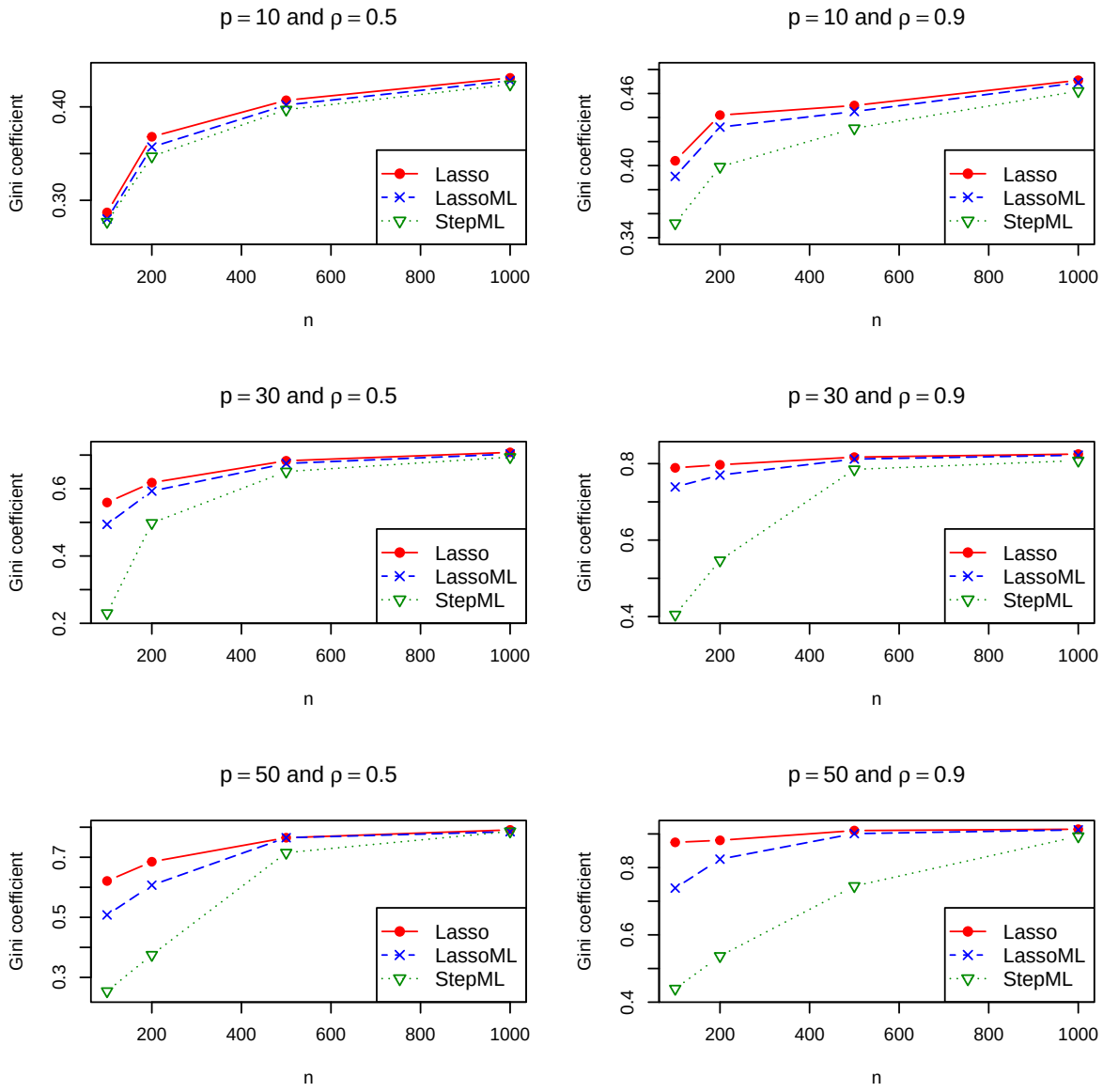

Figure 2: Average Gini coefficient for scenarios with outcome rate event equals to 20%

5. Concluding remarks

In this work, we compared the discrimination performance of three methods used to fit a logistic regression model. The considered methods are the following: lasso, the combination of stepwise for variable selection and maximum likelihood for parameter estimation and the combination of lasso for variable selection and maximum likelihood for parameter estimation. These methods were compared using Monte Carlo simulation studies and nine applications. The main conclusion of the work is that lasso has a better discrimination performance than the other two methods when the ratio of the number of covariates (p) to sample size (n) is high.

The analyses performed here suggest that the three methods have similar discrimination performance when p/n is lower than 0.05. For values higher than 0.05, in general, the greater p/n, the greater the difference in the discrimination performance between lasso and the other two methods. The relative performance of the methods seems to be less affected by the outcome rate event and by the level of correlation between the covariates. However, in general, the superiority of lasso compared to the other methods seems to be slightly higher when the outcome rate event is farther from 0.5 and when the correlation between the covariates is higher.

Considering all the analyses performed in this work, lasso did not present a lower discrimination performance than

Table 2: Average and standard deviation of the Gini coefficient for the nine applications.

Dataset	Method	Gini coefficient	
		Average	Standard deviation
1	Lasso	0.445	0.012
	LassoML	0.446	0.012
	StepML	0.445	0.012
2	Lasso	0.462	0.033
	LassoML	0.461	0.033
	StepML	0.460	0.033
3	Lasso	0.693	0.061
	LassoML	0.694	0.061
	StepML	0.690	0.061
4	Lasso	0.528	0.133
	LassoML	0.530	0.181
	StepML	0.529	0.132
5	Lasso	0.984	0.013
	LassoML	0.967	0.034
	StepML	0.892	0.038
6	Lasso	0.813	0.065
	LassoML	0.784	0.075
	StepML	0.710	0.095
7	Lasso	0.784	0.079
	LassoML	0.776	0.089
	StepML	0.781	0.102
8	Lasso	0.662	0.183
	LassoML	0.591	0.197
	StepML	0.243	0.239
9	Lasso	0.444	0.120
	LassoML	0.335	0.164
	StepML	0.115	0.145

the other methods in any application or scenario of the simulation studies. In addition, lasso is much better than the other methods considered here when p/n is high. Therefore, if the main goal of a work is obtaining a model with good discrimination performance, the logistic regression model should be fitted using lasso instead of maximum likelihood estimation.

References

Detrano, R., Janosi, A., Steinbrunn, W., Pfisterer, M., Schmid, J.-J., Sandhu, S., Guppy, K. H., Lee, S., and Froelicher, V. (1989). International application of a new probability algorithm for the diagnosis of coronary artery disease. *The American journal of cardiology*, 64(5):304–310.

Fischetti, T., Mayor, E., and Forte, R. M. (2017). *R: Predictive Analysis*. Packt Publishing Ltd.

Gaebel, W., Riesbeck, M., Wölwer, W., Klimke, A., Eickhoff, M., von Wilmsdorff, M., de Millas, W., Maier, W., Ruhrmann, S., Falkai, P., et al. (2016). Predictors for symptom re-exacerbation after targeted stepwise drug discontinuation in first-episode schizophrenia: results of the first-episode study within the german research network on schizophrenia. *Schizophrenia research*, 170(1):168–176.

Greenwood, C. J., Youssef, G. J., Letcher, P., Macdonald, J. A., Hagg, L. J., Sanson, A., Mcintosh, J., Hutchinson, D. M., Toumbourou, J. W., Fuller-Tyszkiewicz, M., et al. (2020). A comparison of penalised regression methods for informing the selection of predictive markers. *PloS one*, 15(11):e0242730.

Hastie, T., Tibshirani, R., and Tibshirani, R. (2020). Best subset, forward stepwise or lasso? analysis and recommendations based on xtensive comparisons. *Statistical Science*, 35(4):579–592.

Hastie, T., Tibshirani, R., and Wainwright, M. (2019). *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC.

Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X. (2013). *Applied logistic regression*. John Wiley & Sons, 3 edition.

Leeuwenberg, A. M., van Smeden, M., Langendijk, J. A., van der Schaaf, A., Mauer, M. E., Moons, K. G., Reitsma, J. B., and Schuit, E. (2022). Performance of binary prediction models in high-correlation low-dimensional settings: a comparison of methods. *Diagnostic and prognostic research*, 6(1):1.

Little, M., Mcsharry, P., Roberts, S., Costello, D., and Moroz, I. (2007). Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *Nature Precedings*, pages 1–1.

Liu, H., Wang, X., Tang, K., Peng, E., Xia, D., and Chen, Z. (2021). Machine learning-assisted decision-support models to better predict patients with calculous pyonephrosis. *Translational Andrology and Urology*, 10(2):710.

Martin, G. P., Riley, R. D., Collins, G. S., and Sperrin, M. (2021). Developing clinical prediction models when adhering to minimum sample size recommendations: The importance of quantifying bootstrap variability in tuning parameters and predictive performance. *Statistical Methods in Medical Research*, 30(12):2545–2561.

McCullagh, P. and Nelder, J. A. (1989). *Generalized linear models*. Chapman and Hall, London, second edition.

Müller, B. S., Uhlmann, L., Ihle, P., Stock, C., von Buedingen, F., Beyer, M., Gerlach, F. M., Perera, R., Valderas, J. M., Glasziou, P., et al. (2020). Development and internal validation of prognostic models to predict negative health outcomes in older patients with multimorbidity and polypharmacy in general practice. *BMJ open*, 10(10):e039747.

Pavlou, M., Ambler, G., Seaman, S., De Iorio, M., and Omar, R. Z. (2016). Review and evaluation of penalised regression methods for risk prediction in low-dimensional data with few events. *Statistics in medicine*, 35(7):1159–1177.

Ramana, B. V., Babu, M. S. P., Venkateswarlu, N., et al. (2011). A critical study of selected classification algorithms for liver disease diagnosis. *International Journal of Database Management Systems*, 3(2):101–114.

Riley, R. D., Snell, K. I., Martin, G. P., Whittle, R., Archer, L., Sperrin, M., and Collins, G. S. (2021). Penalization and shrinkage methods produced unreliable clinical prediction models especially when sample size was small. *Journal of clinical epidemiology*, 132:88–96.

Sigillito, V. G., Wing, S. P., Hutton, L. V., and Baker, K. B. (1989). Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10(3):262–266.

Silva, D. M., Pereira, G. H., and Magalhães, T. M. (2022). A class of categorization methods for credit scoring models. *European Journal of Operational Research*, 296(1):323–331.

Sørlie, T., Tibshirani, R., Parker, J., Hastie, T., Marron, J. S., Nobel, A., Deng, S., Johnsen, H., Pesich, R., Geisler, S., et al. (2003). Repeated observation of breast tumor subtypes in independent gene expression data sets. *Proceedings of the national academy of sciences*, 100(14):8418–8423.

Steyerberg, E. W., Eijkemans, M. J., Harrell Jr, F. E., and Habbema, J. D. F. (2000). Prognostic modelling with logistic regression analysis: a comparison of selection and estimation methods in small data sets. *Statistics in medicine*, 19(8):1059–1079.

Street, W. N., Wolberg, W. H., and Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis. In *Biomedical image processing and biomedical visualization*, volume 1905, pages 861–870. SPIE.

Thomas, L., Crook, J., and Edelman, D. (2017). *Credit scoring and its applications*. SIAM.

Thrun, S. B., Bala, J. W., Bloedorn, E., Bratko, I., Cestnik, B., Cheng, J., De Jong, K. A., Dzeroski, S., Fisher, D. H., Fahlman, S. E., et al. (1991). The monk's problems: A performance comparison of different learning algorithms. Technical report, Carnegie Mellon University.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288.

Van Calster, B., van Smeden, M., De Cock, B., and Steyerberg, E. W. (2020). Regression shrinkage methods for clinical prediction models do not guarantee improved performance: simulation study. *Statistical methods in medical research*, 29(11):3166–3178.

Yeh, I.-C. and Lien, C.-h. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert systems with applications*, 36(2):2473–2480.

Zarchi, M. S., Bushehri, S. F., and Dehghanizadeh, M. (2018). Scadi: A standard dataset for self-care problems classification of children with physical and motor disability. *International journal of medical informatics*, 114:81–87.

Appendix

Table 3: Average (standard deviation in parentheses) of the Gini coefficient for scenarios with outcome rate event equals to 50%

		$\rho = 0.5$			$\rho = 0.9$		
n	Method	$p = 10$	$p = 30$	$p = 50$	$p = 10$	$p = 30$	$p = 50$
100	Lasso	0.339(0.221)	0.586(0.182)	0.649(0.163)	0.431(0.206)	0.771(0.128)	0.866(0.091)
	LassoML	0.328(0.225)	0.543(0.190)	0.549(0.202)	0.417(0.212)	0.741(0.155)	0.787(0.147)
	StepML	0.319(0.215)	0.307(0.203)	0.300(0.196)	0.371(0.219)	0.430(0.194)	0.519(0.181)
200	Lasso	0.384(0.142)	0.643(0.112)	0.720(0.102)	0.437(0.139)	0.780(0.081)	0.877(0.062)
	LassoML	0.376(0.143)	0.628(0.114)	0.681(0.128)	0.429(0.139)	0.767(0.093)	0.854(0.094)
	Step+ML	0.367(0.144)	0.583(0.126)	0.417(0.138)	0.409(0.138)	0.691(0.114)	0.572(0.109)
500	Lasso	0.410(0.081)	0.686(0.063)	0.770(0.055)	0.458(0.080)	0.805(0.048)	0.893(0.032)
	LassoML	0.407(0.083)	0.679(0.065)	0.758(0.058)	0.454(0.080)	0.801(0.050)	0.888(0.035)
	StepML	0.404(0.082)	0.664(0.065)	0.728(0.062)	0.444(0.082)	0.780(0.055)	0.847(0.054)
1000	Lasso	0.427(0.058)	0.690(0.044)	0.785(0.035)	0.464(0.056)	0.808(0.034)	0.903(0.022)
	LassoML	0.425(0.058)	0.686(0.044)	0.779(0.037)	0.462(0.056)	0.806(0.035)	0.901(0.023)
	StepML	0.422(0.057)	0.679(0.045)	0.769(0.037)	0.458(0.057)	0.796(0.036)	0.887(0.026)

Table 4: Average (standard deviation in parentheses) of the Gini coefficient for scenarios with outcome rate event equals to 20%

		$\rho = 0.5$			$\rho = 0.9$		
n	Method	$p = 10$	$p = 30$	$p = 50$	$p = 10$	$p = 30$	$p = 50$
100	Lasso	0.287(0.264)	0.559(0.231)	0.621(0.187)	0.404(0.251)	0.789(0.145)	0.875(0.104)
	LassoML	0.280(0.261)	0.494(0.255)	0.508(0.254)	0.391(0255)	0.739(0.191)	0.739(0.218)
	StepML	0.277(0.274)	0.230(0.223)	0.254(0.285)	0.352(0.256)	0.405(0.230)	0.440(0.249)
200	Lasso	0.368(0.182)	0.618(0.147)	0.685(0.156)	0.442(0.175)	0.797(0.106)	0.881(0.077)
	LassoML	0.357(0.184)	0.593(0.170)	0.607(0.154)	0.432(0.180)	0.770(0.142)	0.825(0.149)
	StepML	0.347(0.184)	0.498(0.188)	0.375(0.146)	0.399(0.186)	0.547(0.207)	0.537(0.169)
500	Lasso	0.407(0.109)	0.683(0.081)	0.765(0.074)	0.450(0.100)	0.817(0.060)	0.910(0.037)
	LassoML	0.402(0.110)	0.675(0.084)	0.765(0.079)	0.445(0.102)	0.812(0.063)	0.901(0.054)
	StepML	0.397(0.109)	0.651(0.088)	0.715(0.073)	0.431(0.103)	0.785(0.069)	0.745(0.147)
1000	Lasso	0.431(0.074)	0.708(0.050)	0.791(0.039)	0.471(0.072)	0.825(0.041)	0.914(0.024)
	LassoML	0.428(0.074)	0.703(0.052)	0.785(0.038)	0.469(0.072)	0.822(0.042)	0.912(0.025)
	StepML	0.424(0.075)	0.694(0.053)	0.786(0.039)	0.462(0.073)	0.808(0.043)	0.893(0.028)