Seeking Predictive Invariance in Multi-Group Multiple Regression

Glæsel, Signe

Universiteit Leiden

Faculteit der Sociale Wetenschappen

Seeking Predictive Invariance in Multi-Group Multiple Regression

Master's Thesis

Signe Marie Glæsel

Master's Thesis Psychology,
Methodology and Statistics Unit, Institute of Psychology
Faculty of Social and Behavioral Sciences, Leiden University

Date: December 15th, 2022

Supervisor: Dr. Henk Kelderman, Dr. Marjoleijn Fokkema

Abstract

Objective: The purpose of this project was to find a way to tackle the problem of unfair prediction with respect to group membership. I wanted to achieve predictive invariant prediction and recognize variables where *predictive invariance* (PI) of linear regression across groups held. **Method:** *Simulation study:* I applied LASSO and MCP penalization methods to penalize intercepts, slopes, and their group differences towards zero. Simulation factors included *sample size*, *PI-status of intercept,* and *number of PI-slopes.* The goal was to correctly recognize the PI-status of intercepts and slopes. Outcomes such as *proportion of correctly identified PI-status of parameters* as well as *sensitivity* and *specificity* were inspected. *Empirical study*: Life Satisfaction was predicted from 11 separate predictors. Sex was used as a grouping variable. LASSO and MCP were used to penalize slopes as well as their difference between groups. **Results:** PI intercepts are more often recognized compared to Not PI intercepts. The most influential factors are 1) number of coefficients in the model with the same PI-status and 2) sample size. MCP is found to be more accurate than LASSO in recognizing PI slopes and intercepts. The opposite is true for the recognition of Not PI coefficients. **Conclusion:** The methods can be used improve the fairness of predictions with respect to groups.

Keywords: *Predictive Invariance, Multi-Group Multiple Regression, Penalizing group differences.*

Table of Contents

Seeking Predictive Invariance in Multi-Group Multiple Regression

A lot of research concerns group differences and numerous articles are written on the topic. Some studies have looked at differences between patients and controls to analyze if certain treatment methods are effective (see Lafeuille et al., 2015; Smits et al., 2020). Other research has studied people with different blood types and if they were prone to specific health outcomes (see Latz et al., 2020; Leonard et al., 2010). Some have investigated how being a part of different groups of society may have had an impact on well-being, so be it, gender, ethnic- or socio-economic background, religious groups, etc. (see Cortès-Franch et al., 2019; Hanif et al., 2021; Hetelekides et al., 2021; Wood et al., 2017). Thus, it is only fair to say that how group membership influences prediction can be a major topic.

In this study I investigated the prediction of *life satisfaction* from a set of independent variables[1], using *sex* as a grouping variable – a multigroup multiple regression analysis. Previous literature regarding the prediction of life satisfaction has found differences between male and female subjects for some of the predictive variables used in this analysis as well as a difference in life satisfaction scores overall. Females have higher average life satisfaction scores than men (Al-Attiyah & Nasser, 2016; Joshanloo & Jovanović, 2020). Thus, if we were to assume two multiple regressions, one would expect the intercept for the regression performed on female subjects to have a greater intercept. Moreover, in a recent study by Becchetti and Conzo (2022), they discussed the Gender Life satisfaction / Depression Paradox. They found that despite women having more life satisfaction overall, they were significantly more influenced by their (negative) emotional states than men. As a result, I assume that for variables related to emotional states (i.e., *euphoria*, *dysphoria,* and *general psychological health*) slopes are larger for female subjects, than for male

[1] For a full list as well as abbreviations and statistics, see Table A1.

subjects. Moreover, according to recent research by Joshanloo and Jovanović (2020) women scored higher on life satisfaction across employment groups, suggesting that status of employment was less relevant for women. Joshanloo (2018) had found the same results. In addition, they found evidence that variables related to education were of greater importance when predicting life satisfaction in men. Thus, looking back at our multiple regressions, men should have larger slopes on variables related to *unemployment* and potentially also *study delay*.

In addition to these variables where group differences are evident, there are some variables where the prediction of life satisfaction appears independent of group membership or where a difference has yet to be found. For example, research performed on the prediction of life satisfaction from measures of (positive- and negative-) *self-esteem,* and personality traits (i.e., *impulsivity, neuroticism*) suggest that life satisfaction could be predicted consistently across groups (Suldo et al., 2015; Ye et al., 2012). Moreover, Radošević et al. (2018) found that *need for change* influences life satisfaction. Here a group difference has not been properly investigated and consequently has also not been found. Looking at *disinhibition*, research has, just like with our *need for change* variable, found a link with life satisfaction (Bourbonnais & Durand, 2018). The same researchers found women to display higher levels of disinhibition, yet an investigation of how the disinhibition differs for men and women concerning life satisfaction was not present and thus again, one cannot assume differences to be present.

Using these variables as predictors of life satisfaction, where some clearly depend on group-membership, one could choose to make two separate models for each group. Alternatively, one could make one single prediction that is biased with respect to group membership. However, both methods have their downside. Having separate analyses make it troublesome to say something about the whole population, and a single prediction function could end up being unfair to certain subgroups when group membership is not considered (see Cleary, 1968). It would be of interest to investigate an alternative.

Optimally, one would make a multiple regression prediction that is 1) not two different regressions, and 2) is not generalizing. That is, one would want to be able to include all the variables mentioned, in one large prediction function without disregarding group membership. This is where the topic of predictive invariance (PI)[2] becomes relevant, where prediction does not depend on group membership yet also does not exclude it from the analysis.[3]

Lately, little research has been conducted on the topic of PI, especially considering multiple regression, as compared to latent variable modeling. Millsap (1997) has written about PI and compared it to measurement invariance – a topic revisited both by Borsboom (2006) as well as Millsap (2007) himself. These studies have focused on how PI relates to measurement invariance and how they are comparable but not the same. Partly the reason for this lack of recent research on PI could be found in the discussion by Hunter and Schmidt (2000), as mentioned by Millsap (2007). In short, they argue that one can trust the research done on this topic and it thus needs no further exploration. This has led to there being little enthusiasm regarding the investigation of PI, and if discussed, it has always been linked to measurement invariance. Until the present day, there has been a lack of research that looks at unbiased prediction through the means of PI. This is where this study becomes relevant. It was of interest to investigate how PI could be analyzed in multi-group multiple regression (MGMR). The goal of this research was to answer the following questions:

Considering MGMR: Under what conditions can one a) correctly recognize predictive invariance, and b) make prediction that is fair with respect to multiple groups?

---

[2] For a full list of abbreviations, see Table A2.
[3] The term PI is elaborated on in the Theoretical Framework section.

In the upcoming theoretical chapter, the theory behind this problem and how to investigate it is presented. More specifically the chapter concerns: How PI is linked to other more familiar concepts, how I intend to be referring to PI throughout the paper, how PI translates to multiple linear regression, and in what way one can investigate PI.

Theoretical Framework

In this chapter, the theoretical framework is presented. In detail, I discuss how PI links to other forms of invariance and how it can be applied to linear regression. Moreover, I discuss the use of penalties and how one can approach PI through penalized regression.

PI as a Concept

Let Y be a continuous criterion measure of interest, X - a vector of predictor variables, and G - a discrete group identifier, a general definition of PI is

$$P(Y|X,G) = P(Y|X), \tag{1}$$

for all relevant Y, X, and G (Millsap, 2007).

Consider the example from the introduction and let Y represent *life satisfaction*, X the predicting variables *euphoria*, *general mental health, neuroticism, etc.*, and G the dichotomous grouping variable *sex*. Using Millsap's way of phrasing, PI holds for the model when the probability of life satisfaction, given the scores on its predictors, is independent of sex. That is not to say that group differences in the distributions of the predictors do not exist.

When discussing PI of parameters Huang (2020) used a different terminology than Millsap. They use the terms *homogeneity* and *heterogeneity*. Conversely, if PI held for the slopes of euphoria, the parameters were homogenous. Likewise, if PI did *not* hold, they called parameters *heterogeneous*. In this study, I will be introducing a new term, namely PI-status.[4] PI-status has two

[4] It may seem redundant to introduce yet another term for the same concept, yet with respect to abbreviations and to make the text easier to read this is a necessary detail.

possible states; “PI” – where PI holds and thus parameters are homogeneous, and “Not PI” – where PI does *not* hold, and parameters are heterogeneous. The parameters that have “PI-status PI” are also referred to as “PI parameters”. Likewise, the parameters with PI-status Not PI are also referred to as Not PI parameters.

PI is seen as an analogy to other types of invariances. Specifically, measurement- and structural invariance in structural equation modeling (SEM). For example, in measurement invariance, Equation 1 is applied to the probability of vector of manifest indicators (Y) of latent factors (X) and a grouping variable (G). The difference lies in that measurement invariance ensures the meaningful comparison of factor scores or their distributions of latent factors across groups whereas PI ensures the fairness of *prediction* concerning groups.

PI in Regression

Considering PI in linear regression, Millsap (2007) mentioned that there are multiple types of PI. First, there is *slope invariance*. This is the simplest type of PI. If one faces slope invariance, the variables overall yield the same linear predictor in each group, except that the scores on the outcome variable may still be systematically higher and have higher variance in one group than the other. The second type of PI is what Millsap (2007) coined as *strong regression invariance*. When strong regression invariance holds, both the *slopes* and the *intercept* of the model are invariant. Strong regression invariance is comparable to *strong factorial invariance* in SEM (Meredith, 1993, as cited by Millsap, 2007). An even more restrictive type of PI is what will be referred to as *full predictive invariance*. In the case where PI holds for the intercept, all slopes as well as the *error variance*, one has achieved full predictive invariance. In practice, the interest has focused on the regression intercept and -slopes and thus strong regression invariance (Millsap, 2007).

Using for prediction a model where strong regression invariance does *not* hold will result in *biased* predictions due to group membership. Consequently, I seek a model:

$$Y_g = \beta_{0g} + x_{1g}\beta_{1g} + \cdots + x_{Pg}\beta_{Pg} + \epsilon_g \,, \tag{2}$$

where the intercept, β_0, and slopes, β_1 through β_P, *are equal* across groups, *g*.[5]

Knowing that the goal was to seek the model presented, figuring out how to get there would be the next step. This was where penalization came into play.

Penalizing Prediction Towards Invariance

Ordinarily, penalization has been used for variable selection and interpretation as well as to improve the accuracy of prediction (Hastie et al., 2015, p. 7). The regression coefficients, β, were penalized towards zero. Originally, one can look at this step at which variables are good predictors of the outcome and which are not. In this study, it was of interest to minimize group differences, opting for a fair general prediction where group membership has been considered. In doing so one could pinpoint the PI-status of a slope and thus pinpoint what variables depended on group-membership. This can be done through penalized regression. Penalized regression can put a penalty on the group difference, minimizing the difference between β_{p0} and β_{p1} (p = 1, …, P). Below it is explained in more detail how this difference in parameter value, also referred to as *the increment* component, can be penalized.

The Role of the Increment Component. PI-status, and thus the homogeneity/heterogeneity of groups, can be investigated by looking at an increment component.

[5] If the intercepts and the slopes are invariant over groups the means of the predictions of given values of the predictors are invariant over groups, if the residual variances also are equal over groups the conditional distribution of predictions is invariant too. The latter is considered *full predictive invariance*.

Huang (2018) suggested a reparametrized analysis with penalization. In their analysis, they focused on two components; a *reference* component, $\underline{\beta}$, - defined as the common component across groups, and an *increment* component $\underline{\beta_g}$, - defined as the difference in parameter value between the reference group and a non-reference group, group g. The regression coefficient for any one group is then defined by the sum of the reference- and increment component:

$$\beta_g = \underline{\beta} + \underline{\beta_g}. \tag{3}$$

By investigating the sparsity of the increment component, one can detect the PI-status of a parameter. If the penalized estimate of the increment parameter of a predictor is equal to zero, the predictor's PI-status is PI. If this increment is non-zero, the predictor's PI-status is Not PI.

Penalizing the increment is not per se a new method as Huang (2018) used this method applied to structural equation modeling (SEM). However, applying it to MGMR is new.

Types of Penalization. Two penalized regression methods were included in this study: least absolute shrinkage and selection operator (LASSO) estimation and minimax concave penalty (MCP) estimation. LASSO regression, introduced by Tibshirani (1996), minimizes the residual sum of squares to the absolute value of the coefficients that are less than the penalty parameter, λ (Tibshirani, 1996), truncating at zero, where a λ that gives the most parsimonious model is chosen. In this study, penalties were put on the increment as well as the slope parameter. To compare the results of LASSO, I added MCP as an alternative penalization method. MCP, introduced by Zhang (2010), is a shrinkage estimate that gives close to unbiased and accurate variable selection for high-dimensional linear regression, according to Li and Yang (2019). Literature has already found MCP to be a good alternative to LASSO regarding prediction accuracy and when it comes to creating sparse estimation (Li & Yang, 2019). I expected this to be true for increment components as well.

Applying Penalties. Huang (2018) discuss how penalized estimates of SEM parameters are found by maximizing the penalized likelihood criterion for parameters, and regularization parameter, λ. Since we focus on MGMR and not SEM, the parameters will be denoted as β. The penalized likelihood criterion is defined then by:

$$U(\beta, \lambda) = L(\beta) - R(\beta, \lambda). \tag{4}$$

Here, $L(\beta)$ is the likelihood function for the model, and $R(\beta, \lambda)$ is the function penalizing the parameter estimates.[6] $R(\beta, \lambda)$ is defined by:

$$R(\beta, \lambda) \ = \ \sum_{p\,=\,1}^{P} c_{\underline{\beta_p}} \rho(\left|\underline{\beta_p}\right| \lambda) \ + \sum_{p=1}^{P} \sum_{g=1}^{G} c_{\underline{\beta_{pg}}} \rho(\left|\underline{\beta_{pg}}\right| \lambda)\,, \tag{5}$$

where the first term on the right-hand side of the equation contains the reference parameters and the second refers to the increment parameters. The factor *c* may have a value of 0 or 1 indicating what parameters *should be* penalized. Table 1 gives the penalty function for LASSO and MCP. Note that the main difference between LASSO and MCP is that MCP has a convexity parameter, δ, and LASSO has not (Zhang, 2010).

[6] For more detail on the likelihood function, see Huang (2018), p. 503.

Table 1

Mathematical Expression of Penalty put on the Component per Penalization Method

Penalization method	Mathematical expression
LASSO	$\rho_{LASSO}(\lvert\beta\rvert, \lambda) \;=\; \lambda\lvert\beta\rvert$
MCP	$\rho_{MCP}(\lvert\beta\rvert, \lambda) \;=\begin{cases} \lambda\lvert\beta\rvert - \dfrac{\beta^2}{2\delta}, & if \quad \lvert\beta\rvert \leq \lambda\delta \\ \dfrac{1}{2}\lambda^2\delta, & if \quad \lvert\beta\rvert > \lambda\delta \end{cases}$

Note. ρ is the penalty function. β is the penalized model parameter (read: reference- or increment component). λ is a non-negative penalization parameter. δ controls the convexity of MCP and thus how fast the penalization rate goes to zero.

The goal of this study was to be able to say something about whether MCP or LASSO was a preferred method when it came to investigating PI, using penalization methods previously used in SEM. Additionally, it was interesting to know under what conditions the potential preference occurs. Just how this was investigated is explained in the next section.

Method

In this study, I performed both a simulation study was as well as an analysis of empirical data. In the simulation study, I simulated the PI-status for parameter slopes and the model intercept. I explored to what degree the PI-status was recognized and what factors may have played a role in this. Concerning the empirical data, I investigated the degree to which it was possible to push PI to hold for the parts of the model where there was reason to believe PI held. In either study, I performed penalized MGMR with LASSO and with MCP – penalizing the both reference- and increment components.

Penalization in practice

As already mentioned previously, two separate regressions with LASSO and MCP penalty were used to perform analysis. The model was defined, the grouping variable and reference group were set, and the PI restrictions were applied. The focus was on strong regression invariance. The regularization parameter(s) were chosen by the model based on the Bayesian Information Criterion and not manually selected.

Software.

To perform the analysis, I used the LSLX package.[7] For implementation of method in simulation study, see Appendix B1. For code showing how the method is applied to empirical data, see Appendix B2. LSLX' author Po-Hsien Huang (2020) claims that LSLX is possibly the most sophisticated package for penalized SEM in terms of usability, dependability, efficiency, and functionality. He illustrated the use of LSLX by multi-group factor analysis, but it is sufficiently general to be used for multiple regression with several groups. As a part of the model specification, the penalization functions of Equation 4, Equation 5, and Table 1 could be applied. Within this

[7] R version 4.1.1

package, there was a separate function where one can specify what coefficients to penalize, namely $penalize_coefficient(). In the simulation, I set the reference group to be group “0” and specified "*y <-1/1*", and "*y<->y/1*" to indicate that the increment of the model intercept and slopes were to be penalized. LSLX further has the benefit of having similar model specifications to the better-known LAVAAN package, making it rather user-friendly, see Appendix C for more details.

Simulation Study

Continuous data were simulated from a multivariate normal distribution. All datasets included 12 predicting variables, X_{1g} through X_{12g}, one dichotomous group variable G, and one estimated outcome Y_g. First, the model intercept, β_{00}, and slopes, β_{p0}, (p = 1, …, P) for the reference group were simulated. After that intercepts β_{01}, and slopes, β_{p1}, (p = 1, …, P) for the other group were simulated, depending on PI-status. If a parameter’s PI-status was PI, both would be identical to that of the reference group. If the PI-status was Not PI, they would be equal to the reference component plus an added increment. Individual error terms, ϵ_g, were randomly distributed. Both the model intercept and slopes overall had a mean of 0.1 and a standard deviation of 0.25. The error term, with a mean of 0 and a standard deviation of 0.6, was constant across groups and predictors. The outcome variable was computed through a multiple linear regression model of mentioned predictors. Finally, all non-group variables were standardized using the pooled mean and standard deviation.[8]

I ran a full factorial design with 32 conditions and 100 replications in each cell. The following three design factors were included:

[8] For the simulated data this is not different than standardizing by z-scores. However, as one wants to keep the simulation study and empirical study close, all data sets within this study will be standardized with the same method, i.e., pooled standardizing.

- **Sample size** (*N*, with four levels: 100, 500, 1000, 4000), where both groups had N/2 observations.
- **PI-status of intercept** ($PI_{Intercept}$, with two levels: 1 [PI], 0 [Not PI]), where 1 entailed homogeneous intercepts across groups. 0 entailed heterogeneous intercepts across groups.
- **The number of simulated PI slopes** ($\#PI_{Slopes}$, with four levels: 0, 4, 8, 12). The number indicated how many of the possible 12 slopes were equal across groups. For example, when $\#PI_{Slopes}$ was set to 4 it implied that 4/12 slopes were identical and 8/12 were not. When $\#PI_{Slopes}$ was set to 0, all simulated slopes had PI-status Not PI, and when set to 12, all simulated slopes had PI-status as PI, and thus the condition of slope invariance held for the simulated population.

On each of the 32 cells the two penalization methods, LASSO and MCP, were applied to penalize both the reference- and increment components.

By including these factors, it was possible to investigate what factors influenced the ability to recognize the PI-status of intercepts and slopes and in turn investigate some of the different levels of predictive invariance. For the eight cells where $\#PI_{Slope}$ was 12, slope invariance was investigated. In the four cases where both $\#PI_{Slope}$ was 12 and the PI-status of the intercept was set to PI, strong regression invariance was investigated.

Using different methods, LASSO and MCP, allowed for an investigation into a preference of method when it comes to recognizing PI in MGMR. Based on previous findings, I expected either method overall to do better with larger sample sizes and to find better results with MCP when it came to recognizing PI slopes. How the PI-status of the intercept and how $\#PI_{Slopes}$ would influence the accuracy was more explorative.

To investigate how well the methods managed to recognize the PI-status, several different evaluation criteria were analyzed. The evaluation criteria were divided both by parameter type (i.e., slope and intercept) and by PI-status – PI or Not PI. A detailed description of the criteria can be found in the next sub-chapter.

Evaluation Criteria

In this simulation study, the aim was to assess the accuracy of PI-status recognition, both for intercepts and slopes. The evaluation criteria were divided into two main parts, each with three subcategories:

- **PI-status in Slopes.** How well the methods recognized the correct PI-status of slopes, across simulation factors and between methods.
  - **The number correctly recognized Slopes** (#T_{Slopes})[9]: An indicator for how many of the 12 simulated slopes our method truly recognized the PI-status. This was a discrete outcome with values ranging between 0 and 12 where 12 indicated a perfect result. Whether the PI-status was simulated to be PI or Not PI was not considered.
  - **The number of correctly recognized PI slopes** (#TPI_{Slopes}): The number of slopes for which the method correctly recognized PI slopes. This was a discrete outcome, similar to #T_{Slopes} but in this case, the value depended on #PI_{Slope}.
  - **The number of correctly recognized Not PI slopes** (#$TNPI_{Slopes}$): The number of slopes where the method correctly recognized Not PI slopes. This too ranged

[9] T in the abbreviations stand for True and is an indicator of the PI status of the recognized coefficient being the same as the simulated PI status. Likewise, F stands for False – an indicator of the simulated and predicted PI-status being different.

between 0 and 12 but depended on the number of slopes that had simulated PI-status *Not PI* (#NPI_{Slopes})[10].

- **PI-status in the Intercept.** How well the PI-status of the intercept was recognized across simulation factors and between methods.
    - **Correctly recognized Intercept** ($T_{Intercept}$): [Factor: 1 = recognized, 0 = not recognized]: A dichotomous variable indicating whether our method correctly recognized the PI-status of the intercept, disregarding the actual status as PI or Not PI.
    - **Correctly recognized PI Intercept** ($TPI_{Intercept}$): [Factor: 1 = correctly recognized, 0 = incorrectly recognized]: A dichotomous variable indicating whether the method correctly recognized the PI-status as PI.
    - **Correctly recognized Not PI Intercept** ($TNPI_{Intercept}$): [Factor: 1 = correctly recognized, 0 = incorrectly recognized]: A dichotomous variable indicating whether the method correctly recognized the PI-status as Not PI.

Additional Statistics.

In addition to the Evaluation Criteria *proportion of correctly identified slopes*, *sensitivity*, and *specificity* as well as the *number of correctly identified intercepts* were added to the analysis to give an even clearer picture of the accuracy of identification when investigating PI in slopes and intercepts.

Correctly Recognizing the PI-status of Slopes. Investigation of the correctly identified slopes was divided into three parts, all of which were investigated across simulation factors.

---

[10] In our analysis *number of variables* is equal to 12, thus #NPI_{Slopes} is equal to 12 - #PI_{Slopes}. It would also be correct to say it depends on #PI_{Slopes}, but in that case with a negative rather than positive relationship.

- **Proportion of correctly recognized slopes overall** [Range: 0:1]: The proportion of slopes where the method correctly recognized the PI-status of the slope,
- **Sensitivity of slopes** [Range: 0:1]: The proportion of slopes where the method correctly recognized PI-status as PI, and;
- **Specificity of slopes** [Range: 0:1]: The proportion of parameters where the method correctly recognized PI-status as Not PI.

All three parts related to each other, where the proportion of correctly recognized slopes overall was a more general measure than sensitivity and specificity. More specifically, the proportion of correctly recognized was calculated as:

$$\frac{\#\mathrm{T}_{\mathrm{Slopes}}}{Total\ \#\ of\ predictors}$$

…where the total number of predictor variables here is equal to 12.

Considering sensitivity and specificity, Figure 1 displays how they are calculated, their components, and how they relate. The denominator of sensitivity, the sum of number of True PI slopes and False PI slopes, is the number of slopes given PI-status PI in the simulation (i.e., $\#\mathrm{PI}_{\mathrm{Slopes}}$). Thus, if PI-status was set to PI for eight of the slopes and the model recognized six of them, the sensitivity rate equaled 6/8 = 0.75. In the same fashion, the denominator of specificity, the sum of number of True Not PI slopes and False Not PI slopes, was equal to $\#\mathrm{NPI}_{\mathrm{Slopes}}$. Thus, if PI-status again was set to PI for eight of the slopes, and hence Not PI for four slopes and the model recognized two of these, the specificity rate would equal 2/4 = 0.50.

Figure 1

Sensitivity and Specificity – Components, Relation, and Calculation

		PI-status of slopes recognized by the Model		
		PI	Not PI	
PI-status of slopes in Simulated Population	PI	Number of True PI slopes (#TPI)	Number of False Not PI slopes (#FNPI)	Sensitivity $\frac{\#TPI}{\#TPI\ +\ \#FNPI}$
	Not PI	Number of False PI slopes (#FPI)	Number of True Not-PI slopes (#TNPI)	Specificity $\frac{\#TNPI}{\#TNPI + \#FPI}$

Note. Displaying how #TPI, #FNPI, #FPI, and #TNPI are all linked and how they are used to calculate Sensitivity and Specificity separately. PI = Prediction Invariant, #TPI = Number of slopes with True PI-status as PI, #TNPI = Number of slopes with True PI-status as Not PI, #FPI = Number of slopes with False PI-status as PI, #FNPI = Number of slopes with False PI-status as Not- PI. #TPI + #FNPI = $\#NPI_{Slopes}$. #TNPI + #FPI = $\#PI_{Slopes}$. *True* entails that the simulated and predicted PI-statuses are the same. *False* entails that they are not.

Correctly Recognizing the PI-status of the Intercept. In addition to the evaluation criteria, the *total number of recognized intercepts* was added to the analysis to give a visual of the results. Here one could not look at proportions as there was only one intercept per model. Instead, I investigated the number of $T_{Intercept}$ per cell. In other words, investigating how many of the intercepts, in the 100 replications performed, were correctly recognized.

Having defined the evaluation criteria, or outcomes, the additional statistics, and the focus of the study, I now continue to the part where I go about investigating the topic and seeing what the evaluation criteria and statistics may show.

Analysis of Evaluation Criteria.

To investigate under what conditions the PI-status of slopes and intercepts were correctly recognized, I performed a multitude of different analyses. First off, to assess how the accuracy of the recognition of slopes, I performed a Quasi-Poisson Regression[11]. Secondly, when investigating the recognition of the intercept, I applied a Logistic Regression. See Table 2 for details on all analyses. In all analyses, I used simulation factors as independent variables setting a reference category per simulation factor. Each analysis was done separately for the outcomes extracted from the MCP- and LASSO analysis. To state how influential the levels of our simulation factors were, I inspected model effects, ratios[12], and 95% confidence intervals (CI). To tell what method did better, LASSO or MCP, I performed an analysis of deviance, using the Akaike Information Criterion (AIC) as the selector. The lower the value the better.

There are two things one should pay attention to when later looking at the results of these analyses. First, in the analyses, all coefficients and ratios were seen as a contrast to the specified reference categories. For example, if the Rate Ratio, or Odds Ratio, of a sample size of 500 (versus 100) was 1.50, it entailed that the method in question did 50% better when the sample size was 500, compared to when it was 100. Second, not all data was used for every analysis. It should go without saying that analysis on PI parameters excluded the data where all slopes had PI-status Not PI, and analysis on Not PI parameters excluded the data where all slopes had PI-status set to PI in the simulation. This explains why the reference category for $\#PI_{Slopes}$ in the analysis on PI slopes was 4 – not 0, and why the category where $\#PI_{Slopes}$ was equal to 12 was not

[11] I performed a Quasi-Poisson Regression as I expected to find data where the mean was larger than the variance and where the ratio undoubtedly did depend on the number of slopes set to be PI. Hence, I was dealing with potentially over-dispersed data and Quasi Poisson Regression was appropriate.
[12] For Quasi Poisson Regression these ratios are called Rate Ratios, for Logistic regression they are called Odds Ratios.

a part of the analysis on Not PI slopes. Moreover, the PI-status of the intercept was included in the analysis on $T_{Intercept}$, but not in the $TPI_{Intercept}$ nor the $TNPI_{Intercept}$ analysis.

In this part of the study, I had hoped to be able to identify what factors influence the correct recognition of PI in intercept and slopes as well as to test what methods could potentially perform this task better.

Table 2

Overview Analysis performed on Evaluation Criteria gained from Simulation Study

Evaluation Criteria	Reference category per simulation factor [a]	Type of Analysis	Offset
Number of True Slopes ($\#T_{Slopes}$)	$PI_{Intercept}$: PI $\#PI_{Slope}$: 0 Sample Size: 100	Quasi-Poisson Regression	-
Number of slopes with True PI-status: PI ($\#TPI_{Slopes}$)	$PI_{Intercept}$: PI $\#PI_{Slope}$: 4 Sample Size: 100	Quasi-Poisson Regression	$\#PI_{Slope}$
Number of slopes with True PI-status: Not PI ($\#TNPI_{Slopes}$)	$PI_{Intercept}$: PI $\#PI_{Slope}$: 0 Sample Size: 100	Quasi-Poisson Regression	$\#NPI_{Slope}$
True Intercept ($T_{Intercept}$)	$PI_{Intercept}$: PI $\#PI_{Slope}$: 0 Sample Size: 100	Logistic Regression	-
True PI-status: PI for the Intercept ($TPI_{Intercept}$):	$\#PI_{Slope}$: 0 Sample Size: 100	Logistic Regression	-
True PI-status: Not PI for the Intercept ($TNPI_{Intercept}$):	$\#PI_{Slope}$: 0 Sample Size: 100	Logistic Regression	-

Note. Overview of type analysis done on different evaluation criteria, or outcomes, and what reference categories were used for each predicting factor. PI = Predictive Invariant. All analyzes are done separately for MCP and LASSO.

- = offset not applicable.

[a] = Simulation factors used as predictors and the value used as contrast.

Empirical Study

Data

The original data used for this project was an empirical data set consisting of 1775 responses on several hundred questionnaire items where each item was a part of a test where the test score was the sum score across items. The data was part of a longitudinal study conducted between 1987 and 1991. Only the data from 1987 was used in this analysis as it was the most complete. A total of 11 tests were included and used as continuous predictor variables. The data consisted of 888 female- and 887 male subjects. Missing values were dealt with via LSLX's default two-step method for missing data.[13] In this study, the aim was to predict life satisfaction from different test scores related to personality, self-esteem measures, and work status.[14] Additionally, sex was used as a grouping variable. This data set was chosen because it had a multitude of variables. This came in handy when studying penalization. Moreover, it was a large sample where the groups were of equal size which makes the study of PI more powerful.

Analysis

The data were analyzed similarly to one cell of the simulated data, using LASSO and MCP separately. Full predictive invariance was enforced, and *females* were used as the reference group. Increment components were investigated. Increments equal to zero indicated PI. Non-zero increments indicated Not PI.

In this study, I had hoped to see that a) the methods recognized the PI-status of the intercept and slope coefficient in line with theory and b) that the methods penalized the model towards PI so that the prediction was fair with regards to group membership.

---

[13] According to Huang (2020), this is a more efficient method than Listwise Deletion.
[14] For a full list, see Table A1.

It is important to note that unlike in the simulation study, it was of course impossible to know what slopes had what PI-status in the population. However, previous literature indicated what one could expect. In line with the theory presented in the *introduction*, the expected PI-status of the slope was PI for the following six variables: *positive*- and *negative self-esteem*, *impulsivity*, *neuroticism*, *need for change*, and *disinhibition*. For the five remaining variables included in the analysis, namely *euphoria*, *dysphoria*, *general psychological health*, *unemployment,* and *study delay,* the theory suggested that PI-status was Not PI. For *euphoria*, *dysphoria,* and *general psychological health* the parameters' group differences were expected to be negative. For *unemployment,* and *study delay* the parameters' group differences should according to theory be positive. Finally, seeing that females were more satisfied with life than men, I expected PI-status as Not PI for the model intercept, and for the increment to be negative.

Having explained what both the simulation- and empirical study were based on as well as how I went about analyzing the different data, I now move on to the result section where I present the results of my analysis. Any discussion of said results is included in the *Discussion* section.

Results

In this section, I investigated the ability of LASSO and MCP to recognize the PI-status of intercept- and slope coefficients. A look at simulation factors and how they affected said ability is presented in the following section.

Simulations Study

PI in Slope Parameters

Recognition of PI-status in Slopes. The results supported our expectations with regards to sample size improving the number of recognized slopes and MCP being better at recognizing slopes where PI-status was set to PI rather than Not PI, see Table 3. The PI-status of the intercept had no effect. The $\#PI_{Slopes}$ had significant effects. Results indicated that a combination of slopes with PI and Not PI slopes decreased the number of correctly recognized slopes. This effect was larger for LASSO than for MCP. LASSO showed worse results when slope invariance was simulated than when PI-status was simulated to be Not PI for all slopes. For MCP, there were opposite results, where the best-case scenario would be where slope invariance was present in the data. Looking at a combination of factors, the best results were found where sample size was high and there were no PI slopes, see Figure 2. Both the intercept of the LASSO model, and of the MCP model were significant, suggesting that at the baseline level, both methods were very well able to recognize the PI-status of the slopes correctly. Overall, 85.3% of slopes were recognized correctly.[15] LASSO correctly recognized 84.1% and MCP 86.5%. When comparing LASSO and MCP by an analysis of deviance I found residual deviance by AIC to be 618.0 for MCP and 676.5 for LASSO – a difference of 58.5 in MCP's favor.

[15] The percentages of correctly recognized values are calculated excluding the missing values.

Table 3

Quasi-Poisson Regression: Correct Recognition of the PI-status of Slopes Overall by Regression Estimates and Rate Ratios

Variable	LASSO						MCP					
	Estimate	*SE*	*p*	*RR*	95% CI		*Estimate*	*SE*	*p*	*RR*	95% CI	
					LL	*UL*					*LL*	*UL*
(Intercept)	2.255	.007	.000***	**9.54**	**9.40**	**9.67**	2.134	.007	.000***	**8.48**	**8.36**	**8.60**
PI-status Intercept [a]												
PI	0.000	.005	.991	1.00	0.99	1.01	-0.004	.005	.387	1.00	0.99	1.01
# PI_{Slopes} [b]												
4	-0.144	.007	.000***	**0.87**	**0.85**	**0.88**	-0.103	.007	.000***	**0.90**	**0.89**	**0.91**
8	-0.139	.007	.000***	**0.87**	**0.86**	**0.88**	-0.021	.007	.002**	**0.98**	**0.97**	**0.99**
12	-0.087	.007	.000***	**0.92**	**0.90**	**0.93**	0.080	.006	.000***	**1.08**	**1.07**	**1.10**
Sample Size [c]												
500	0.165	.007	.000***	**1.18**	**1.16**	**1.20**	0.228	.007	.000***	**1.26**	**1.24**	**1.27**
1000	0.187	.007	.000***	**1.21**	**1.19**	**1.22**	0.271	.007	.000***	**1.31**	**1.29**	**1.33**
4000	0.221	.007	.000***	**1.25**	**1.23**	**1.27**	0.319	.007	.000***	**1.38**	**1.36**	**1.39**

Note. Displaying the effect of simulation factors on correctly recognizing slopes, regardless of PI-status, across methods. When $p < 0.05$, and the 95% CI of the Rate Ratio is *larger* than 1, the method does significantly better under the category in question compared to the reference category. When $p < 0.05$, and the 95% CI of the Rate Ratio is *less* than 1, the method does significantly worse under the category in question compared to the reference category. N = 3178. Rate Ratios indicating significant effects are in bold.

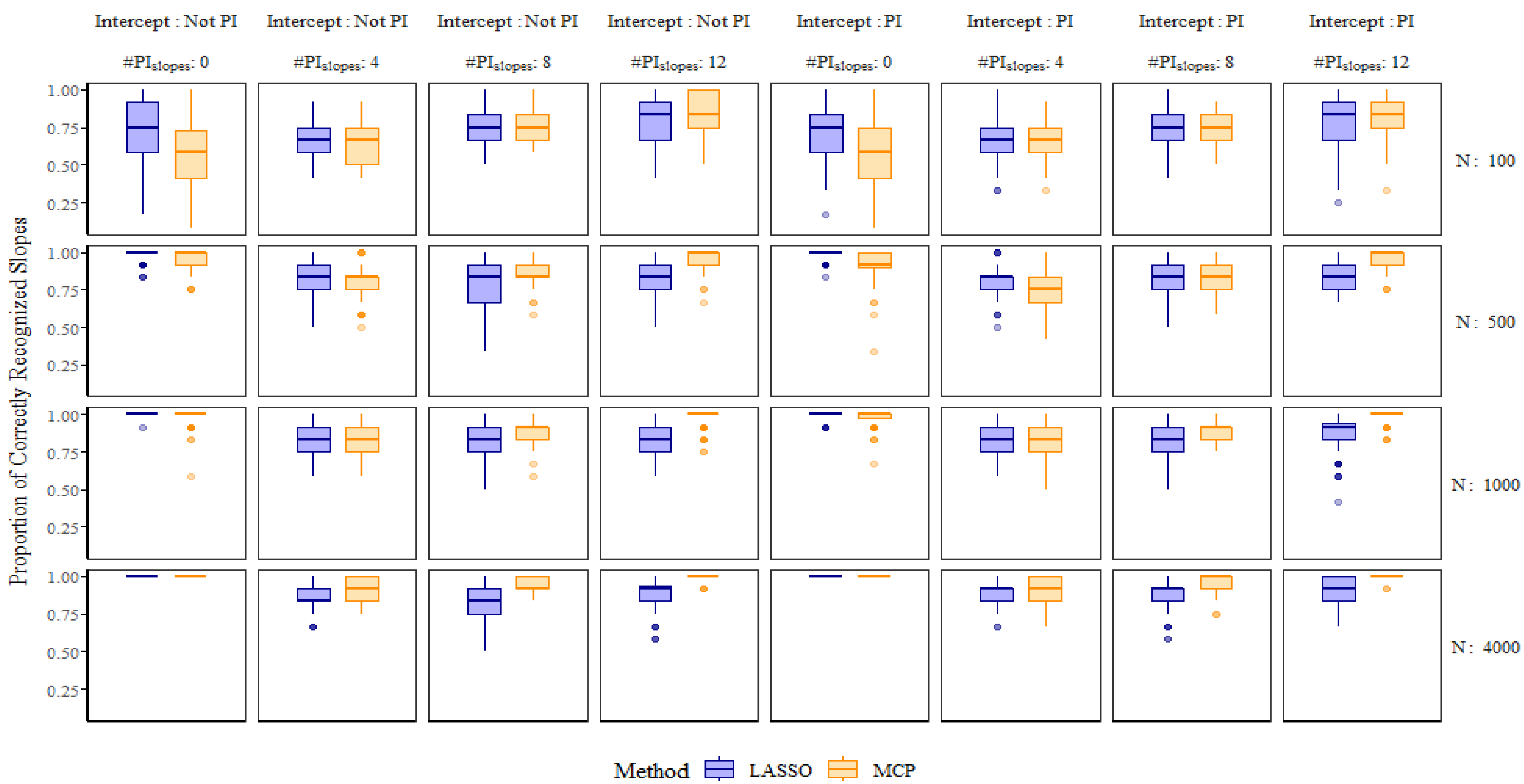
SE = Standard Error, RR = Rate Ratio, CI = confidence interval: LL = lower limit: UL = upper limit.

[a] 0 = intercept not-PI, [b] 0 = 0 PI slopes, [c] 0 = sample size 100.

* = p < 0.05, ** = p < 0.01, *** = p < 0.001.

Figure 2

Proportion of Correctly Recognized Slopes



Note. Displaying the proportion of slopes for which the PI-status was correctly recognized across simulation factors. PI = Predictive Invariant. N = 6356. Missing = 44.

Recognition of PI-status: PI in Slopes. In this analysis, I found that the larger $\#PI_{Slope}$, the greater the chance of recognizing them as such was. PI-status of the intercept was found to be significant by the LASSO regression when looking at the beta coefficients, but not when looking at the RRs. For MCP the PI-status of the intercept was deemed irrelevant. For detailed statistics see Table 4.

The intercepts for both the LASSO and the MCP model were found significant – indicating that PI slopes overall were correctly recognized. This is further supported when looking at the percent of correctly recognized slopes in this analysis. Overall, the two methods correctly recognized 87.5% of the slopes for which PI was set to hold. MCP correctly predicted 93.1% of the slopes. That is 11.4% more than LASSO, which predicted 81.7% of the PI slopes correctly. The analyses of deviance showed that the AIC value of residual deviance for LASSO was 675.25, versus 233.53 for MCP. This was a 441.72 difference indicating that MCP was favored.

Sensitivity. To clarify the results and be able to speak of a combination of simulation factors, I looked at sensitivity, see Figure 3. In all scenarios presented, both methods did recognize slopes with PI-status correctly the majority of the time. Regarding method preference, MCP had higher scores of the proportion correctly recognized. This was true in all possible combinations of factors. In total 12.5% of all slopes were not recognized correctly. When $\#PI_{Slopes}$ was low, the spread of the sensitivity rate was enlarged and when combined with using LASSO as a method there were cases where none of the PI slopes were recognized, despite being simulated as such. Correctly recognizing the PI-status of slopes as PI was more likely when slope invariance held for the data. A combination of high $\#PI_{Slope}$ and large sample size shows the best results both for LASSO and MCP.

Table 4

Quasi-Poisson Regression: Correct Recognition of PI slopes by Regression Estimates and Rate Ratios

Variable	LASSO						MCP					
	Estimate	*SE*	*p*	*RR*	95% CI		*Estimate*	*SE*	*p*	*RR*	95% CI	
					LL	*UL*					*LL*	*UL*
(Intercept)	-0.332	.013	.000***	**0.72**	**0.70**	**0.74**	-0.212	.007	.000***	**0.81**	**0.80**	**0.82**
PI-status Intercept [a]												
PI	0.019	.008	.022*	1.02	1.00	1.04	0.004	.005	.404	1.00	0.99	1.01
# PI_{Slopes} [b]												
8	0.063	.012	.000***	**1.07**	**1.04**	**1.09**	0.031	.007	.000***	**1.03**	**1.02**	**1.05**
12	0.112	.012	.000***	**1.12**	**1.09**	**1.14**	0.050	.006	.000***	**1.05**	**1.04**	**1.06**
Sample Size [c]												
500	0.035	.012	.003**	**1.04**	**1.01**	**1.06**	0.106	.007	.000***	**1.11**	**1.10**	**1.13**
1000	0.050	.012	.000***	**1.05**	**1.03**	**1.08**	0.138	.007	.000***	**1.15**	**1.13**	**1.16**
4000	0.081	.012	.000***	**1.08**	**1.06**	**1.11**	0.166	.007	.000***	**1.18**	**1.17**	**1.20**

Note. Displaying the effect of simulation factors on correctly recognizing PI slopes, across methods. When $p < 0.05$, and the 95% CI of the Rate Ratio is *larger* than 1, the method does significantly better under the category in question compared to the reference category. When $p < 0.05$, and the 95% CI of the Rate Ratio is *less* than 1, the method does significantly worse under the category in question compared to the reference category. N = 2380. Rate Ratios indicating significant effects are in bold.

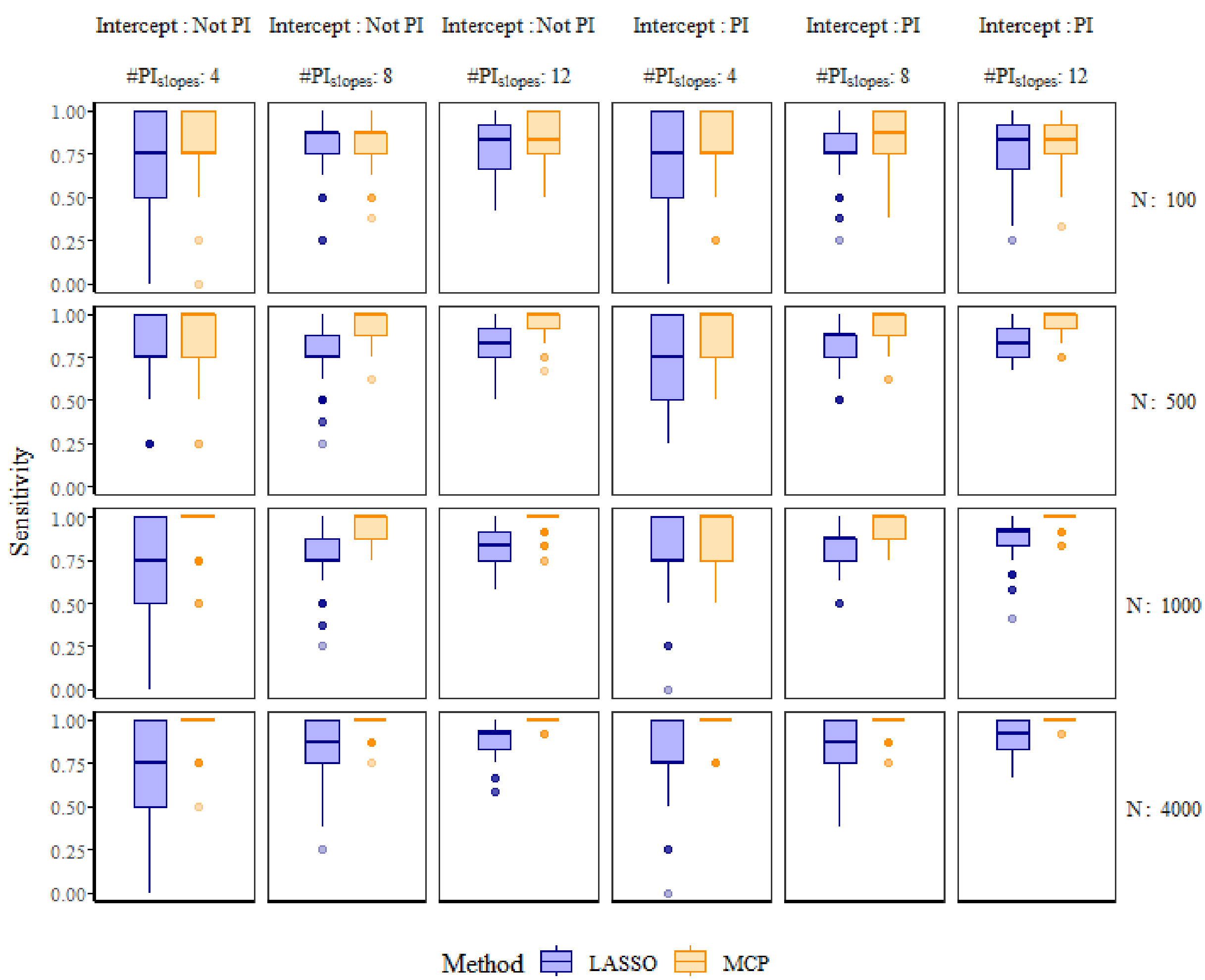
SE = Standard Error, RR = Rate Ratio, CI = confidence interval: LL = lower limit: UL = upper limit.

[a] 0 = intercept not-PI, [b] 0 = 4 PI slopes, [c] 0 = sample size 100.

* = p < 0.05, ** = p < 0.01, *** = p < 0.001.

Figure 3

Sensitivity: The Proportion of PI slopes Correctly Recognized



Note. Displaying Sensitivity of Slopes – across simulation factor levels and methods applied.

PI = Predictive Invariant. Higher scores of sensitivity indicate higher accuracy in correctly predicting the simulated PI-status of slopes as PI. $\#PI_{Slopes}$ equal to 0 is excluded as there is naturally no data on sensitivity to be found.

Number of simulated datasets = 4760.

Recognition PI-status: Not PI in Slopes. Results indicated that across penalization methods, sample size was the most influential factor where a sample size of 4000 increased correct recognition. This effect was found larger for MCP than for LASSO, see Table 5. Increasing $\#PI_{Slopes}$ had a negative effect on the recognition. Another way to phrase this would be to say: the more simulated slopes with PI-status as Not PI, the easier the recognition of such slopes were. This effect was again found to be stronger for MCP. $PI_{Intercept}$ was deemed significant when inspecting the regression coefficients for LASSO, yet irrelevant once RR was considered. For MCP it was consistently irrelevant. Overall, the methods recognized 83.1% of these slopes correctly. That is 86.6% for LASSO and 79.7% for MCP. Moreover, the analysis of deviance showed residual deviance of 527.65 for LASSO and 772.64 for MCP – a 244.98 difference in the favor of LASSO.

Specificity. In the same fashion that I previously investigated sensitivity when looking at $\#TPI_{Slopes}$, I looked at specificity when dealing with $\#TNPI_{Slopes}$, see Figure 4. Specificity had some features in common with sensitivity. However, regarding what method performed best and how $\#PI_{Slopes}$ influenced the results, I found an opposite pattern. This was in line with the Quasi-Poisson regression analysis regarding $\#TNPI_{Slopes}$. In addition, LASSO outperformed MCP with higher rates of sensitivity in all cases, expect the ones where both were 100% accurate. Both methods did worse when $\#PI_{Slopes}$ was high[16]. Moreover, increasing sample size increased specificity. The PI-status of the intercept seemed to have no impact. The best-case scenario entailed a large sample size and low $\#PI_{Slopes}$.

To summarize, correctly recognizing the PI-status of slopes depended on sample size and the number of slopes that had the same PI-status as the slope investigated. Regarding method preference, LASSO had higher rates of correct recognition of Not PI slopes. MCP had higher rates of correct recognition of Not PI slopes.

[16] Changing the angle and looking at it from a Not-PI point of view one can deduct that an increasing number of Not-PI slopes is related to a bigger proportion of Not-PI slopes being identified.

Table 5

Quasi-Poisson Regression: Correct Recognition of Not PI slopes by Regression Estimates and Rate Ratios

Variable	LASSO						MCP					
	Estimate	*SE*	*p*	*RR*	95% CI		*Estimate*	*SE*	*p*	*RR*	95% CI	
					LL	*UL*					*LL*	*UL*
(Intercept)	-0.328	.009	.000***	**0.72**	**0.71**	**0.73**	-0.481	.012	.000***	**0.62**	**0.60**	**0.63**
PI-status Intercept [a]												
PI	-0.016	.007	.017*	0.98	0.97	1.00	-0.012	.009	.162	0.99	0.97	1.00
$\#PI_{Slopes}$ [b]												
4	-0.118	.008	.000***	**0.89**	**0.87**	**0.90**	-0.179	.010	.000***	**0.84**	**0.82**	**0.85**
8	-0.146	.010	.000***	**0.86**	**0.85**	**0.88**	-0.211	.013	.000***	**0.81**	**0.79**	**0.83**
Sample Size [c]												
500	0.295	.010	.000***	**1.34**	**1.32**	**1.37**	0.382	.013	.000***	**1.47**	**1.43**	**1.50**
1000	0.322	.010	.000***	**1.38**	**1.35**	**1.41**	0.436	.013	.000***	**1.55**	**1.51**	**1.59**
4000	0.359	.010	.000***	**1.43**	**1.40**	**1.46**	0.508	.013	.000***	**1.66**	**1.62**	**1.70**

Note. Displaying the effect of simulation factors on correctly recognizing the PI-status of slopes as Not PI, across methods. When $p < 0.05$, and the 95% CI of the Rate Ratio is *larger* than 1, the method does significantly better under the category in question compared to the reference category. When $p < 0.05$, and the 95% CI of the Rate Ratio is *less* than 1, the method does significantly worse under the category in question compared to the reference category. N = 2392. Rate Ratios indicating significant effects are in bold.

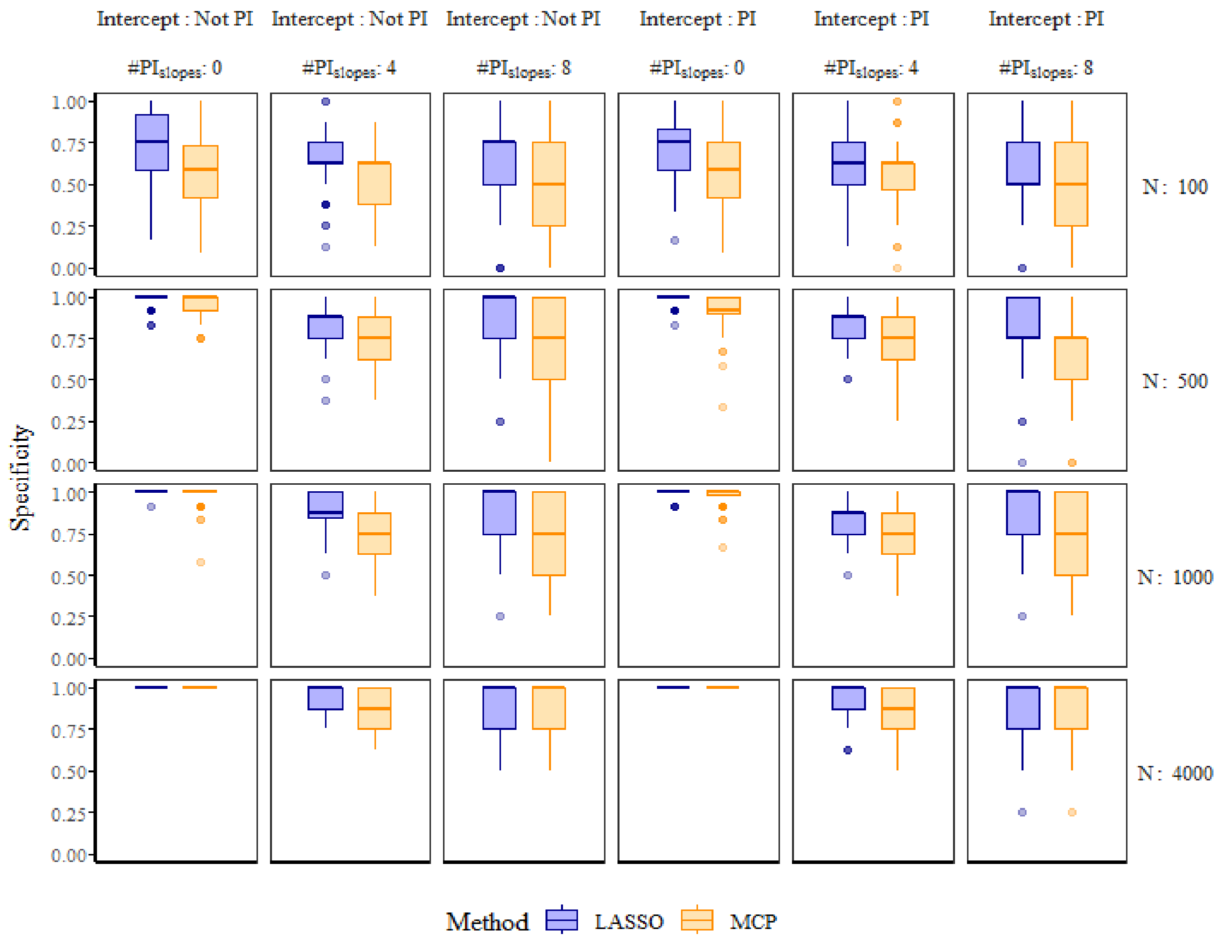
SE = Standard Error, RR = Rate Ratio, CI = confidence interval: LL = lower limit: UL = upper limit.

[a] 0 = intercept not-PI, [b] 0 = 4 NPI slopes, [c] 0 = sample size 100.

* = p < 0.05, ** = p < 0.01, *** = p < 0.001.

Figure 4

Specificity: The Proportion of Not PI slopes Correctly Recognized



Note. Displaying Specificity of Slopes – across simulation factor levels and methods applied. PI = Predictive Invariant. Higher scores of specificity indicate higher accuracy in correctly predicting the simulated PI-status of slopes as Not PI. $\#PI_{Slopes}$ equal to 12 is excluded as there is naturally no data on specificity to be found. Number of simulated datasets = 4784.

Having investigated how well PI-status could be recognized in slope parameters and how simulation factors influence this across different penalization methods, how well PI was recognized in the intercept naturally becomes the next topic.

PI in the Intercept Parameter

Logistic regression was performed on the recognition of the intercept status as a function of the simulation factors. This analysis was done both *overall* and as separate analyses for intercepts with PI-status PI and Not PI.

Recognition of PI-status in Intercept. Regarding the intercepts, correctly recognition of PI-status heavily depended on the sample size, improving results as sample size increased. This was in line with expectations. See Table 6 for details. Either method recognized intercepts with PI-status PI significantly better than they did Not PI intercepts. Thus, data where predictive invariance held for the intercept was more likely to be recognized as such than data where the PI-status of the intercept was set to Not PI. These effects were both larger for MCP. The number of correctly recognized intercepts was linked to a combination of PI and Not PI slopes in the model. What separated LASSO and MCP was the extent to which $\#PI_{Slopes}$ influenced the results. For LASSO only $\#PI_{Slopes}$ equal to eight was found to significantly improve the results. For MCP both $\#PI_{Slopes}$ equal four and eight were found to be significant. Additionally, it was found that at the baseline level MCP did not correctly recognize intercepts well, showing a significant negative model intercept. No effect of this sort was found for LASSO.

LASSO recognized 81.2% of the intercepts correctly, and MCP recognized 82.2%. Across methods that entailed an 81.7% rate of correctly recognized intercepts. An analysis of deviance showed an AIC value of 2684.0 for LASSO and 2202.4 for MCP, suggesting MCP was better by a 481.6 difference.

Table 6

Logistic Regression: Correct Recognition of Intercepts by Regression Estimates and Odds Ratios – Independent of PI-status

Variable	LASSO						MCP					
	Estimate	*SE*	*p*	*OR*	95% CI		*Estimate*	*SE*	*p*	*OR*	95% CI	
					LL	*UL*					*LL*	*UL*
(Intercept)	-0.154	.119	.196	0.86	0.68	1.08	-0.667	.132	.000***	**0.51**	**0.40**	**0.66**
Intercept Status [a]												
PI	0.745	.100	.000***	**2.11**	**1.73**	**2.56**	1.985	.125	.000***	**7.28**	**5.72**	**9.33**
#PI_{Slopes} [b]												
4	0.241	.135	.074	1.27	0.98	1.66	0.319	.152	.036*	**1.38**	**1.02**	**1.86**
8	0.431	.139	.002**	**1.54**	**1.17**	**2.02**	0.330	.153	.031*	**1.39**	**1.03**	**1.88**
12	0.083	.133	.536	1.09	0.84	1.41	-0.054	.148	.715	0.95	0.71	1.27
Sample Size [c]												
500	1.279	.122	.000***	**3.59**	**2.83**	**4.58**	1.434	.131	.000***	**4.20**	**3.25**	**5.44**
1000	1.577	.131	.000***	**4.84**	**3.76**	**6.28**	1.892	.143	.000***	**6.64**	**5.04**	**8.82**
4000	2.351	.164	.000***	**10.50**	**7.68**	**14.62**	3.786	.262	.000***	**44.07**	**27.15**	**76.37**

Note. Displaying the effect of simulation factors on correctly recognizing intercepts, regardless of PI-status, across methods. When $p < 0.05$, and the 95% CI of the Odds Ratio is *larger* than 1, the method does significantly better under the category in question compared to the reference category. When $p < 0.05$, and the 95% CI of the Odds Ratio is *less* than 1, the method does significantly worse under the category in question compared to the reference category. N = 3178. Odds Ratios indicating significant effects are in bold.

SE = Standard Error, OR = Odds Ratio, CI = confidence interval: LL = lower limit: UL = upper limit.

[a] 0 = intercept not-PI, [b] 0 = 0 PI slopes, [c] 0 = sample size 100.

* = p < 0.05, ** = p < 0.01, *** = p < 0.001.

Recognition of the PI-status: PI in Intercept. When investigating the correctly recognized intercepts with PI-status PI, $TPI_{Intercept,}$ results indicated that for LASSO $\#PI_{Slopes}$ was the most influential factor. See Table 7 for detailed results. Here recognition increased as $\#PI_{Slopes}$ increased. A similar size effect was found for $\#PI_{Slopes}$ using MCP. However, for MCP sample size remained the main factor of influence. Either method showed better results with a larger sample size. A finding that I hadn't seen in the overall analysis was that either method was able to recognize PI intercepts at the baseline level indicated by their significant positive model intercept. This finding, combined with that of how $\#PI_{Slopes}$ has a positive effect on the results, suggests that the closer to full regression invariance the data is, the more likely it is to find PI to hold for the model.

Overall, 89.6 % of the intercepts with PI-status PI were correctly recognized. Looking at it separately for each method, LASSO and MCP recognized 86.1% and 93.1% respectively. Figure 5 displays the number of intercepts, with PI-status PI, that were recognized per 100 replications across simulation factors. In the best-case scenario, the sample size was large, and more of the other coefficients (read: slopes) were PI in the data. Additionally, it appeared that MCP outperformed LASSO which proved to be even more evident after having performed an analysis of deviance. Results showed AIC values of 1200.90 and 707.94 for LASSO and MCP, a 492.96 difference in MCP’s favor. This suggested a preference for MCP when looking to recognize intercepts where PI holds.

Table 7

Logistic Regression: Correct Recognition of Intercepts with PI-status PI by Regression Estimates and Odds Ratios

Variable	LASSO						MCP					
	Estimate	*SE*	*p*	*OR*	95% CI		*Estimate*	*SE*	*p*	*OR*	95% CI	
					LL	*UL*					*LL*	*UL*
(Intercept)	0.688	.162	.000**	**1.99**	**1.45**	**2.75**	0.977	.188	.000***	**2.66**	**1.85**	**3.87**
$\#PI_{Slopes}$ [a]												
4	0.840	.187	.000***	**2.32**	**1.61**	**3.36**	1.330	.285	.000***	**3.78**	**2.21**	**6.78**
8	1.361	.215	.000***	**3.90**	**2.59**	**6.03**	1.314	.285	.000***	**3.72**	**2.17**	**6.67**
12	1.443	.222	.000***	**4.23**	**2.77**	**6.64**	1.407	.297	.000***	**4.08**	**2.34**	**7.52**
Sample Size [b]												
500	0.499	.206	.015*	**1.65**	**1.10**	**2.48**	1.028	.270	.000***	**2.79**	**1.67**	**4.83**
1000	0.405	.202	.045*	**1.50**	**1.01**	**2.24**	0.724	.248	.004**	**2.06**	**1.28**	**3.39**
4000	0.574	.209	.006**	**1.77**	**1.18**	**2.69**	2.389	.440	.000***	**10.90**	**4.97**	**28.78**

Note. Displaying the effect of the simulation factors on correctly recognizing intercepts, regardless of PI-status, across methods. When $p < 0.05$ and the 95%, CI of the Odds Ratio is *larger* than 1, the method does significantly better under the category in question compared to the reference category. When $p < 0.05$ and the 95%, CI of the Odds Ratio is *less* than 1, the method does significantly worse under the category in question compared to the reference category. N = 1588. Odds Ratios indicating significant effects are in bold.

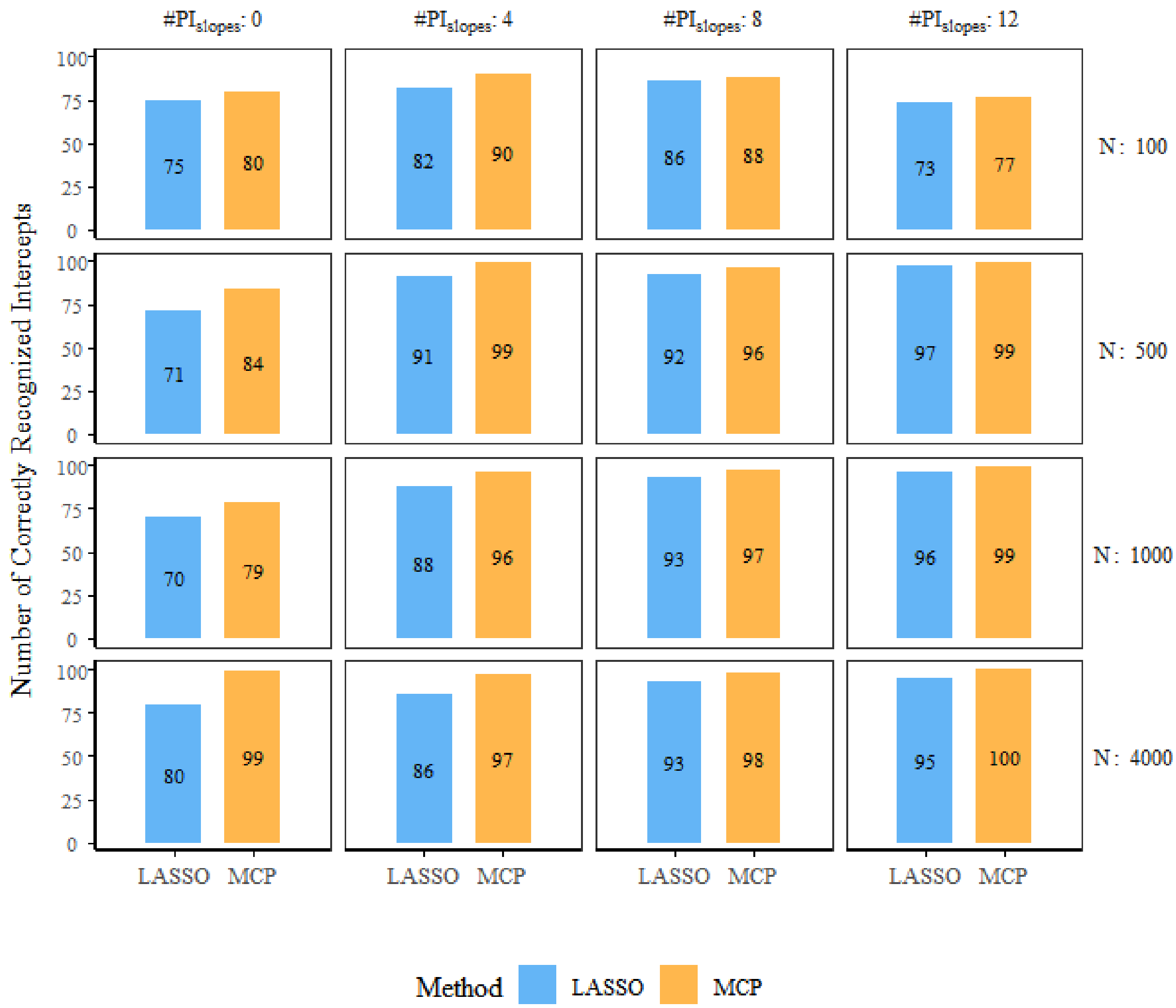
SE = Standard Error, OR = Odds Ratio, CI = confidence interval: LL = lower limit: UL = upper limit.

[a] 0 = 0 PI slopes, [b] 0 = sample size 100.

* = p < 0.05, ** = p < 0.01, *** = p < 0.001.

Figure 5

Number of Correctly Recognized Intercept with PI-status PI per 100 Replications



Note. Displaying the number of correctly recognized intercepts for models with PI as PI-status of the intercept, across simulation factors. PI = Predictive Invariant. Total number of identified intercepts: 2846. Total possible: 3200. Missing data: = 12. $\text{Total}_{\text{LASSO}}$ =1386. $\text{Total}_{\text{MCP}}$ = 1478.

Recognition of PI-status: Not PI in Intercept. Regarding the correct recognition of intercepts with PI-status as Not PI, $\text{TNPI}_{\text{Intercept}}$, I found results like those from the analysis on $\#\text{TNPI}_{\text{Slopes}}$, see Table 8 for details. For LASSO, the greater $\#\text{PI}_{\text{Slopes}}$ was, the lesser the odds were of recognizing an intercept with PI-status Not PI. This effect was also present for MCP but to a lesser degree where the results were significantly worse only in the cases where slope invariance was implemented, i.e. where $\#PI_{Slopes}$ was equal to 12. Looking at Figure 6, it becomes apparent that this could be explained through that MCP overall correctly recognized fewer intercepts with PI-status Not PI than did LASSO. Having inspected the negative model intercept for MCP it was apparent that even at the baseline level MCP did not correctly recognize intercepts with PI-status Not PI. Sample size was the more influential factor for both methods, where a sample size of 4000 compared to a sample size of 100 made it substantially more likely to correctly recognize an intercept with PI-status Not PI. This effect was larger for LASSO than for MCP. Both methods did worse with a small sample size, where in the worst-case scenario neither method correctly recognized so much as a fourth of the intercepts. The best-case scenario for correctly recognizing Not PI intercept was a low $\#PI_{Slopes}$ and large sample size. In that scenario, the type of method did not matter as either did well. However, in general, there was a preference for LASSO when looking at Not PI intercepts.

Overall, our methods managed to recognize 73.8% of intercepts with PI-status Not PI. Separated, that was 76.2% for LASSO and 71.4% for MCP. The analysis of deviance showed an AIC value of 1236.4 for LASSO and 1409.5 for MCP. This is a deviance of 172.1 in the favor of LASSO.

Table 8

Logistic Regression: Correct Recognition of Intercepts with PI-status Not PI by Regression Estimates and Odds Ratios

Variable	LASSO						MCP					
	Estimate	*SE*	*p*	*OR*	95% CI		*Estimate*	*SE*	*p*	*OR*	95% CI	
					LL	*UL*					*LL*	*UL*
(Intercept)	0.058	.166	.725	1.06	0.77	1.47	-0.459	.158	.004**	**0.63**	**0.46**	**0.86**
$\#PI_{Slopes}$ [a]												
4	-0.533	.210	.011*	**0.59**	**0.39**	**0.88**	-0.224	.191	.240	0.80	0.55	1.16
8	-0.545	.210	.010*	**0.58**	**0.38**	**0.87**	-0.200	.191	.294	0.82	0.56	1.19
12	-1.240	.208	.000***	**0.29**	**0.19**	**0.43**	-0.774	.188	.000***	**0.46**	**0.32**	**0.66**
Sample Size [b]												
500	1.933	.167	.000***	**6.91**	**5.00**	**9.64**	1.696	.157	.000***	**5.45**	**4.02**	**7.44**
1000	2.564	.190	.000***	**12.98**	**9.01**	**19.02**	2.408	.175	.000***	**11.12**	**7.93**	**15.77**
4000	4.987	.465	.000***	**146.51**	**65.24**	**419.28**	4.366	.326	.000***	**78.73**	**43.45**	**157.88**

Note. Displaying the effect of simulation factors on correctly recognizing intercepts, regardless of PI-status, across methods. When $p < 0.05$, and the 95% CI of the Odds Ratio is *larger* than 1, the method does significantly better under the category in question compared to the reference category. When $p < 0.05$, and the 95% CI of the Odds Ratio is *less* than 1, the method does significantly worse under the category in question compared to the reference category. N = 1590. Odds Ratios indicating significant effects are in bold.

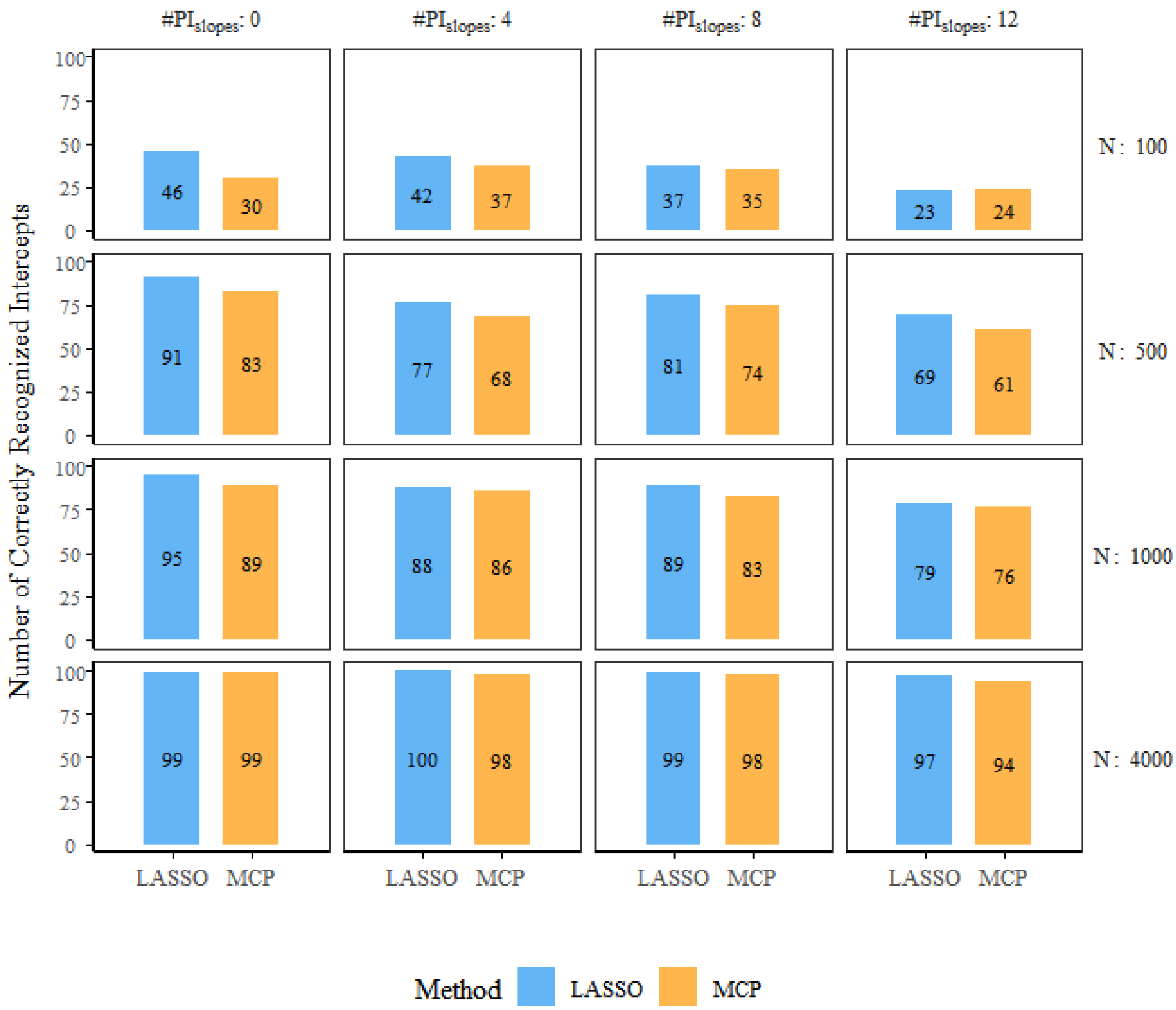
SE = Standard Error, OR = Odds Ratio, CI = confidence interval: LL = lower limit: UL = upper limit.

[a] 0 = 0 PI slopes, [b] 0 = sample size 100.

* = p < 0.05, ** = p < 0.01, *** = p < 0.001.

Figure 6

Number of Correctly Recognized Intercept with PI-status Not PI per 100 Replications



Note. Displaying the number of correctly recognized intercepts for models with Not PI as PI-status of the intercept, across simulation factors. PI = Predictive Invariant. Total number of identified intercepts: 2347. Total possible: 3200. Missing data: = 10. $\text{Total}_{\text{LASSO}}$ =1212. $\text{Total}_{\text{MCP}}$ = 1135.

To sum up, it was apparent that the accuracy of recognizing the PI-status of the intercept was positively related to the same factors as slopes with the same PI status, i.e., sample size and the number of slopes that had the same PI-status as the intercept in question. Additionally, the PI-status of the intercept influenced whether or not it would be correctly recognized. When it came to what method was preferred, they were the same as for slopes. For coefficients with PI-status PI, MCP was preferred and for coefficients with PI-status Not PI, the preference was for LASSO.

Non-Estimable Models

Considering non-estimable models, I found a pattern directly linked to the convergence of models when using the LSLX package. The said pattern can be seen in Table 9. In short, if the sample size was 100, there were instances where the methods did not converge (read: did not produce results)[17]. There were no non-estimable models found for other sample sizes. The number of non-estimable models was equal across methods, suggesting that LSLX overall handles small sample sizes worse than larger ones. Moreover, as $\#PI_{Slopes}$ increased the number of missing analyses increased. In total there were missing analyses on 44 out of a possible 6400 replications. That is, there were 0.69% missing analyses, all due to non-convergence. Knowing that there was one model intercept per analysis, one was able to deduce that there also were 44 missing intercepts. Regarding missing slopes, the number of missing analyses for the PI slopes was equal to the number in Table 9 multiplied by $\#PI_{Slopes}$. This equaled a total of 448 missing slopes out of the possible total of 19200 PI slopes (2.33 %). For the Not PI slopes, the number of non-estimable models was equal to the number of non-estimable models in Table 9 multiplied by $\#NPI_{Slopes}$ (i.e., 12 - $\#PI_{Slopes}$). That was 104 missing slopes, equaling 0.54% of the Not PI slopes.

[17] According to Huang (2020) this could be due to the δ-parameter being too small. However, that would not explain why the convergence issue is equally present for LASSO and MCP, as LASSO does not consider the δ-parameter.

Table 9

Non-estimable models Grouped by Simulation Factors

Sample Size	PI-status intercept: Not PI				PI-status intercept: PI			
	$\#PI_{Slopes}$: 0	$\#PI_{Slopes}$: 4	$\#PI_{Slopes}$: 8	$\#PI_{Slopes}$: 12	$\#PI_{Slopes}$: 0	$\#PI_{Slopes}$: 4	$\#PI_{Slopes}$: 8	$\#PI_{Slopes}$: 12
100	2	1	2	5	-	-	3	9
500	-	-	-	-	-	-	-	-
1000	-	-	-	-	-	-	-	-
4000	-	-	-	-	-	-	-	-

Note. Displaying the number of replications for which there are non-estimable models, across all simulation factors. The pattern and number of non-estimable models were identical for LASSO and MCP. Total number of missing analyses = 44. The possible maximum number of missing analyses is 100 per cell (i.e., a total of 6400).
PI = Predictive Invariant, $\#PI_{Slopes}$ = Number of slopes set to be PI in the simulation, - = no missing values.

Empirical Study

Concerning the empirical data, I sought to find out to what degree the PI-status of the intercept- and slope coefficients could be correctly identified consistent with the literature and to what extent the methods could push for strong regression invariance when predicting life satisfaction when using sex as a grouping variable. The results of the analysis are presented in Table 10. Coefficients marked in bold had their PI-status identified in line with the presented theory.

Inspecting the results using LASSO as the penalization method, the PI-status of the model intercept was identified as Not PI, with a negative increment. This translated to males overall scoring lower on life satisfaction than females – which was in line with the theory. Investigating PI-status for the slopes where PI was to be expected, the slopes for all variables but one were found to be PI. Positive self-esteem had an increment higher than zero, indicating that for this variable the PI-status of the slope was Not PI and that this variable was of greater influence for the male group when predicting life satisfaction. This was not in line with the presented literature. In total, five out of six variables that were assumed to have PI-status PI were found to be so when using LASSO as the penalization method.

Looking at the variables where the slopes were assumed to be Not PI, we saw that the method had only identified the expected PI-status of the slope of unemployment as Not PI. The increment was positive, suggesting that this variable was more important for males when compared to females when predicting life satisfaction. The remaining variables all had the PI-status of their slopes identified as PI. Based on these results one can state that our LASSO analysis on the empirical data, had a high sensitivity rate and a low specificity rate. Regarding the level of invariance that LASSO managed to enforce, intercept invariance was not enforced and consequently not strong regression invariance. The model came close to slope invariance, penalizing nine out of eleven slopes to exact zero.

Analyzing the results of using MCP as the penalization method, similar results were found, yet with what appears to be more conservative penalties. The increment of the intercept was again found to be negative and identified as Not PI, which was in line with the theory. Looking at the slopes, results indicated that for all slopes expected to be PI, the PI-status was identified as such – perfect sensitivity. For all slopes where PI-status was expected to be Not PI, the method had recognized none of them correctly – specificity equaling zero. At first glance this may appear as a negative thing, however, when considering the second half of this study's focus, namely pushing for predictive invariance, the results are promising. MCP had penalized all group differences to zero. This meant that for this model slope invariance did hold.

Comparing the two methods, they both correctly recognize the expected PI-status of the model intercept, as well as six out of eleven slopes. MCP pushed for PI to a larger degree than LASSO and had in addition a better overall fit ($RMSEA_{LASSO}$ = 0.051; $RMSEA_{MCP}$ = 0.038).

It is important to note that some of the variables themselves were penalized to have *zero* slopes, and hence could be removed from the model in a model selection scenario. The parameters' group differences for these variables are naturally zero too. This may give a skewed view of the sensitivity and specificity rates.

Keeping these results in mind, I will go on to discuss their implications and give advice regarding how I best believe future research can improve the research on the topic PI in MGMR and the quest for fair prediction.

Table 10

Accuracy of Recognizing PI-Status on Empirical Data: Results of Regression Analysis Predicting Life satisfaction - Using LASSO and MCP

Variable	Expected PI-status [a]	*LASSO*				*MCP*			
		β_p	*p*	β_{p1}	*p*	β_p	*p*	β_{p1}	*p*
(intercept)	Not-PI	.070	.015*	**-.141**	.000***	.113	.000***	**-.229**	.000***
Positive Self-Esteem	PI	.192	.000***	.022	.590	.224	.000***	**.000**	-
Negative Self-Esteem	PI	-.052	.076	**.000**	-	-.032	.280	**.000**	-
Disinhibition	PI	-.020	.388	**.000**	-	-.019	.409	**.000**	-
Need for Change	PI	-.044	.071	**.000**	-	-.047	.057	**.000**	-
Impulsivity	PI	.058	.006**	**.000**	-	.073	.001**	**.000**	-
Neuroticism	PI	.000	-	**.000**	-	.000	-	**.000**	-
Euphoria	Not PI	.198	.000***	.000	-	.215	.000***	.000	-
Dysphoria	Not PI	-.158	.000***	.000	-	-.178	.000***	.000	-
GPH [b]	Not PI	-.211	.000***	.000	-	-.225	.000***	.000	-
Unemployment	Not PI	.177	.015**	**.037**	.529	.155	.000***	.000	-
Study Delay	Not PI	-.003	.850	.000	-	.000	-	.000	-

Note. Displaying reference- and increment values for regression models using LASSO and MCP as penalization methods when predicting Life satisfaction. Reference group = female, thus all differences indicate how males compare to females. β_p = coefficient estimate in reference group, β_{p1} = estimated increment for group 1 (males), PI = Prediction invariant. - = value not applicable. Slopes for which the method identified the anticipated PI-status are in bold.

[a] = The expected status of the slope as either predictive invariant or not predictive invariant, [b] = General Psychological Health.

** = p < 0.01, *** = p < 0.001.

$RMSEA_{LASSO} = 0.051$, $RMSEA_{MCP} = 0.038$.

Discussion

Summary of Results

When penalizing the parameters' group differences as means to recognize the PI-status of intercepts, all simulation factors were influential. When looking at the PI-status of slopes, similar results were found – except that the PI-status of the intercept had no effect. Overall, a larger sample size leads to greater accuracy when recognizing the PI-status. Accuracy in correctly recognizing the PI-status of intercept- and slope coefficients was positively related to the number of other coefficients with the same PI-status. Additionally, PI intercepts were more likely to be identified than Not PI intercepts. MCP is positively linked to sensitivity and correctly recognizing PI intercepts and slopes. LASSO is in the same manner positively linked to specificity and correctly recognizing intercepts and Not PI slopes.

Regarding the empirical data, results showed that the difference between LASSO and MCP is that MCP pushed all group differences in slope to zero, enforcing slope invariance to hold, whereas LASSO recognized the PI-status of the slope of Unemployment and Positive Self-Esteem as Not PI. For the former this was in line with theory, but not for the latter. Neither method managed to push for intercept invariance and thus also not strong regression invariance. This is not a surprise seeing that theory suggested that females indeed had higher life satisfaction scores than male subjects.

Implications

Based on the results of the simulation study, in order to recognize a slope's PI-status one would want data that has a large sample size, a higher number of PI slopes and use MCP to push for PI. However, when investigating what penalization method is better on real data one must consider what the goal is. The preferred method depends on whether the goal is to correctly recognize the PI-status of a coefficient, having higher Sensitivity and Specificity, or to push for PI. In the case of wanting to correctly recognize the PI-status, the results indicate that this accuracy highly depends on the number of true PI slopes. When investigating the more practical scenario, it becomes apparent that unlike our simulation study it is here not possible to *know* the true PI-status of the slopes. One can merely make educated guessed based on supporting theory. Thus, the focus here should rather be on seeing how well the methods can push for PI.

Regardless of wanting to have a model where strong regression invariance holds, one would only want to have such a model if it in fact is true. That is, having a perfect sensitivity rate is good, yet not when specificity is at zero. Pushing all variables to be predictive invariant is not justified if there indeed are great group differences between group-intercepts or -slopes in the true model. Consequently, the optimal goal would be to have a model that 1) correctly identifies the PI-status of slopes, and 2) penalize for PI for those Not PI slopes when the differences are negligible. In other words, one should aim for a model with a perfect sensitivity rate and a high specificity rate where the model would push appropriately sized group differences in intercept and slope to zero.

Normally, when using penalization methods to penalize slopes the aim is to pinpoint what variables are stronger predictors and what ones can be left out, i.e., variable selection. In that case, the slopes pushed to zero are considered to have too little impact on the prediction to be included in a final model. In the same fashion, one could use the method presented in this paper to select the variables with predicted PI-status PI and thus get a model where slope invariance holds. In this case, the variables where the group difference is penalized to exact zero are the variables one would wish to keep in the model, rather than leave out, as they are the variables where PI holds.

Strength

First off, this study is one of only a few on the topic and thus is adding to the research currently available. Second, this study investigates how group differences can be recognized as well as minimized through penalization methods – using currently available methods applied to a new problem. Moreover, the study investigates both intercept- and slope coefficients, taking the research beyond PI in the single coefficients – allowing for an investigation of intercept- and slope invariance and ultimately strong regression invariance. Finally, regarding the empirical part of the study, the strength lies in that it is a helpful addition that bridges the gap between how the method works when the PI-status of slopes is known, and the real world when we only can hypothesize the PI-status of slopes.

Limitations and future research

Despite the good things I have to say about the study, there are undoubtfully topics that one could (re)consider. In particular, there are two topics on which there is room for improvement or that simply should be more carefully considered in the future. First, the stimulation set up. Second, the way penalization methods are used.

Regarding the simulation set up, all data simulated are balanced designs, thus one cannot state how different group sized may influence the results. In addition, only two groups are considered. How penalizing parameters' differences between more than two groups influences the results, is yet to be investigated. Further, I have not considered differential multicollinearity in the predictors and how this may influence the results. Seeing how other types of invariances are investigated through a look at correlations, maybe one should consider that here too. Moreover, throughout the simulation study there were always a set of 12 predictive variables. Future research may want to investigate how a larger or smaller set of predictors could influence the results. If we consider the variables and their slopes a little more, it would be of interest to see how the size of the coefficients could influence the detection of PI-status as well as the push for PI.

The second topic regarding limitations and future research concerns how the penalties are applied. I used the Bayesian Information Criterion as loss function for the regularization parameters. Huang (2020, p. 5) suggests several criteria that through LSLX may be applied to select the regularization parameters. An investigation into what selection criterion is indeed best for this method is of interest. In addition, there is a fundamental issue with how the penalties are applied when performing LASSO and MCP regression analysis via LSLX. The penalties put on both the slope parameter and group difference depend on the same regularization parameters. This could be a problem because regression parameters and their group differences are not necessarily on the same scale. A potential solution to this, which in not implemented in LSLX is adaptive LASSO as discussed by Geminiani et al. (2021).

Conclusion

To conclude on the findings and methods proposed in this study it is safe to say it has added insight into how one can use LASSO and MCP to penalize the model parameters' group differences and by that recognize for what variables the prediction is (not) conditional on the group. There are flaws that in the future should be considered, yet one can draw two main conclusions from this study. 1) It is possible to recognize the PI-status of slopes, where the accuracy of this identification depends on the sample size, what penalization method one uses and, the number of coefficients with the same PI-status as the PI-status of the coefficient one wished to recognize. 2) Seeking PI is possible, yet one needs to be careful with the reasoning behind choosing either method, where MCP is best at recognizing PI coefficients and LASSO is best at recognizing Not PI coefficients.

Pushing for PI is possible, yet one should always take other research into account to build a strong theoretical background to support what variables are to be expected having a PI-status PI and what variables are not. An uneducated guess, merely pushing for strong regression invariance could lead to low sensitivity rates where group differences are ignored.

References

Al-Attiyah, A., & Nasser, R. (2016). Gender and age differences in life satisfaction within a sex-segregated society: Sampling youth in Qatar. *Nternational Journal of Adolescence and Youth*, *21*(1), 84–95. https://doi.org/10.1080/02673843.2013.808158

Becchetti, L., & Conzo, G. (2021). The Gender Life Satisfaction/Depression Paradox. *Social Indicators Research*, *160*, 35–113. https://doi.org/10.1007/s11205-021-02740-5

Borsboom, D. (2006). The attack of the psychometricians. *Psychometrika*, *71*(3), 425–440. https://doi.org/DOI: 10.1007/s11336-006-1447-6

Bourbonnais, K., & Durand, G. (2018). The incremental validity of the Triarchic model of psychopathy in replicating “The dark side of love and life satisfaction: Associations with intimate relationships, psychopathy and Machiavellianism”. *The Quantitative Methods for Psychology*, *14*(3), r12–r16. https://doi.org/10.20982/tqmp.14.3.r001

Cleary, T. A. (1968). Test Bias: Prediction of grades of negro and white students in integrated colleges. *Journal of Educational Measurement*, *5*(2), 115–1124.

Cortès-Franch, I., Puig-Barrachina, V., Vargas-Leguás, H., Arcas, M., & Artazcoz, L. (2019). Is being employed always better for mental wellbeing than being unemployed? Exploring the role of gender and welfare state regimes during the economic crisis. *International Journal of Environmental Research and Public Health*, *16*(23), 4799. https://doi.org/10.3390/ijerph16234799

Geminiani, E., Marra, G., & Moustaki, I. (2021). Single- and Multiple-Group Penalized Factor Analysis: A Trust-Region Algorithm Approach With Integrated Automatic Multiple Tuning Parameter Selection. *Psychometrika*, *86*(1), 69–95.

Hanif, S., Ali, M. H., & Shaheen, F. (2021). Religious extremism, religiosity and sympathy toward the Taliban among students across Madrassas and worldly dducation schools in Pakistan. *Terrorism and Political Violence*, *33*(3), 489–504. https://doi.org/10.1080/09546553.2018.1548354

Hastie, T., Tibshirani, R., & Wainwright, M. (2015). *Statistical Learning with Sparsity: The Lasso and Generalizations*. CRC press.

Hetelekides, E., Bravo, A., Burgin, E., & Kelley, M. (2021). PTSD, rumination, and psychological health: Examination of multigroup models among military veterans and college students. *Current Psychology*. https://doi.org/10.1007/s12144-021-02609-3

Huang, P. H. (2018). A penalized likelihood method for multi-group structural equation modelling. *British Journal of Mathematical and Statistical Psychology*, *71*(3), 499–522.

Huang, P. H. (2020). lslx: Semi-Confirmatory Structural Equation Modeling via Penalized Likelihood. *Journal of Statistical Software*, *93*(7), 1–37. https://doi.org/doi: 10.18637/jss.v093.i07

Joshanloo, M. (2018). Gender differences in the predictors of life satisfaction across 150 nations. *Personality and Individual Differences*, *135*, 312–315. https://doi.org/10.1016/j.paid.2018.07.043

Joshanloo, M., & Jovanović, V. (2020). The relationship between gender and life satisfaction: Analysis across demographic groups and global regions. *Archives of Women's Mental Health*, *23*, 331–338. https://doi.org/10.1007/s00737-019-00998-w

Lafeuille, M.-H., Grittner, A. M., Gravel, J., Bailey, R., Martin, S., Garber, L., Duh, M. S., & Lefebvre, P. (2015). Economic simulation of canagliflozin and sitagliptin treatment outcomes in patients with type 2 diabetes mellitus with inadequate glycemic control.

Journal of Medical Economics, *18*(2), 113–125. https://doi.org/10.3111/13696998.2014.980503

Latz, C. A., DeCarlo, C., Boitano, L., Png, C. Y. M., Patell, R., Conrad, M. F., Eagleton, M., & Dua, A. (2020). Blood type and outcomes in patients with COVID-19. *Annals of Hematology*, *99*, 2113–2118. https://doi.org/10.1007/s00277-020-04169-1

Leonard, D. S., Fenton, J. E., & Hone, S. (2010). ABO blood type as a risk factor for secondary post-tonsillectomy haemorrhage. *International Journal of Pediatric Otorhinolaryngology*, *74*(7), 729–732. https://doi.org/10.1016/j.ijporl.2010.03.003

Li, N., & Yang, H. (2019). Nonnegative estimation and variable selection under minimax concave penalty for sparse high-dimensional linear regression models. *Statistical Papers, 62,* 661–680. https://doi.org/10.1007/s00362-019-01107-w

Millsap, R. E. (1997). Invariance in measurement and prediction: Their relationship in the single-factor case. *Psychological Methods*, *2*(3), 248–260.

Millsap, R. E. (2007). Invariance in measurement and prediction revisited. *Psychometrika*, *72*(4), 461–473. https://doi.org/10.1007/S11336-007-9039-7

Radošević, T., Bulatović, G., & Bulatović, L. (2018). Quality of life shown in correlations between significatn socio-personal values of students and employees and their attitudes towards change. *International Journal for Quality Research*, *12*(3), 723–740. https://doi.org/DOI – 10.18421/IJQR12.03-11

Smits, R., D’Hauwers, K., IntHout, J., Braat, D., & Fleischer, K. (2020). Impact of a nutritional supplement (Impryl) on male fertility: Study protocol of a multicentre, randomised, double-blind, placebo-controlled clinical trial (SUppleMent Male fERtility, SUMMER trial). *BMJ Open, 10*. https://doi.org/doi: 10.1136/bmjopen-2019-035069

Suldo, S. M., Minch, D. R., & Hearon, B. V. (2015). Adolescent life satisfaction and personality characteristics: Investigating relationships using a Five Factor Model. *J Happiness Stud*, *16*, 965–983. https://doi.org/DOI 10.1007/s10902-014-9544-1

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

Wood, N., Bann, D., Hardy, R., Gale, C., Goodman, A., Crawford, C., & Stafford, M. (2017). Childhood socioeconomic position and adult mental wellbeing: Evidence from four British birth cohort studies. *PLoS ONE*, *12*(10). https://doi.org/10.1371/journal.pone.0185798

Ye, S., Yu, L., & Li, K. (2012). A cross-lagged model of self-esteem and life satisfaction: Gender differences among Chinese university students. *Personality and Individual Differences*, *52*(4), 546–551. https://doi.org/10.1016/j.paid.2011.11.018

Zhang, C. H. (2010). Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, *38*(2), 894–942.

Appendix A

Abbreviations

Table A1

Variable Abbreviations and Statistics for Empirical Data

Variable Name	Abbreviation	Mean	SD
Life Satisfaction	LifeSat	26.18	5.72
Dysphoria	Dys	1.56	2.66
Euphoria	Euph	6.95	3.04
Positive Self Esteem	PosSE	15.95	3.32
Negative Self Esteem	NegSE	21.79	9.02
Neuroticism	Neuro	25.05	9.38
Disinhibition	Dis	26.51	7.39
Need for Change	NFC	25.44	6.88
Impulsivity	Imp	24.49	4.52
General Psychological Health	GPH	19.80	4.53
Unemployment	Unemp	2.77	0.59
Years of delay in studies	StuD	0.34	0.66

Note. Overview of abbreviations, means, and standard deviations across variables included in the empirical study.

Table A2

Abbreviation used in Paper

Abbreviation	Full Name	Meaning
PI	Predictive Invariance [a] Predicative invariant [b]	[a] An appropriate lack of group difference. [b] A coefficients PI-status indicating homogeneous groups.
MGMR	Multi-Group Multiple Linear Regression	The type of analysis we are applying PI to.
Not PI	Not predictive invariant	A coefficient's PI-status indicating heterogeneous groups.
SEM	Structural Equation Modeling	Modeling technique using latent variables.
LASSO	Least Absolute Shrinkage and Selection Operator	A method used to apply penalties to components of the model, using the λ parameter.
MCP	Minimax Concave Penalty	A method used to apply penalties to components of the model, using the λ- and δ parameters.
N	Sample Size	Simulation Factor: Number of simulated values per dataset. Used when referring to the simulation factor.
$PI_{Intercept}$	PI-status of the intercept	Simulation Factor: Indicator for if the intercept is set to (not-) predictive invariant in the simulation).

Abbreviation	Full Name	Meaning
$\#PI_{Slopes}$	Number of predictive invariant slopes	Simulation Factor: Number of variables in the simulation for which the slope was simulated to be predictive invariant.
$\#NPI_{Slopes}$	Number of *not* predictive invariant slopes	The number of variables in the simulation for which the slope was simulated to *not* be predictive invariant. *In our study, $\#NPI_{Slopes}$ was equal to 12- $\#PI_{Slopes}$.*
$\#T_{Slopes}$	True slopes	The number of slopes where the PI-status was correctly recognized.
$\#TPI_{Slopes}$	True predictive invariant slope	The number of slopes simulated to be predictive invariant in the population and found to be predictive invariant by the model.
#FPI	False predictive invariant	The number of coefficients (here: slopes) simulated to be *not* predictive invariant in the population yet found to be predictive invariant by the model.
$\#TNPI_{Slopes}$	True *not* predictive invariant slope	The number of slopes simulated to be *not*-predictive invariant in the population and found to be *not* predictive invariant by the model.
#FNPI	False *not* predictive invariant	The number of coefficients (here: slopes) simulated to be predictive invariant in the population yet found to be *not* predictive invariant by the model.

Abbreviation	Full Name	Meaning
$T_{Interceot}$	True intercept	Indicator of whether or not the PI-status of the intercept was correctly recognized.
$TPI_{Intercept}$	True *not* predictive invariant intercept	Indicator of whether or not the PI-status of the intercept was correctly recognized as predictive invariant.
$TNPI_{Slopes}$	True *not* predictive invariant intercept	Indicator of whether or not the PI-status of the intercept was correctly recognized as *not* predictive invariant.
CI	Confidence Interval	Range indicating the magnitude of results. *In contrast analysis, values larger than 1 indicate a positive effect, values below 1 indicate a negative effect, and values including 1 indicate no effect.*
AIC	Akaike Information Criterion	Selector used to see how well the model fits the data. The lower the value the better. This value allows for comparison between models. *Here it allows for comparison between LASSO- and MCP models.*
RR	Rate Ratio	Group difference measure.
OR	Odds Ratio	Group difference measure.

Note. Abbreviations used throughout the text, the full name as well as what they refer to in more detail.

Appendix B

R-code

Below is the R code showing how I used LSLX in a function to implement predictive invariance restrains on model coefficients B1) in the simulation study and B2) for the empirical data. Only the code for how LASSO was implemented is depicted, seeing that MCP was implemented in the exact same way.

B1: Method for simulation study

```
Method_LASSO <- function(data){
  # Input: data = data frame with predictors, an outcome and a group variable
  # Output: List containing;  mod.coef = data frame with
  #                                     model coefficients per group
  #                           penalty.lvls = penalty levels
  #                           fit.indices = fit indices for model

  nPred <- ncol(data)-2

  # Empty data frame for storing data
  mod.coef <- data.frame(matrix(NA, nrow = 3, ncol = nPred),
                         inter = c(NA, NA, NA),
                         group = c(0, 1, 999))
  for(i in 1:nPred){
    names(mod.coef)[i] <- paste("beta", i, sep = "")
  }

  rownames(mod.coef) <- c("group = 0",
                          "group = 1",
                          "groupDiff")

  # Specifying penalised regression model
  model_reg <-
    "y <~ x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 + x10 + x11 + x12"

  # Object
  lslx_reg <- lslx$new(model = model_reg,
                       data = data,
                       group_variable = "group",
                       reference_group = 0, verbose = F)

  # Penalising increment components for intercept and residual error
```

```
  lslx_reg$penalize_coefficient(c("y<-1/1", "y<->y/1"), verbose = F)

  # Model fit and error check
  test.1 <- try(lslx_reg$fit(penalty_method = "lasso", verbose = T))
  test.2 <- try(lslx_reg$extract_coefficient_matrix(selector = "bic",
                                                    block = "y<-y"))
  if(inherits(test.1, "try-error") | inherits(test.2, "try-error")){
    # all output stays empty
    penalty.lvls <- NA
    fit.indices <- NA

    print("It didn't work ...")
    print("... no results for you.")
  } else {

    ## Extract model Coefficients
    # Slopes
    mod.coef[mod.coef$group == 0,1:nPred] <-
      as.data.frame(lslx_reg$extract_coefficient_matrix(selector = "bic",
                                                        block = "y<-y")$"0")[
1,

-1]
    mod.coef[mod.coef$group == 1,1:nPred] <-
      as.data.frame(lslx_reg$extract_coefficient_matrix(selector = "bic",
                                                        block = "y<-y")$"1")[
1,

-1]
    # Intercept
    mod.coef$inter[mod.coef$group == 0] <-
      lslx_reg$extract_coefficient_matrix(selector = "bic",
                                          block = "y<-1")$"0"[1, 1]
    mod.coef$inter[mod.coef$group == 1] <-
      lslx_reg$extract_coefficient_matrix(selector = "bic",
                                          block = "y<-1")$"1"[1, 1]

    # Increment
    mod.coef[mod.coef$group ==  999, -ncol(mod.coef)] <-
      mod.coef[mod.coef$group == 1, -ncol(mod.coef)] -
      mod.coef[mod.coef$group == 0, -ncol(mod.coef)]

    ## Extract other information
    penalty.lvls <- lslx_reg$extract_penalty_level(selector = "bic")
    fit.indices <- lslx_reg$extract_fit_index(selector = "bic")

    print("IT WORKED! YOU GOT RESULTS!!!")
  }

  model.summary <- list(mod.coef = mod.coef,
```

```
                          penalty.lvls = penalty.lvls,
                          fit.indices = fit.indices)

  return(model.summary)
}
```

B2: Method for empirical study

```
# Model
model_reg <- "# Regressions
              LifeSat <~ PosSE + NegSE + Dis + NFC + Imp + Neuro + Euph + Dys
+ GPH +  Unemp + StuD
              "
# Initializing object
lslx_reg_LASSO <-lslx$new(model = model_reg,
                          data = data,
                          group_variable = "Sex",
                          reference_group = 0)
# Penalizing increment components
lslx_reg_LASSO$penalize_coefficient(c("LifeSat<-1/1", "LifeSat<->LifeSat/1"))
# Model fit
lslx_reg_LASSO$fit(penalty_method = "lasso")
# Summary
lslx_reg_LASSO$summarize(selector = "bic")
```

Appendix C

LAVAAN to LSLX Syntax Comparison

LSLX	LAVAAN	Command	Example
<=	~	Regression (with the parameters on the RHS FREELY estimating the variable on the LHS)	y <= x1 + x2
<~		Penalized Regression	y <~ x1 + x2
<=>	~~	(Co)variance	x1 + x2 <=> x1 + x2
<~>		Penalized (Co)variance	x1 <~> x1
<=:	=~	Defining a (reflective) latent factor on the RHS	x1 + x2 + x3 <=: f1
<~:	pen()*f =~	Defining latent factors on the LHS but with a penalty	f1 :~> x1 + x2 + x3
fix(1)*	1*	Fixes loading to be 1	fix(1) * x1 + x2 + x3 <=: f1
pen()*		Penalized estimate (overrules <=)	y <= pen()*x1 + x2
free()*		Frees estimate (overrules <~)	y <~ x1 + free()*x2
1		Intercept	y <= 1 + x1 + x2

Note. Displaying a comparison between LSLX- and LVAAN syntax, including examples. f = name of factor, LHS = Left Hand Side, RHS = Right Hand Side.