An alternative to standardizing predictors in the LASSO with an eye on selection psychology

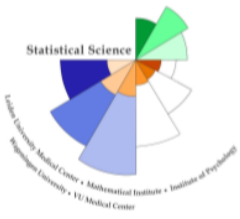
Amy Vervoorn (s1235915)

Thesis advisor: Prof. Henk Kelderman
Second advisor: Prof. dr. M. J. de Rooij

MASTER THESIS

Defended on Month Day, Year

Specialization: Statistical Science



**STATISTICAL SCIENCE
FOR THE LIFE AND BEHAVIOURAL
SCIENCES**

Abstract

A job application process often includes a test battery with several skills and personality tests. The performance on these tests is used to predict an overall job performance score and can help decide whether or not to hire someone. Prediction based on a test battery is often done by ordinary least-squares (OLS) models. OLS models try to correctly explain the relationship between the dependent variable and the tests itself. However, prediction is also important when you want to select the best candidates. OLS models are not sparse and often have high variance, thus it may not be the best model in terms of prediction. To improve prediction, machine learning methods, such as the least absolute shrinkage and selection operator (LASSO) regression, can be used. The LASSO adds bias to estimates and reduces variance to improve prediction. One disadvantage of LASSO regression is that its not scale invariant in the predictors. Therefore, predictors are standardized, typically by using the observed-score variance.

In psychological tests, scores consist of two parts: the error part and the true-score part. The observed-score variance thus also consists of two parts: error variance and true-score variance. The true-score variance part is the most important part for prediction. However, the error variance part can cloud the effect of the true-score variance and influences whether a test is present in the prediction of the LASSO or not.

This study examines two alternatives to standardization by the observed-score variance for the LASSO. The first one standardizes by the true-score variance, to minimize the effect of the error variance in the statistical model for variable selection. The second alternative is a transformation by the ordinary least-squares coefficient, based on the nonnegative garrote model, to add explanatory value to the model and overshadow the effect of the error variance. We examine the truthfulness of variable selection, truthfulness of coefficient size, and prediction accuracy through simulation with multiple scenarios of design factors. Design factors include number of observations, reliabilities of the tests, covariance between latent variables and the number of true nonzero regression coefficient. The methods were also compared with respect to an empirical data set of test results for psychological trait tests measuring general mental health to determine differences and semblencas between real-world data and simulation.

Results showed that the methods act differently under different circumstances. Both alternatives improved the variable selection and truthfulness of coefficients in most scenarios, while the prediction was approximately the same for all three methods. This thesis gives recommendations for which method is best to use in which scenario, and shows the effects of the design factors on the truthfulness of the three methods in the simulation study. Limitations of this simulation study are given together with recommendations for further research.

Contents

Chapter 1

Introduction

Imagine you are applying for a job. You go to the interview hoping to impress. A few days later you get a call for the next round, involving a second interview and a few skill and personality tests. After having done all this, the final call comes with bad news. The test didn't go as well as planned and therefore you aren't selected for the job. But how does that work, declining or hiring someone based on tests? How do they determine your overall success? Could they really predict your true skill level and determine which skills are more important for job performance, and decline you based thereon?

In the field of selection psychology, test batteries are used to predict a criterion such as future job performance to determine whether to hire someone. These test batteries contain several tests, each containing various items. However, the length of this test battery is constrained by time and resources, which has implications for the statistical model and how the scores should be assessed. The statistical model should predict well and be easy to interpret, which means it should be sparse. Sparse models are models for which some of the tests contribute in predicting a criterion (Zhao & Yu, 2006). This makes assessing the test scores easier and faster, because you do not need to examine each test and can eliminate test which are redundant to the criterion. Thus, the test battery should consist of a relatively small number of relevant tests that measure traits and skills most related to the criterion of interest.

1.1 Linear regression model

To date, ordinary linear regression is used to select tests to include in the battery that contribute significantly to the prediction in a large sample of subjects from the population of interest (Shmueli, 2010). The linear regression formula

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + \epsilon \tag{1.1}$$

is used to explain future job performance y from a set of test scores $\{x_j\}_{j=1}^{p}$. The coefficient for each test can be estimated from N number of observations in a sample from the population of interest by minimizing the least square error (LSE), that is,

$$\underset{\beta_0,\beta_j}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip})^2. \tag{1.2}$$

This technique is called ordinary least squares (OLS). A linear regression model has the advantage that the squared error for this model is invariant under a change of units of the independent variables, because multiplying a predictor, say x_1, by a constant c,

$$x_{i1}^* = cx_{i1}, \tag{1.3}$$

can be compensated by dividing the corresponding coefficient by that same constant,

$$\hat{\beta}_1^* = \hat{\beta}/c$$

, so that

$$\hat{\beta}_1 x_{i1} = \hat{\beta}_1^* x_{i1}^*$$

in (1.2) does not change the LSE because β_1 is unrestricted.

A disadvantage of the linear regression model is that it is not sparse. It gives non-zero estimates for virtually all coefficients, which means all tests attribute to the prediction, making interpretation difficult when the number of tests p is large. If the number of tests administered is larger than the number of people on which you base the statistical model ($p > N$), the estimates are not unique and the model will over-fit the data (Hastie, Tibshirani, & Wainwright, 2015, p. 2).

There are methods to select relevant tests based on a ordinary least squares, for example, a best-subset regression (Hastie, Tibshirani, & Friedman, 2009, p. 57) or step-wise selection (Hastie et al., 2009, p. 58), where the linear regression model goes through multiple steps of eliminating variables before it reaches the final regression model. However, these methods show high variability in the selection and can be computationally time-consuming when the number of tests is large (Breiman, 1995; Zou, 2006; Zhao & Yu, 2006).

Furthermore, in the field of psychology, statistical models, like ordinary least squares, are most concerned with causal explanation of the criterion. This type of analysis is called explanatory modeling. Explanatory modeling seeks to find a true model to describe the relationship between the tests and the criterion (Shmueli, 2010; Yarkoni & Westfall, 2017). To achieve this goal, unbiased estimates are important. The ordinary least-squares models can produce unbiased estimates of its parameters with the smallest variance as compared to other least-squares models (Hastie et al., 2009, p. 51), when all the assumptions for linear regression are met. Often in practice, not all assumptions are met, so there is some form of bias in the estimators. Still, the least-squares estimators often have low bias. The focus of these models is to minimize bias, with enlarged variance as a consequence. Due to the larger variance, the prediction of the criterion could suffer. An estimator could predict better if there was a bit of bias introduced to reduce the variance (Hastie et al., 2009, p. 52).

1.2 LASSO regression model

Another outlook on statistical modeling is predictive modeling. Predictive modeling focuses on finding a model that provides optimal predictions of the criterion for 'new' people. To optimize predictions, bias is added to the estimators of the model to decrease variance (Bleidorn & Hopwood, 2018; Yarkoni & Westfall, 2017; Breiman, 2001; Waljee, Higgins, & Singal, 2014). To decrease variance and add bias, one can constrain the regression coefficients by using a penalized regression analysis. One example is the Least Absolute Shrinkage and Selection Operator (LASSO) (Tibshirani, 1996). The LASSO finds the solution $(\hat{\beta}_0, \hat{\beta}_j)$ that minimizes the residual sum of squares subject to a penalty placed on the sum of the absolute values of the coefficients, that is,

$$\underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{N} (y_i - \beta_0 - x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}. \tag{1.4}$$

The penalty parameter λ limits how well the model fits on the data, how complex the model is, how many coefficients are 'free' to be nonzero and how large they can be. This value must be specified by an external procedure. The LASSO intentionally produces biased shrunken estimates, and produces some coefficients that are exactly 0, making it easier to interpret (Tibshirani, 1996; Hastie et al., 2015, p. 8). It does the estimation of the coefficients and the variable selection simultaneously, and therefore does not suffer the drawbacks of best-subset regression or step-wise selection (Zou, 2006). More and more, predictive models such as the LASSO have been used to powerfully predict human behaviour and personality traits (Bleidorn & Hopwood, 2018; Yarkoni & Westfall, 2017).

The LASSO is consistent regarding variable selection when certain requirements are met within the data, such as sparsity in the true regression model and high dimensionality (Meinshausen, Bühlmann, et al., 2006). A study looked into the consistency of variable selection (van Loon, 2016) and concluded the LASSO can find 100% of the true predictors when conditions are favorable, and less than 1% if the conditions are unfavorable. A high sample size, the use of a balanced design, and a high signal-to-noise ratio counted as favorable conditions (van Loon, 2016). Another study looked at LASSO in terms of model selection, and found this mainly depends on the covariance of the predictor variables (Zhao & Yu, 2006). They showed that LASSO may not be able to distinguish the correct test, if a test with no predictive value shares a high covariance with a relevant test in the true model. Thus, the LASSO gives the best results regarding variable selection if we have many independent potential tests for many people, where only some contribute to the prediction of the criterion.

1.3 Problem with penalized regression

There is, however, a problem applying penalized regression models that ordinary least-squares models do not have. In penalized regression models, prediction error is not invariant under linear scale transformation of the predictor variables, and thus the LASSO depends on how the tests are measured (Adams, Fagot, & Robinson, 1965; Hastie et al., 2015, p. 9). For the LASSO, we have to minimize the least

square error as in (1.4). If both $\beta_1 > 0$ and $0 < c < 1$, a change in the first predictor will cause

$$
\begin{aligned}
|\beta_1^*| + \sum_{j=2}^{p} |\beta_j| &= \frac{|\beta_1|}{c} + \sum_{j=2}^{p} |\beta_j| && (1.5)\\
&= \frac{|\beta_1|}{c} - |\beta_1| + \sum_{j=1}^{p} |\beta_j| && (1.6)\\
&= |\beta_1|(1/c - 1) + \sum_{j=1}^{p} |\beta_j| && (1.7)\\
&\geq \sum_{j=1}^{p} |\beta_j|. && (1.8)
\end{aligned}
$$

Because of the change of scale of $x_{i1}^* = cx_{i1}$, the corresponding regression coefficient becomes larger. It therefore imposes an additional penalty on the penalized loss function. If the unit of the scale is larger (e.g. a 1 through 7 scaled test versus a dichotomous test), the numerical changes of the predictor become small relative to the changes in scale itself and corresponding regression coefficient for that test become larger too. Because the outcome of the LASSO variable selection procedure is not invariant under a change of unit of the predictor variables, the selection becomes dependent on the arbitrary scale on which the variables are measured.

To overcome this effect, the predictors are standardized like in formula 1.9 (Tibshirani, 1996).

$$x_j^* = \frac{x_j - \bar{x}_j}{\sqrt{Var(x_j)}} \tag{1.9}$$

The predictors are then said to measure on 'the same scale'. However, this means that the estimated regression equation does not only depend on the conditional distribution $f(y|x)$, but also on $f(x)$. Furthermore, it makes the scale on which the predictors are measured dependent on the reliability of the test. Increase the reliability and the scale changes as well, affecting the size of the regression coefficients and the effect of penalization. This effect is further discussed in the next section.

1.4 Error in psychological tests and error variance

Standardizing the scores of psychological tests can be problematic because the scores are prone to measurement error. Think of social desirability, a tendency to avoid extreme answers on a personality test, or a strict interviewer versus a lenient one scoring the interview. As a result, any selection and interpretation based on these tests can be the result of a wrong decision because of error (Borsboom, Romeijn, & Wicherts, 2008). So, how can you best analyze these kinds of tests?

The analysis of observed test scores is based on Classical Test Theory (Mellenbergh, 2011, p. 108). Classical Test Theory gives a foundation of theories on how observed test scores can represent actual skills and how to analyze the test scores. In Classical Test Theory, observed test scores are assumed to consist of two parts: the true score and an error (Mellenbergh, 2011, p. 110),

$$x_j = T_j + E_j, \quad T_j \perp E_j, \qquad j = 1, \ldots, p, \tag{1.10}$$

where T_j and E_j are the true and error part of the observed score x_j. Since T_j and E_j are independent ($T_j \perp E_j$) (Mellenbergh, 2011, p. 118), we can write the observed-score variance as

$$Var(x_j) = Var(T_j) + Var(E_j), \quad j = 1, \ldots, p. \tag{1.11}$$

Standardizing by the observed-score variance as in formula 1.9 is thus equal to standardizing by the true-score variance but also by the error variance. Substituting (1.11) in (1.9) yields

$$x_j^* = \frac{x_j - \bar{x}_j}{\sqrt{Var(T_j) + Var(E_j)}}. \tag{1.12}$$

In selection psychology, the aim is to select candidates based on their true skills. A focus on prediction is thus important. However, predictive modeling is focused on precision in prediction, which could lead to limiting the theoretical accuracy and explanation of the relationship (Bleidorn & Hopwood, 2018). If the explanation of the relationship is clouded by the focus on prediction, how can you adequately determine which skills are most important for job performance? When a candidate is rejected because of the test battery, they may want to work on a particular skill to improve their overall predicted job performance. It would be helpful for them to know which skill has a high true effect on their job performance.

The true-score portion of each answer for the test represents the true skills. The error variance in the tests can cloud the size of the effect that particular test has on job performance in the linear model, and can sometimes discriminate against certain (groups of) people (Borsboom et al., 2008). Think about it this way: when the error variance of a test increases, while the true-score variance remains the same, the relative quantity of observed variance taking part in the prediction becomes smaller, and the regression coefficient will decrease for that test. Making the test "less important" in the prediction, but only because of the error variance of that test.

Because in practice, we do not always know the true-score variance, only the observed-score variance, the relative quantity of true-score variance is represented by the reliability of a test. A test with reliability ρ_j has a true-score variance of $Var(T_j) = Var(x_j)\rho_j$ and error variance of $Var(E_j) = Var(x_j)(1 - \rho_j)$. Under standardization, the tests that are relatively unreliable will have a smaller part of the observed variance taking part in the prediction and, thus, a smaller corresponding regression coefficient. This small coefficient is prone to being penalized too much in the LASSO model. This is undesirable if the objective is to select tests that best measure the construct. Under Classical Test Theory, the reliability of a test can be changed by adding or removing items (Mellenbergh, 2011, p. 128).

The effect of error variance is often seen as a source of variance to be minimized within the test itself, by increasing the reliability. But what if we can minimize its effect further through the statistical model? As alternatives to standardization before fitting the LASSO by the observed-score variance, we explore the options of standardizing by the true-score variance instead and correcting the data by ordinary least-squares coefficient, based on the nonnegative garrote model (Breiman, 1995). These two methods will be further discussed in the next two sections.

Standardizing by the true-score variance is expected to eliminate the effect of measurement error on variable selection. The nonnegative garrote (Breiman, 1995) has a close relationship to the LASSO, as it both shrinks and zeroes coefficients. Its coefficients are constructed of ordinary least-squares coefficients and shrunken nonnegative coefficients. It will help increase the explanatory value of the statistical model, increasing a focus on explanation, while hopefully not worsen the prediction.

1.5 Standardizing by the true-score variance

To overcome the penalty put on less reliable tests, it could be more appropriate to standardize by the true-score variance only,

$$x_j^{\dagger} = \frac{x_j - \bar{x_j}}{\sqrt{Var(x_j) * \rho_j}}. \tag{1.13}$$

The error variance no longer poses a problem. In the linear regression model, the error variance is mostly represented in the residual variance, and the observed test variance represent the true-score variance, meaning the true scores do most of the predicting, that is,

$$\begin{aligned}
y &= \beta_0 + \sum_{j=1}^{p} \beta_j x_j^{\dagger} + \epsilon^{\dagger} && (1.14)\\
&= \beta_0 + \sum_{j=1}^{p} \beta_j \frac{x_j}{\sqrt{Var(T_j)}} + \epsilon^{\dagger} && (1.15)\\
&= \beta_0 + \sum_{j=1}^{p} \beta_j \frac{T_j + E_j}{\sqrt{Var(T_j)}} + \epsilon^{\dagger} && (1.16)\\
&= \beta_0 + \sum_{j=1}^{p} \beta_j \frac{T_j}{\sqrt{Var(T_j)}} + \sum_{j=1}^{p} \beta_j \frac{E_j}{\sqrt{Var(T_j)}} + \epsilon^{\dagger} && (1.17)\\
&= \beta_0 + \sum_{j=1}^{p} \beta_j \frac{T_j}{\sqrt{Var(T_j)}} + \epsilon^{\star}. && (1.18)
\end{aligned}$$

Thus, in the population, the errors do no longer exercise an influence specific for a certain test and the variable selection is less influenced by the reliability of the test.

1.6 Transformation by the ordinary least squares

The second alternative method we chose to explore takes an explanatory approach to a predictive model. The ordinary least squares coefficients have low bias, and thus show a more accurate relationship between the test and the criterion. This approach is based on the nonnegative garrote model. The original nonnegative garrote estimator introduced by Breiman (1995) is a scaled version of the least squares estimate. The model is a two-stage procedure; first an initial estimate of coefficients based on ordinary least squares is made, then in the second step, these

coefficients are transformed through multiplying by a nonnegative factor c (Breiman, 1995; Xiong, 2010). The shrinkage factor c_j is given to minimize

$$\begin{aligned} &\underset{\beta_0, c_j}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} c_j \beta_j^{LS} x_{ij})^2, \\ &\text{subject to } c_j \geq 0 \quad \text{and} \sum_{i=1}^{N} (c_j \leq s) \quad \forall j, \end{aligned} \tag{1.19}$$

where β_j^{LS} is the least squares estimate, and s the tuning parameter. In the second stage, regularization is applied to the coefficients of these transformed variables ($\beta_j^{LS} x_{ij}$. This results in the model being scale invariant (Hastie et al., 2015, p. 21). For those coefficients whose full least squares estimate is large, the shrinking factor will be close to 1, meaning no shrinkage. But for a redundant predictor, the least squares estimate is likely to be small and consequently the shrinking factor will have a good chance of being exactly 0 (Yuan & Lin, 2007).

In this study, we propose a manner of transformation based on the nonnegative garrote combining the ordinary least squares with the LASSO regression. In the first step, a linear regression model is fitted on the data, and the coefficients of that model are used to alter the data. This method suffers some of the same downfalls ordinary least squares has. With a small sample size, ordinary least squares estimates may perform poorly, and the nonnegative garrote is expected to suffer as well. As for ordinary least squares, the nonnegative garrote cannot be applied when the sample size is smaller than the number of tests (Yuan & Lin, 2007).

Chapter 2

Methods

We compare three methods; (1) LASSO after standardization by the observed-score variance, (2) LASSO after standardization by the true-score variance, and (3) LASSO after transformation of the predictors by the ordinary least-squares coefficients. These methods are compared with respect to two questions:

1. How do the methods compare with respect to the truthfulness of model selection?

2. How do the differences between the methods for question 1 interact with various design factors?

2.1 Simulation procedure

To compare the methods and answer which method is more truthful and what influences truthfulness, we are going to simulate a test battery for job performances. The test battery consists of 20 tests, each containing 25 items ($k = 1, ..., r$). We simulate on an item level because we need to calculate the reliability for each test to standardize by the true-score variance. We sample on item level based on a single-factor model (McDonald, 2013, p. 78). The single-factor model is based on the idea that items explain a common underlying trait or skill, represented by the latent variable (Θ_{ij}). That latent variable has a different unknown value for each person, as each person has a different amount of a certain skill or trait:

$$x_{ijk} = \mu_j + \lambda_{jk} * \Theta_{ij} + E_{ijk}. \tag{2.1}$$

In this model, x_{ijk} represents the observed item scores. Items differ in their mean in the population of interest (μ_j). Some items measure the true skill more sensitively than others, and are therefore more representative of the latent skill. How well items differentiate the latent skill is represented by the factor loading (λ_{jk}). Furthermore, each item has their own amount of error. Combining (1.10) and (2.1), the true-score part of the items is captured in the mean, factor loading and latent variable of the single-factor model. To simplify the single-factor model, all the test items and scores are measured on a continuous scale. Furthermore, all test contain parallel items. Parallel items are items that have the same factor loading and identical error variances for that particular test.

We run a simulation with a full factorial design, with in total 54 conditions. Each condition is repeated 100 times. For each simulation, an outcome variable is computed to represent the score on a criterion like job performance. The outcome variable is computed trough a linear regression model of some of the predictor variables, like in (1.1).

The simulation procedure is as follows:

1. Specify N values of the latent variable for each test based on a multivariate normal distribution $\Theta \sim N(0, \Sigma)$, where Σ is a matrix with off diagonal entries being the covariance between the latent variables and diagonal entries are the variance of the latent variable, which is fixed on 1.

2. Specify factor loadings λ_{jk} based on the reliability for each test (see Appendix A).

3. Specify N independent and identically distributed (iid) errors for each item for each test based on a normal distribution $E \sim N(0, 1 - \lambda_{jk}^2)$.

4. Obtain the item scores for each test based on the single-factor model as in (2.1), where μ_j is fixed on 0 in this study.

5. Create the sum-scores of each observed score for each item to gain the observed test scores.

6. Set the vector of length p for the regression coefficients, where the first ℓ are one, and $p - \ell$ are zero.

7. Create the output variable y through a true regression model as in (1.1) The linear model includes iid residuals $\epsilon \sim N(0, 25)$.

2.2 Design factors

In the simulation, we want to assess the influence of various design factors.

1. Number of observations (N, having three levels: 50, 100, 500)

2. Reliability (ρ_j, having two levels: low reliabilities (between 0.4 and 0.6), high reliabilities (between 0.7 and 0.85))

3. Number of true nonzero regression coefficients (ℓ, having three levels: $\ell \in \{4, 8, 12\}$ out of $p = 20$)

4. Covariance between latent variables ($\sigma_{\Theta_j \Theta_j}$, having three levels: 0, 0.4, 0.8)

We experiment with the number of observations because this can have a great effect on statistical models. Ordinary least-squares models perform better if the number of observations is large. Research has shown that modern methods, such as the LASSO, are able to outperform traditional models such as OLS regression in prediction accuracy where the sample size to predictor ratio is low. When the sample size increased, the modern methods only showed slightly less error (Putka, Beatty, & Reeder, 2017). We thus expect the number of observations to have a

different effect on the least-squares transformation than on the other two methods, because this methods is most like OLS.

Secondly, we examine the effect of the reliability. The reliability will represent the amount of true-score variance for each test. The true-score standardization excludes the error variance, and only refers to the true-score variance. Therefore, we would like to see what effect the reliability has on the difference between observed-score standardization and true-score standardization on the truthfulness of the model.

Thirdly, the number of true nonzero regression coefficient is manipulated to examine what happens with the size of the coefficients when more tests attribute to the prediction and more coefficient could be 'free' to be nonzero in the LASSO regression methods.

Lastly, the covariance between the latent variables will be manipulated to examine the effect of multicollinearity on the three different methods. A drawback of LASSO models is they have a lessened ability to deal with multicolinearity among predictors (Tibshirani, 1996; Putka et al., 2017). The LASSO regression has more difficulty estimating parameters accurately when the covariance is relatively higher as compared to other methods (Zhang, Yin, & Xiong, 2014).

2.3 Statistical Analysis

When the data set is simulated, the analysis can begin. The analysis will differ between the methods. For the observed-score standardization and true-score standardization, the analysis will be as follows:

1. Calculate the observed-score variance or the true-score variance for each test.
2. Standardize the observed score by either the true-score variance or the observed-score variance.
3. Fit the LASSO model on the standardized data, using a penalty parameter that provides the simplest model within one standard error of the most optimal model (the "one-standard-error" rule) (Hastie et al., 2009, p. 244).
4. Compare the $\hat{\beta'}s$ with the true regression model coefficients.

To compute the true-score variance of each test, we use the reliability ($Var(T_j) = Var(x_j)\rho_j$). The most used and most known reliability measure is Cronbach's alpha (Cronbach, 1951). However, this is not the measure we use in this study. A study comparing reliability methods (Oosterwijk, 2016) concluded that reliability measures often estimate with imprecision. And thus, it is important to find the measure with the least amount of imprecision. Cronbach's alpha estimates was found to be less precise than other methods. Oosterwijk (2016) concluded the most favorable reliability measure is Guttman's Lambda 2 (Guttman, 1945):

$$\lambda_2 = 1 - \sum_{k=1}^{r} \frac{\sigma_k^2}{\sigma_x^2} + \sqrt{\frac{\frac{r\sum_k^r \sum_{k'}^r \sigma_{kk'}^2}{(r-1)}}{\sigma_x^2}}, \text{where } k = 1, ..., r, \text{ and } k \neq k'. \tag{2.2}$$

Lambda 2 uses the number of items of a test (r) and the covariances between items ($\sigma^2_{kk'}$) for a given person to calculate the reliability. It gives higher estimates and concerns parallel tests, therefore differs from Cronbach's alpha. Guttman's Lambda 2 had the best combination of high precision and low bias (Oosterwijk, 2016). Therefore, this study uses Guttman's Lambda 2 as a reliability measure.

For the least-squares transformation, the analysis will look different for the first two steps, and will be as follows:

1. Calculate the least-squares linear-regression model and extract the coefficients β_j^{LS}.

2. Multiple the testscores with the absolute size of the corresponding coefficient $Z_j = |\beta_j^{LS}| * x_j$. We chose to correct by the absolute size of the coefficients, because we did not want the sign of the least-squares coefficient to affect the sign of the LASSO coefficients.

The third and fourth step of the analysis are the same for all three methods, as described in the beginning of this section.

The truthfulness of model selection by the methods will involve three assessments;

1. The first assessment is going to be the proportion of correctly identified tests for each method. This assessment is based on the sensitivity and the specificity (Parikh, Mathai, Parikh, Sekhar, & Thomas, 2008). Sensitivity is the proportion of tests having $\beta_j = 1$ in the true regression model for which the LASSO methods accurately predicts them as taking part in the prediction by estimating $\hat{\beta}_j \neq 0$. Specificity is the proportion of tests having $\beta_j = 0$ in the true regression model for which the LASSO methods accurately predicts them as not taking part in the prediction by estimating $\hat{\beta}_j = 0$. The sensitivity and specificity is a proportion between 0 and 1. The best results are a sensitivity and a specificity of 1.

2. The second assessment is the root mean square error (RMSE) between the set of observed coefficients by the LASSO methods and the true coefficients $RMSE_{coef} = \sqrt{\frac{\sum_{j=1}^{p}(\beta_j - \hat{\beta}_j)^2}{p}}$. The RMSE is also known as the Euclidean distance and is therefore easy to interpret.

3. The third assessment concerns prediction RMSE, between the predicted dependent variable and the actual dependent variable $RMSE_{prediction} = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N}}$.

We hypothesize true-score standardization will better select the model than the other methods in most scenarios of design factors. Given that the transformation in true-score standardization does not depend on the error variance, we expect true-score standardization will have less error than the observed-score standardization. Ordinary least-squares transformation is expected to have more error than true-score standardization, but less than observed-score standardization in most scenarios in terms of variable selection. Because the least-squares transformation is based on an explanatory model, we hypothesize the euclidean distance between the true regression coefficients and the observed regression coefficient will be smaller for the least-squares transformation than for the other two methods. We expect the largest

coefficient RMSE can be found for observed-score standardization, because that method relies more on error variance than true-score standardization.

2.4 Empirical Data

Furthermore, the three methods are compared with respect to an empirical data set. This data set contains data of 1204 respondents on 9 different tests on psychological attributes related to psychological well-being in young adults. The personality trait tests measure traits such as *impulsivity*, *life satisfaction*, *need for change in work environment*, and more, to predict general mental health. Furthermore, it contains background information about peoples' gender and age.

Because this is real world data, there are some differences between this data set and the simulation data. First of all, all the items were measured on a categorical scale. Some test have items measured on a dichotomous scale, other test use a scale from 1 to 5 or a scale from 1 to 7. Secondly, not all tests have the same length. Most test have 6 to 11 questions, quite a bit shorter than the tests in the simulation. In the simulation, design factors can be manipulated. However, in a real-world setting, some design factors are unknown, like the true size of the regression coefficients, the covariance between the latent variables, and the true number of nonzero regression coefficients. Other design factors can be manipulated in the design of the study, such as the number of observations and the reliabilities of the tests (by adding or deleting observations). We have decided not to manipulate any design factors, but rather keep them fixed and look at the whole of the data. The reliability for (most of) the tests lie between 0.783 and 0.874. However, for the test measuring *impulsivity*, the reliability is far lower than the other test (0.371). Because we do not know the true underlying latent skill of each person for each test, the covariance between them cannot be measured. However, we do look at the correlation between the observed tests scores to give an indication of how strong the tests are connected. The correlations between the testscores are in general low (between -0.001 and 0.330), with the highest correlation between the tests *disinhibition* and *need for change* at 0.446. The goal of the empirical data set is to study what the LASSO methods do in a real-world test battery and if the results concur with the simulation study.

2.5 Influence of the methods and design factors on the assessment measurements

We want to quantify the effect of the design factors and the methods on each of the measures of accuracy. Therefore, we fit a repeated-measures analysis of variance (Anova) (Type II Sum of Squares). The repeated-measures Anova assumes an underlying normal distribution of the dependent variable. Since the assessments are not measured on a normal distribution, we had to transform the values first to resemble a normal distribution before fitting the repeated-measures Anova.

Another assumption of repeated-measures Anova is the assumption of sphericity. This assumption is met when the sum of variances minus covariance of the three methods (OS, TS, and LS) are equal within each combination of the design factors. The data violates this assumption, so the p-values are to be taken with a grain of salt as they appear to be smaller then they are.

We assess the size of the effect of the design factors and the methods by the partial eta squared (η_p^2).

$$\eta_p^2 = \frac{SS_{effect}}{SS_{effect} + SS_{error}} \tag{2.3}$$

Partial eta squared is an effect size based on the amount of variance one effect has while the variance of the other effects are left out of the equation. It is an effect size that is often used in repeated-measures Anova (Richardson, 2011).

2.6 Software

This simulation study and all analyses were done using the open source software R (version 3.6.0), in the RStudio environment. Different software packages were used, such as *glmnet* (to execute the LASSO methods), *mtvnorm* (to simulate from a multivariate normal distribution), *Lambda4* (to calculate Guttman's Lambda 2), and *lme4* and *sjstats* (for the repeated-measures Anova and to calculate the effect size).

Chapter 3

Results

3.1 Sensitivity

Table 3.1 shows the results of the repeated-measures Anova with the effect sizes for each main and interaction effect. This section will discuss the larger effects on sensitivity, and the next sections will explore the larger effects for the other measures of accuracy.

Based on table 3.1, we can see that all interaction effects have a significant effect on sensitivity, with the largest effect size for the interaction between covariance and methods and the number of observations and methods. The effect of the design factors is visualized in figures 3.1 and 3.2. The actual numbers for sensitivity and specificity can be seen in Table B.1 and B.2 in Appendix B.

For the three methods, the difference in sensitivity is most observable when the number of observations is small. When the number of observations is large, the sensitivity of the methods increases, and can even reach a complete accuracy. This effect of number of observations seems largest for ordinary least-squares transformation LASSO. When the number of observations is small, the true-score standardization LASSO and ordinary least-squares transformation LASSO suffer more than the observed-scores standardization LASSO in terms of sensitivity.

When we focus on the covariance in figures 3.1 and 3.2, the sensitivity increases when the covariance increases for the observed-score standardization and the true-score standardization LASSO. When the covariance is small to nonexistent and the number of observations is low, the ordinary least-squares transformation method outperforms the other methods. However, when the covariance increases, the observed score standardization and the true-score standardization methods outperform the ordinary least-squares transformation LASSO.

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)	η_p^2
reliability	12.38	12.38	1	46	29.58	1989e-06	0.122
cov	3.85	1.92	2	46	4.60	0.0151	0.041
ℓ	5.90	2.95	2	46	7.05	0.0021	0.062
N	73.99	37.00	2	46	88.43	$<$ 2.2e-16	0.453
method	284.59	142.29	2	16130	340.10	$<$ 2.2e-16	0.040
reliability:method	8.36	4.18	2	16130	9.99	4.624e-05	0.001
cov:method	286.40	71.60	4	16130	171.13	$<$ 2.2e-16	0.041
ℓ:method	43.59	10.90	4	16130	26.05	$<$ 2.2e-16	0.006
N:method	105.34	26.33	4	16130	62.94	$<$ 2.2e-16	0.015

Table 3.1: Results of repeated-measures Anova for sensitivity

cov = covariance, ℓ = number of true nonzero regression coefficients, N = number of observations, Sum Sq = Sum of Squares, Mean Sq = Mean Squares, NumDF = degrees of freedom for numerator, DenDF = degrees of freedom for denominator, η_p^2 = partial eta squared.

Figure 3.1: The effect of design factors on sensitivity and specificity when reliability is high. The boxes represent the design factors number of true nonzero regression coefficients (ℓ = 4, 8, 12) and covariance of latent variables (cov = 0, 0.4, 0.8)

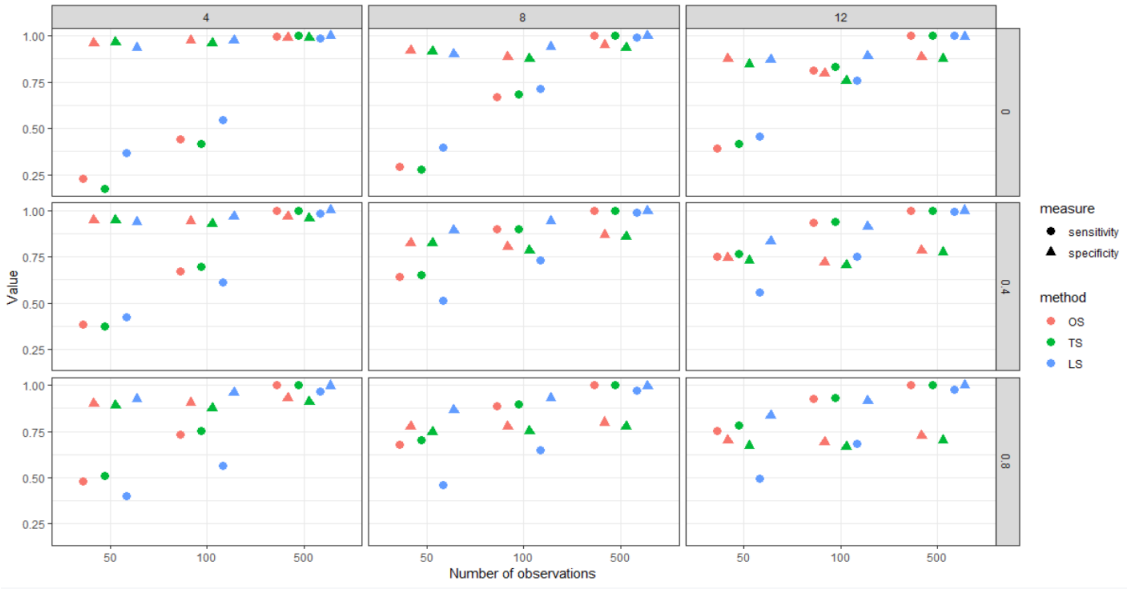


Figure 3.2: The effects of design factors on sensitivity and specificity when reliability is low. The boxes represent the design factors number of true nonzero regression coefficients ($\ell = 4, 8, 12$) and covariance of latent variables (cov $= 0, 0.4, 0.8$)

3.2 Specificity

The effect sizes and the results of the repeated-measures Anova are shown in table 3.2. All interaction effects are significant, with the largest effect sizes for the interaction between methods and number of true nonzero regression coefficient and the interaction effect between covariance and methods.

The interaction with the largest effect size is the interaction between number of true nonzero regression coefficients and the methods. As the number of true nonzero regression coefficients increases, the differences between the methods are largest. With the largest increase in specificity for ordinary least-squares transformation LASSO (Figures 3.1, 3.2).

One effect we can see in figures 3.1 and 3.2 is that the specificity and sensitivity act differently for each method. For ordinary least-squares transformation LASSO, the specificity increases as the number of observations increases (Figures 3.1, 3.2) for each combination of number of true nonzero regression coefficients, reliability and covariance. However, for observed-score standardization and true-score standardization LASSO, the specificity decreases as the number of observations increases when the number of true nonzero regression coefficients is large. When looking at the variable selection, there is a relationship between the sensitivity and the specificity. Both should be 1 to have an optimal variable selection. For the ordinary least-squares transformation method, we see both sensitivity and specificity increase when the number of observation increases (Figures 3.1, 3.2). This method reaches an optimal form of variable selection when the number of observations is 500. However, for the observed-score and true-score standardization method, when the number of observations increases, the sensitivity increases, yet the specificity decreases (Figures 3.1, 3.2) in situations where the number of true regression coefficients is large. For a small number of observations (50), the observed-score standardization and true-score standardization LASSO tends to give a large penalty parameter, constraining too many

tests to 0 effect. This results in a high specificity, but low sensitivity. When the number of observations in higher (500), the penalty parameter decreases, and lesser tests are constrained to 0 effect, resulting in a large sensitivity, but small specificity, especially when the number of true nonzero regression coefficients is larger.

The specificity decreases as the covariance increases for the observed-score standardization and the true-score standardization LASSO when the reliabilities are high (Figures 3.1, 3.2). For the ordinary least-squares transformation LASSO, the specificity increases as the covariance increases when the reliabilities are high. However, when the reliabilities are low, the specificity decreases as the covariance increases for all three methods.

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)	η_p^2
Reliability	21.93	21.93	1	46	39.57	1.062e-07	0.020
cov	97.30	48.65	2	46	87.76	< 2.2e-16	0.083
ℓ	257.39	128.69	2	46	232.16	< 2.2e-16	0.139
N	71.26	35.63	2	46	64.28	4.773e-14	0.062
method	2181.76	1090.88	2	16130	1967.97	< 2.2e-16	0.196
Reliability:method	112.36	56.18	2	16130	101.35	< 2.2e-16	0.013
cov:method	369.91	92.48	4	16130	166.83	< 2.2e-16	0.049
ℓ:method	507.82	126.96	4	16130	229.03	< 2.2e-16	0.054
N:method	143.27	35.82	4	16130	64.62	< 2.2e-16	0.016

Table 3.2: Results of repeated-measures Anova for specificity

cov = covariance, ℓ = number of true nonzero regression coefficients, N = number of observations, Sum Sq = Sum of Squares, Mean Sq = Mean Squares, NumDF = degrees of freedom for numerator, DenDF = degrees of freedom for denominator, η_p^2 = partial eta squared.

3.3 Coefficient RMSE

Table 3.3 shows the results of the repeated-measures Anova for the coefficient RMSE. Comparing the methods in figure 3.3, we can see the ordinary least-squares transformation method creates the smallest Euclidean distance between the true regression coefficients and the observed regression coefficients throughout all the design factor combinations.

A significant interaction with high effect size is the reliability with the methods (Table 3.3). When the reliabilities of the tests are low, the RMSE for the observed-score standardization and the true-score standardization LASSO is smaller (Figure 3.3). The difference between those methods is most prominent when the reliabilities are low. The effect of the reliabilities is strongest for true-score standardization LASSO than for observed-score standardization and least-squares transformation LASSO.

Effects of the number of true nonzero regression coefficient are also visible in interaction with the methods. The fewer nonzero regression coefficients, the smaller the RMSE (Figure 3.3). Comparing the observed-score standardization method with the ordinary least-squares method, this effect is larger for the observed-score standardization method than for the ordinary least-squares method. This effect does not

differ between the observed-score standardization and the true-score standardization methods.

The last interaction we discuss for the coefficient RMSE is the interaction between covariance and the methods. As the covariance increases, the coefficient RMSE increases for observed-score standardization and true-score standardization LASSO, while the coefficient RMSE stays relatively constant for the ordinary least-squares transformation method. The differences between the methods are thus most visible when the covariance is large.

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)	η_p^2
Reliability	19.65	19.65	1	46	262.99	< 2.2e-16	0.596
cov	5.50	2.75	2	46	36.82	2.837e-10	0.292
ℓ	26.04	13.02	2	46	174.25	< 2.2e-16	0.661
N	0.02	0.01	2	46	0.15	0.8580	0.002
method	7698.55	3849.27	2	16130	51518.85	< 2.2e-16	0.865
Reliability:method	999.48	499.74	2	16130	6688.55	< 2.2e-16	0.453
cov:method	248.82	62.20	4	16130	832.54	< 2.2e-16	0.171
ℓ:method	1060.33	265.08	4	16130	3547.87	< 2.2e-16	0.468
N:method	48.88	12.22	4	16130	163.56	< 2.2e-16	0.039

Table 3.3: Results of repeated-measures Anova for coefficient RMSE

cov = covariance, ℓ = number of true nonzero regression coefficients, N = number of observations, Sum Sq = Sum of Squares, Mean Sq = Mean Squares, NumDF = degrees of freedom for numerator, DenDF = degrees of freedom for denominator, η_p^2 = partial eta squared.

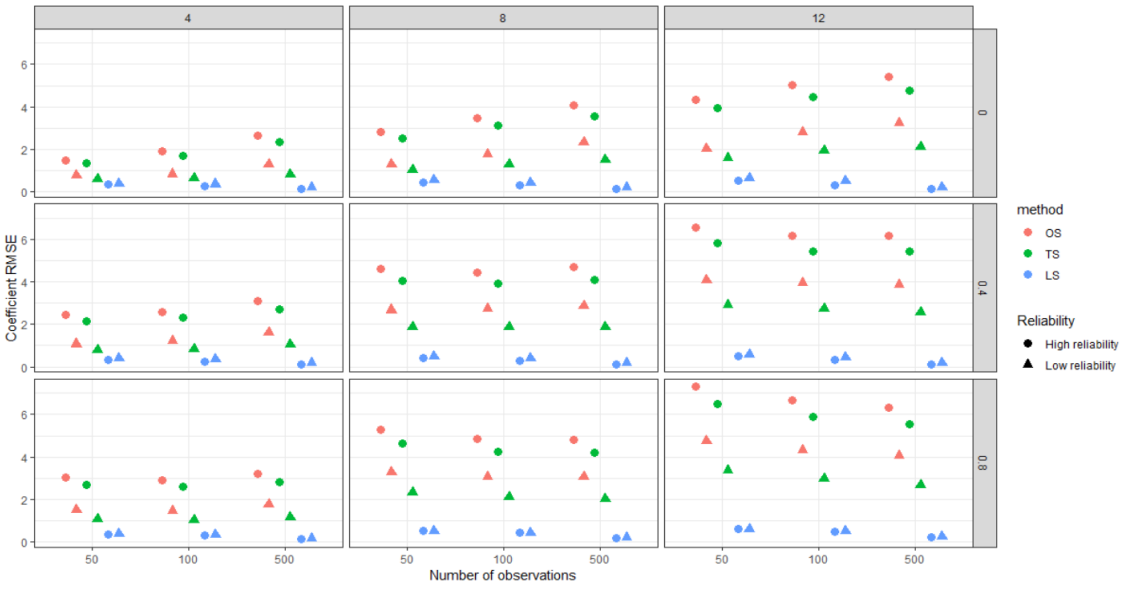


Figure 3.3: The effects of design factors coefficient RMSE. The boxes represent the design factors number of true nonzero regression coefficients (ℓ = 4, 8, 12) and covariance of latent variables (cov = 0, 0.4, 0.8)

3.4 Prediction RMSE

The prediction RMSE is based on a 10-fold cross validation for each of the 100 replications for each combination of design factors. Table 3.4 shows the results of the repeated-measures Anova. The most important effects are the interaction effect of number of observations with the methods, and the interaction effect of number of true nonzero regression coefficients and the methods.

When the number of observation increases, the prediction error decrease for all three methods (Figure 3.4, Table B.5, B.6). However, this decreasing effect is smaller for the ordinary least-squares method compared to the observed-score standardization method. The effect is the same for the true-score standardization and observed-score standardization methods. From figure 3.4, it is visible most effects are gone when the number of observations is large. When the number of observations is increased to 500, the prediction RMSE is almost equal for all three LASSO methods.

	Sum Sq	Mean Sq	NumDF	DenDF	F value	Pr(>F)	η_p^2
Reliability	2.75	2.75	1	46	4.26	0.0446	0.004
cov	1.50	0.75	2	46	1.16	0.3220	0.002
ℓ	41.42	20.71	2	46	32.10	1.875e-09	0.057
N	254.55	127.27	2	46	197.26	< 2.2e-16	0.272
method	457.47	228.74	2	16130	354.52	< 2.2e-16	0.042
Reliability:method	1.35	0.68	2	16130	1.05	0.3501	0.000
cov:method	0.58	0.14	4	16130	0.22	0.9253	0.000
l:method	10.99	2.75	4	16130	4.26	0.0019	0.001
N:method	269.66	67.42	4	16130	104.49	< 2.2e-16	0.025

Table 3.4: Results of repeated-measures Anova for prediction RMSE

cov = covariance, ℓ = number of true nonzero regression coefficients, N = number of observations, Sum Sq = Sum of Squares, Mean Sq = Mean Squares, NumDF = degrees of freedom for numerator, DenDF = degrees of freedom for denominator, η_p^2 = partial eta squared.

3.5 Empirical data

With respect to the empirical data, we compare the methods on their choice of tests, the size of the coefficients, and the prediction RMSE. The coefficients and the prediction RMSE were computed through 10-fold cross validation. The mean prediction error per method can be seen in table 3.5.

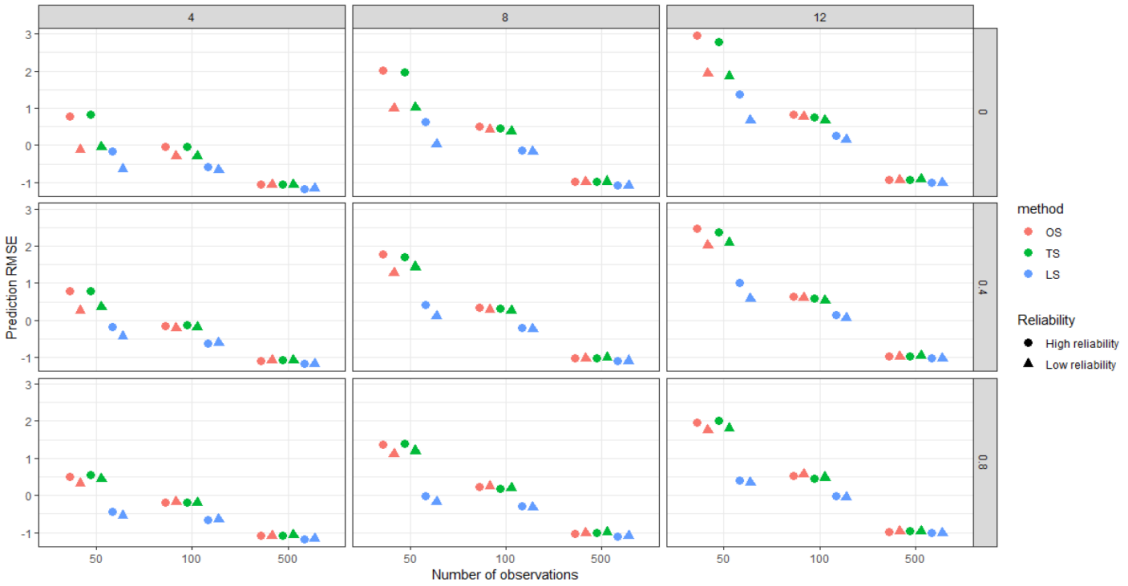


Figure 3.4: Effects of design factors on prediction RMSE. The boxes represent the design factors number of true nonzero regression coefficients (ℓ = 4, 8, 12) and covariance of latent variables (cov = 0, 0.4, 0.8)

	RMSE
OS_LASSO	13.55
TS_LASSO	11.79
LS_LASSO	13.38

Table 3.5: Prediction RMSE for empirical data, predicting general mental health OS_LASSO = observed-score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO, RMSE = prediction RMSE between actual dependent variable and predicted dependent variable.

The smallest prediction error is given by the true-score standardization LASSO. The observed-score standardization and the least-squares transformation methods show quite similar larger prediction RMSE values. Table 3.6 shows the mean cross validated coefficients for the tests for all three methods. The observed-score standardization LASSO and the true-score standardization LASSO select the same number of tests. However, the ordinary least-squares transformation LASSO selects one test fewer, constraining *Negative Self Esteem* to zero coefficient. Regarding table 3.6, the signs of the coefficients may seem counter intuitive, why would *dysphoria* have a positive effect on general mental health, while a high life satisfaction has a negative effect on general mental health? In this survey, general mental health was measured on a four point scale, with 1 indicating that for that particular question, you feel better than usual, and 4 indicating that you feel much less than usual. A negative coefficient thus means that it will relate to feeling better than usual. A high score on the test measuring *Euphoria* and *Life Satisfaction* thus increases your general mental health. Comparing the three methods with respect to the size of the coefficients, the true-score standardization method is the most conservative and the ordinary least-squares LASSO gives the largest effects to the individual tests.

Combining the results of the empirical data and the simulation, we could look at the results of the simulation when the number of observations is high and the reliabilities of the tests are high. In these simulation scenarios, we see the prediction RMSE show small differences for the three methods, in favour of the least-squares transformation method. In the empirical data, the best prediction RMSE is given by the true-score standardization LASSO.

The empirical data shows us that true-score standardization LASSO is the best pick, when prediction is the main focus. The methods do not differ much in the variables they select for the regression (Table 3.6). However, the size of the regression coefficients differs between the methods. The coefficients for ordinary least-squares transformation LASSO are larger in size. There is less shrinkage for the ordinary least-squares transformation method. The ordinary least-squares transformation method would give coefficients with less bias in theory, but this concurs with more variance, and more prediction error than the true-score standardization method.

Coefficient	OS_LASSO	TS_LASSO	LS_LASSO
Intercept	19.34	20.53	18.68
Disinhibition	0	0	0
Dysphoria	0.44 (0.07)	0.31 (0.06)	0.64 (0.15)
Euphoria	-0.18 (0.05)	-0.07 (0.05)	-0.14 (0.20)
Impulsivity	0	0	0
Life Satisfaction	-0.41 (0.05)	-0.40 (0.03)	-0.59 (0.14)
Need For Change	0	0	0
Negative Self Esteem	0.26 (0.04)	0.21 (0.03)	0
Neuroticism	0.71 (0.06)	0.61 (0.05)	1.05 (0.05)
Positive Self Esteem	0	0	0

Table 3.6: Coefficients for predicting general mental health in the empirical data for each method

OS_LASSO = observed-score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO.

Chapter 4

Conclusions and discussion

The focus of statistical modeling in psychology is either on explanation or on prediction. We studied a statistical model with a focus on selection and prediction, and looked at how accurately it can explain the effects of the tests under different forms of standardization, to combine a predictive focus with a more explanatory focus. We tested these methods with respect to the truthfulness of variable selection, coefficient size, and prediction accuracy under different scenarios of design factors. We also examined the interaction of the design factors with the three methods to determine which design factors would have most influence on the accuracy measures.

We interpret the truthfulness of the variable selection through the sensitivity and the specificity. When the number of observations is large, the ordinary least-squares transformation method reaches an optimal variable selection by having a high sensitivity and high specificity. However, the observed-score standardization and true-score standardization methods show a different relationship between sensitivity and specificity. We showed that the number of observations, the covariance, and the number of true nonzero regression coefficients have a different effect on the specificity and the sensitivity when looking at the true-score standardization and observed-score standardization methods compared to the ordinary least-squares transformation method.

The coefficient RMSE shows whether or not the size of the coefficients for the methods capture the true effect of the test. The ordinary least-squares transformation method shows the smallest Euclidean distance, and the observed-score standardization LASSO shows the largest Euclidean distance between the true regression coefficients and the estimated regression coefficients. The design factors with a large interaction effects on the methods are the number of true regression coefficients, reliability of the tests, and the covariance between the latent variables.

When you want to select candidates who have the skills required for the job in the shortest amount of time, prediction is also important, next to the truthfulness of variable selection and test importance. With regards to the prediction RMSE, the methods show small differences in prediction over each of the scenarios. The most important design factor for prediction is the number of observations. When the number of observations is 500 the decrease of prediction RMSE is larger for true-score standardization and observed-score standardization LASSO when compared to the ordinary least-squares transformation LASSO.

Both true-score standardization and ordinary least-squares transformation LASSO outperform the observed-score standardization method in the simulation and the empirical data analysis. Which alternative method of standardization a researcher should use thus depends on the question a researcher/selector may ask themselves. Does the focus of the research lie on variable selection, coefficient truthfulness, or prediction accuracy? Based on the nature of the question, and the nature of the study design, a different LASSO may be best suited. In this study, we showed that prediction and selection can improve by transforming the predictors in other ways. These alternative methods can be used in practice to improve understanding of effects of skills, and prediction of future job performance. A better representation of future job performance can help inform job selectors which applicant is better suited for the job.

4.1 Limitations and further explorations

In this simulation study, we made some choices in the design that result in the data satisfying some conditions unlikely to hold in real-life situations. For example, we measure the items and tests on a numeric scale, making it easier to follow the single-factor model. In real-life, most psychometric test items will be measured on an ordinal scale. Secondly, the covariance between the latent variables was constant between all tests. In real life, some tests will more closely be related to each other than other tests. Thirdly, the true regression coefficient in this study were either 1 or 0, to simplify their interpretation, and to not let that interpretation interact with other design factors, such as number of true nonzero regression coefficients or reliabilities of the tests. In real life, each test will have a different effect and some may even have a negative effect on the outcome measure. Fourthly, the variance of the residuals in the true regression model to calculate the dependent variable is constant for each scenario of design factors. This choice was for consistency, the residual in a linear regression does not take into account how many tests are present in prediction and how these tests interact. However, this does result in different proportions of variation of the dependent variable accounted for by the independent variables. It would be interesting to see what happens when the residual variance is created, such that the proportion variance accounted for is constant for each scenario.

In the assessment of the design factors and methods, we transform the outcome measures to be able to fit the repeated-measures Anova and calculate the partial eta-squared. There are other methods of assessing the effects of the design factors, for example, generalized linear mixed models. In this study, the predictor variables were categorical, which made interpretation dependent on the arbitrary order we assigned. The interpretation was thus more difficult. Future research could expand by making numerical predictors, and assessing the effects with a generalized linear mixed model.

Furthermore, we measured the true-score variance by the reliability. However, reliability measures, are known to be biased an not give accurate representations (Oosterwijk, 2016). When we designed the data to have low reliability and a small number of observations, the reliability showed the largest bias, and was sometimes estimated to be negative for certain tests. For these test, true-score standardization would not be possible, because you would divide by a negative number. We decided

to solve this by using the absolute value of that reliability as the estimated reliability in the standardization. It would be helpful for true-score standardization to find a computation of reliability that is less biased, or find another method to calculate the true-score variances.

Lastly, for each LASSO regression, we set the penalty parameter based on the 'one standard-error rule', to better compare each method. As discussed, we see a negative relationship between the sensitivity and the specificity when the number of observations increases for observed-score and true-score standardization. This appears to be related to the size of the penalty parameter. Further research could explore the influence of the penalty parameter on this trade-off between sensitivity and specificity and which penalty parameter results in the 'best' relationship between sensitivity and specificity.

References

Adams, E. W., Fagot, R. F., & Robinson, R. E. (1965). A theory of appropriate statistics. *Psychometrika*, *30*(2), 99–127.

Bleidorn, W., & Hopwood, C. J. (2018). Using machine learning to advance personality assessment and theory. *Personality and Social Psychology Review*, 190-203.

Borsboom, D., Romeijn, J.-W., & Wicherts, J. M. (2008). Measurement invariance versus selection invariance: Is fair selection possible? *Psychological Methods*, *13*(2), 75.

Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics*, *37*(4), 373–384.

Breiman, L. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, *16*(3), 199–231.

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *psychometrika*, *16*(3), 297–334.

Guttman, L. (1945). A basis for analyzing test-retest reliability. *Psychometrika*, *10*(4), 255–282.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: prediction, inference and data mining*. Springer-Verlag, New York.

Hastie, T., Tibshirani, R., & Wainwright, M. (2015). *Statistical learning with sparsity: the lasso and generalizations*. CRC press.

McDonald, R. P. (2013). *Test theory: A unified treatment*. Psychology Press.

Meinshausen, N., Bühlmann, P., et al. (2006). High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, *34*(3), 1436–1462.

Mellenbergh, G. J. (2011). *A conceptual introduction to psychometrics: Development, analysis and application of psychological and educational tests*. Eleven International The Hague, Netherlands.

Oosterwijk, P. (2016). *Statistical properties and practical use of classical test-score reliability methods*. Ridderprint.

Parikh, R., Mathai, A., Parikh, S., Sekhar, G. C., & Thomas, R. (2008). Understanding and using sensitivity, specificity and predictive values. *Indian journal of ophthalmology*, *56*(1), 45-50.

Putka, D. J., Beatty, A. S., & Reeder, M. C. (2017). Modern prediction methods: New perspectives on a common problem. *Organizational Research Methods*, 689-732.

Richardson, J. T. (2011). Eta squared and partial eta squared as measures of effect size in educational research. *Educational Research Review*, *6*(2), 135–147.

Shmueli, G. (2010). To explain or to predict? *Statistical science*, *25*(3), 289-310.

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

van Loon, W. (2016). *Performance of lasso-penalized classifiers in high-dimensional datasets* (Masters Thesis, Leiden University). Retrieved from `https://openaccess.leidenuniv.nl/handle/1887/39093`

Waljee, A. K., Higgins, P. D., & Singal, A. G. (2014). A primer on predictive models. *Clinical and translational gastroenterology*, *5*(1), e44.

Xiong, S. (2010). Some notes on the nonnegative garrote. *Technometrics*, *52*(3), 349–361.

Yarkoni, T., & Westfall, J. (2017). Choosing prediction over explanation in psychology: Lessons from machine learning. *Perspectives on Psychological Science*, *12*(6), 1100–1122.

Yuan, M., & Lin, Y. (2007). On the non-negative garrotte estimator. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *69*(2), 143–161.

Zhang, K., Yin, F., & Xiong, S. (2014). Comparisons of penalized least squares methods by simulations. *arXiv preprint arXiv:1405.1796*.

Zhao, P., & Yu, B. (2006). On model selection consistency of lasso. *Journal of Machine learning research*, *7*(Nov), 2541–2563.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American statistical association*, *101*(476), 1418–1429.

Appendix A

Factor loadings based on reliability

We first simulate the latent variable Θ out of a multivariate normal distribution

$$f(\Theta) : \Theta_{i1}, \Theta_{i2}, ..., \Theta_{ip} \sim N(\mu, \Sigma). \tag{A.1}$$

where $\mu = 0$ and Σ is a $p * p$ matrix, where the diagonals represent $\sigma^2_{x_j} = 1$ and the covariance is represented in the off-diagonals. For each person i with item k on test j

$$j = 1, ..., p \tag{A.2}$$
$$k = 1, ..., r \tag{A.3}$$
$$i = 1, ..., N \tag{A.4}$$

The observed scores x_{ijk} are build through the single-factor model.

$$x_{ijk} = \mu_{jk} + \lambda_{jk} * \Theta_{ij} + \epsilon_{ijk}, \forall ijk \tag{A.5}$$

The observed item scores can be distinguished into an true score (τ) and an error score (ϵ).

The reliability of an item is for a given test j

$$\rho_j = \frac{VAR(\tau_{jk})}{VAR(\tau_{jk}) + VAR(\epsilon_{jk})} \tag{A.6}$$

Rewriting this gives:

$$\begin{aligned} VAR(\tau_{jk}) &= VAR(\lambda_{jk} * \Theta_j) && \text{(A.7)} \\ &= \lambda^2_{jk} * VAR(\Theta_j) && \text{(A.8)} \\ &= \lambda^2_{jk} && \text{(A.9)} \end{aligned}$$

We will sum over items, so we can use ρ_j as an input for the whole test variance.

$$\begin{aligned} VAR(\epsilon_j) &= VAR(\epsilon_{j1} + \epsilon_{j2} + ... + \epsilon_{jr}) && \text{(A.10)} \\ &= \sigma^2_{\epsilon_{j1}} + \sigma^2_{\epsilon_{j2}} + ... + \sigma^2_{\epsilon_{jr}} && \text{(A.11)} \\ &= r * \sigma^2_{\epsilon_j} && \text{(A.12)} \end{aligned}$$

Because we constrain $x_{ijk} \sim N(0, 1)$, we can assume

$$\lambda^2_{jk} = 1 - \sigma^2_{\epsilon_{jk}} \tag{A.13}$$

and

$$\sigma^2_{\epsilon_{jk}} = 1 - \lambda^2_{jk} \tag{A.14}$$

We simulate parallel items, making the variance of the test score

$$\begin{aligned} VAR(x_j) &= VAR(\sum_{k=1}^{r} x_{jk}) && \text{(A.15)} \\ &= VAR[\sum_{k=1}^{r}(\lambda_{jk}\Theta j)] + \sum_{k=1}^{r} VAR(\epsilon_{jk}) && \text{(A.16)} \\ & && \text{(A.17)} \end{aligned}$$

and the variance of the true score part

$$\begin{aligned} VAR(\tau_j) &= E[(\sum_{k=1}^{r}(\lambda_{jk}\Theta_j))^2] && \text{(A.18)} \\ &= \sum_{k=1}^{r}\sum_{k'=1}^{r} E(\lambda_{jk}\Theta_j^2)(\lambda_{jk'}\Theta_j^2) && \text{(A.19)} \\ &= \sum_{k=1}^{r}\sum_{k'=1}^{r} \lambda_{jk}\lambda_{jk'} \sum_{k=1}^{r} \Theta_j^2 && \text{(A.20)} \\ &= (\lambda_{j1} + \lambda_{j2} + ... + \lambda_{jr})^2 && \text{(A.21)} \\ &= r^2\lambda_j^2 && \text{(A.22)} \end{aligned}$$

Substituting this in the reliability function, the reliability will be:

$$\begin{aligned} \rho_j &= \frac{r^2\lambda_j^2}{r^2\lambda_j^2 + r(1-\lambda_j^2)} && \text{(A.23)} \\ &= \frac{r\lambda_j^2}{r\lambda_j^2 + (1-\lambda_j^2)} && \text{(A.24)} \\ &= \frac{r\lambda_j^2}{1 + (r-1)\lambda_j^2} && \text{(A.25)} \end{aligned}$$

What will be the formula for λ such that we can add ρ_j as an input in the simulation procedure.

$$\begin{aligned} \rho_j &= \frac{r\lambda_j^2}{1 + (r-1)\lambda_j^2} && \text{(A.26)} \\ 1 + (r-1)\lambda_j^2 &= \frac{r\lambda_j^2}{\rho_j} && \text{(A.27)} \\ (r-1)\lambda_j^2 &= \frac{r\lambda_j^2}{\rho_j} - 1 && \text{(A.28)} \\ (r-1)\lambda_j^2 * \rho_j &= r\lambda_j^2 - \rho_j && \text{(A.29)} \\ (r-1)\lambda_j^2 * \rho_j - r\lambda_j^2 &= -\rho_j && \text{(A.30)} \\ \lambda_j^2 * ((r-1)\rho_j - r) &= -\rho_j && \text{(A.31)} \\ \lambda_j^2 &= \frac{-\rho_j}{(r-1)\rho_j - r} && \text{(A.32)} \\ \lambda_j &= \sqrt{\frac{\rho_j}{r - (r-1)\rho_j}} && \text{(A.33)} \end{aligned}$$

Appendix B

Tables of results

			OS_LASSO		TS_LASSO		LS_LASSO	
Cov	ℓ	N	sen	spe	sen	spe	sen	spe
0	4	50	0.49	0.93	0.48	0.93	0.62	0.93
		100	0.87	0.95	0.88	0.94	0.87	0.98
		500	1.00	0.99	1.00	0.99	1.00	1.00
	8	50	0.65	0.83	0.66	0.82	0.68	0.88
		100	0.96	0.82	0.97	0.81	0.93	0.94
		500	1.00	0.96	1.00	0.95	1.00	1.00
	12	50	0.76	0.75	0.79	0.72	0.71	0.84
		100	0.98	0.70	0.98	0.70	0.94	0.92
		500	1.00	0.90	1.00	0.89	1.00	1.00
0.4	4	50	0.81	0.87	0.81	0.87	0.66	0.93
		100	0.94	0.88	0.95	0.87	0.85	0.96
		500	1.00	0.94	1.00	0.93	1.00	1.00
	8	50	0.86	0.75	0.87	0.76	0.68	0.89
		100	0.99	0.75	0.99	0.75	0.87	0.95
		500	1.00	0.82	1.00	0.82	1.00	1.00
	12	50	0.90	0.66	0.90	0.66	0.69	0.85
		100	0.99	0.67	0.99	0.67	0.88	0.94
		500	1.00	0.72	1.00	0.72	1.00	1.00
0.8	4	50	0.76	0.83	0.74	0.83	0.51	0.92
		100	0.89	0.84	0.89	0.81	0.68	0.96
		500	1.00	0.88	1.00	0.85	0.98	1.00
	8	50	0.80	0.72	0.81	0.72	0.50	0.89
		100	0.94	0.71	0.94	0.69	0.70	0.94
		500	1.00	0.72	1.00	0.72	0.97	1.00
	12	50	0.83	0.64	0.85	0.63	0.54	0.85
		100	0.95	0.62	0.96	0.60	0.72	0.92
		500	1.00	0.67	1.00	0.66	0.98	1.00

Table B.1: Correctness of model for each method when reliabilities are high.

Cov = covariance, ℓ = number of true nonzero regression coefficient, N = number of observations, OS_LASSO = observed score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO, sen = sensitivity, spe = specificity.

			OS_LASSO		TS_LASSO		LS_LASSO	
Cov	ℓ	N	sen	spe	sen	spe	sen	spe
0	4	50	0.23	0.96	0.17	0.96	0.37	0.93
		100	0.44	0.97	0.42	0.96	0.54	0.97
		500	1.00	0.99	1.00	0.99	0.99	1.00
	8	50	0.29	0.92	0.28	0.91	0.40	0.90
		100	0.67	0.88	0.68	0.88	0.71	0.94
		500	1.00	0.95	1.00	0.94	0.99	1.00
	12	50	0.39	0.88	0.42	0.85	0.45	0.87
		100	0.81	0.80	0.83	0.76	0.76	0.89
		500	1.00	0.88	1.00	0.88	1.00	0.99
0.4	4	50	0.38	0.95	0.38	0.95	0.42	0.94
		100	0.67	0.94	0.70	0.93	0.61	0.97
		500	1.00	0.96	1.00	0.96	0.98	1.00
	8	50	0.64	0.82	0.65	0.82	0.51	0.89
		100	0.90	0.80	0.90	0.78	0.73	0.94
		500	1.00	0.87	1.00	0.86	0.99	1.00
	12	50	0.75	0.74	0.76	0.73	0.56	0.83
		100	0.94	0.72	0.94	0.70	0.75	0.91
		500	1.00	0.78	1.00	0.77	0.99	1.00
0.8	4	50	0.48	0.90	0.51	0.89	0.40	0.92
		100	0.73	0.91	0.75	0.88	0.56	0.96
		500	1.00	0.93	1.00	0.91	0.97	1.00
	8	50	0.68	0.78	0.70	0.75	0.46	0.86
		100	0.89	0.78	0.90	0.75	0.65	0.93
		500	1.00	0.80	1.00	0.77	0.97	1.00
	12	50	0.75	0.70	0.78	0.67	0.50	0.83
		100	0.93	0.69	0.93	0.67	0.68	0.91
		500	1.00	0.73	1.00	0.70	0.98	1.00

Table B.2: Correctness of model for each method when reliabilities are low.

Cov = covariance, ℓ = number of true nonzero coefficients, N = number of observation, OS_LASSO = observed score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO, sen = sensitivity, spe = specificity.

Cov	ℓ	N	OS_LASSO	TS_LASSO	LS_LASSO
0	4	50	1.48	1.33	0.33
		100	1.90	1.68	0.25
		500	2.66	2.36	0.14
	8	50	2.81	2.51	0.44
		100	3.47	3.10	0.29
		500	4.05	3.55	0.15
	12	50	4.34	3.93	0.51
		100	5.02	4.45	0.32
		500	5.42	4.77	0.15
0.4	4	50	2.44	2.15	0.32
		100	2.60	2.31	0.24
		500	3.08	2.72	0.11
	8	50	4.59	4.06	0.42
		100	4.45	3.90	0.30
		500	4.69	4.11	0.12
	12	50	6.56	5.83	0.50
		100	6.18	5.45	0.35
		500	6.16	5.42	0.12
0.8	4	50	3.02	2.67	0.37
		100	2.90	2.59	0.31
		500	3.19	2.80	0.14
	8	30	5.28	4.64	0.51
		100	4.85	4.25	0.42
		500	4.82	4.20	0.19
	12	50	7.31	6.47	0.60
		100	6.67	5.87	0.49
		500	6.33	5.55	0.22

Table B.3: coefficient RMSE for each method when reliabilities are high.

Cov = covariance, ℓ = number of true nonzero regression coefficients, N = number of observations, OS_LASSO = observed score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO.

Cov	ℓ	N	OS_LASSO	TS_LASSO	LS_LASSO
0	4	50	0.77	0.61	0.40
		100	0.83	0.65	0.36
		500	1.30	0.82	0.21
	8	50	1.31	1.02	0.54
		100	1.77	1.27	0.44
		500	2.34	1.51	0.22
	12	50	2.02	1.58	0.63
		100	2.79	1.95	0.49
		500	3.22	2.11	0.22
0.4	4	50	1.07	0.79	0.38
		100	1.21	0.85	0.34
		500	1.61	1.04	0.17
	8	50	2.67	1.86	0.50
		100	2.73	1.88	0.40
		500	2.87	1.89	0.18
	12	50	4.10	2.90	0.58
		100	3.95	2.73	0.46
		500	3.85	2.55	0.18
0.8	4	50	1.48	1.06	0.39
		100	1.46	1.03	0.35
		500	1.76	1.14	0.18
	8	50	3.29	2.33	0.52
		100	3.08	2.12	0.44
		500	3.05	2.01	0.21
	12	50	4.76	3.38	0.61
		100	4.32	2.98	0.51
		100	4.04	2.68	0.23

Table B.4: Coefficient RMSE for each method when reliabilities are low.

Cov = covariance, ℓ = number of true nonzero regression coefficients, N = number of observations, OS_LASSO = observed score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO.

Cov	ℓ	N	OS_LASSO	TS_LASSO	LS_LASSO
0	4	50	29.37	29.47	27.68
		100	27.91	27.92	26.90
		500	26.04	26.05	25.85
	8	50	31.63	31.53	29.13
		100	28.90	28.79	27.74
		500	26.18	26.19	26.00
	12	50	33.34	33.01	30.48
		100	29.46	29.36	28.43
		500	26.28	26.29	26.13
0.4	4	50	29.42	29.41	27.66
		100	27.70	27.74	26.84
		500	26.00	26.02	25.84
	8	50	31.19	31.08	28.71
		100	28.58	28.53	27.58
		500	26.14	26.14	25.98
	12	50	32.49	32.30	29.83
		100	29.12	29.06	28.22
		500	26.22	26.23	26.13
0.8	4	50	28.90	28.96	27.19
		100	27.63	27.63	26.78
		500	25.99	26.02	25.85
	8	50	30.45	30.49	27.96
		100	28.39	28.32	27.44
		500	26.11	26.14	25.98
	12	50	31.56	31.63	28.71
		100	28.92	28.80	27.94
		500	26.19	26.22	26.13

Table B.5: Prediction RMSE for each method when reliabilities are high.

Cov = covariance, ℓ = true nonzero regression coefficients, N = number of observations, OS_LASSO = observed score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO.

Cov	ℓ	N	OS_LASSO	TS_LASSO	LS_LASSO
0	4	50	27.76	27.88	26.80
		100	27.44	27.45	26.76
		500	26.02	26.06	25.86
	8	50	29.79	29.81	28.01
		100	28.73	28.64	27.65
		500	26.16	26.20	26.01
	12	50	31.50	31.36	29.17
		100	29.36	29.19	28.24
		500	26.25	26.29	26.14
0.4	4	50	28.45	28.63	27.19
		100	27.60	27.64	26.86
		500	26.00	26.03	25.85
	8	50	30.31	30.59	28.16
		100	28.51	28.44	27.53
		500	26.12	26.16	25.99
	12	50	31.63	31.76	29.03
		100	29.06	28.93	28.08
		500	26.21	26.25	26.12
0.8	4	50	28.58	28.78	27.00
		100	27.65	27.63	26.80
		500	26.00	26.02	25.86
	8	50	30.02	30.16	27.68
		100	28.44	28.33	27.37
		500	26.12	26.16	25.99
	12	50	31.17	31.26	28.60
		100	28.99	28.85	27.87
		500	26.20	26.23	26.11

Table B.6: Prediction RMSE for each method when reliabilities are low.

Cov = covariance, ℓ = number of true nonzero regression coefficients, N = number of observations, OS_LASSO = observed score standardization LASSO, TS_LASSO = true-score standardization LASSO, LS_LASSO = ordinary least-squares transformation LASSO

Appendix C

R-code

```
simdata <- function(N, rho, l, cov, nitems = 25, ntests = 20){
  #args:
  #N is sample size
  #rho is pattern of reliability
  #l is size of true regression model
  #cov is covariance between test
  #nitems is the number of items in each test
  #ntests is the number of tests in the test battery


  #initial messages and operations
  if(length(unlist(rho)) != ntests){
    print("Every test must have a reliability")
    break
  }

  require(mvtnorm)


  #create variance/covariance matrix for latent variables
  varcov <- matrix(cov, nrow=ntests, ncol=ntests)
  diag(varcov) <- 1 #variance of observed item score

  #create latent trait for each test for each person
  mus <- rep(0, ntests)
  thetak <- rmvnorm(N, mean = mus, sigma=varcov, method="chol")

  #create test loadings (lambda) based on the reliability
  lambdak2 <- unlist(rho)/(nitems - (nitems-1)*unlist(rho))
  lambdajk2 <- matrix(rep(lambdak2, nitems), ncol = nitems)

  #create error vars based on the reliability and the loadings
  varepsilon <- (1 - lambdajk2)

  #create  item scores
```

```
  xijk <- lapply(1:ntests, function(x){
    sapply(1:nitems, function(y){
      t(2 + sqrt(lambdajk2[x, y]) %*% thetak[,x] +
          rnorm(N, 0, varepsilon[x, y]))
    })
  })

  #add the observed-scores to gain a testscore
  X_testdata <- sapply(xijk, rowSums)


  #create the vector for coefficients
  beta <- matrix(c(rep(1, l), rep(0, ntests-l)), ncol = 1)

  #create the output variable
  Y <- X_testdata %*% beta + rnorm(N, 0, nitems)

  return(list(Xtestdata = data.frame(X_testdata),
              Ytestdata = data.frame(Y), rawdata = xijk, betas = beta))

}


#function for observed-score standardization
OS_LASSO <- function(data, output){
  #args:
  #data is the dataset with testscores
  #output is the vector of output values

  #initial operations
  require(glmnet)

  #Calculate observed score variance
  obs_vars <- apply(data, 2, var)

  #standardize by observed score variance
  stand_obsvars <- sapply(1:ncol(data), function(x)
    {(data[,x] - mean(data[, x]))/sqrt(obs_vars[x])})

  cv.model <- cv.glmnet(stand_obsvars, unlist(output),
                        family = "gaussian", standardize = FALSE,
                        alpha = 1, nlambda = 100, type.measure = "mse",
                        grouped = FALSE, keep = TRUE)

  return(cv.model)
}
```

```
#function for true-score standardization
TS_LASSO <- function(rawdata, data, output){

  #iniial operations
  require(Lambda4)
  require(glmnet)

  #calculate the true-score variance
  obs_vars <- apply(data, 2, var)

  rel <- sapply(1:ncol(data), function(x) {
    guttman(as.data.frame(rawdata[x]), missing = "complete",
            standardize = TRUE)$Lambda2
  })

  if(any(rel < 0)){
    rel <- abs(rel)
  }

  true_vars <- obs_vars * rel

  #Standardize by true score variance
  stand_truevars <- sapply(1:ncol(data), function(x){
    (data[,x] - mean(data[,x]))/sqrt(true_vars[x])})

  cv.model <- cv.glmnet(stand_truevars, unlist(output),
                        family = "gaussian", standardize = FALSE,
                        alpha = 1, nlambda = 100, type.measure = "mse",
                        grouped = FALSE, keep = TRUE)

  return(cv.model)
}



#function for least-squares standardization
LS_LASSO <- function(data, output){

  #initial operations
  require(glmnet)

  #do a linear regression with the data and extract the beta's
  data_lm <- data.frame(data, "Y"=output)
```

```
  lm_model <- lm(Y ~ ., data = data_lm)
  coefs_lm <- lm_model$coefficients[-1]

  #standardize by the linear model coefficients
  stand_lmvars <- sapply(1:ncol(data), function(x)
    {data[,x]*abs(coefs_lm[x])})

  #perform the lasso
  cv.model <- cv.glmnet(stand_lmvars, unlist(output),
                        family = "gaussian", standardize = FALSE,
                        alpha = 1, nlambda = 100, type.measure = "mse",
                        grouped = FALSE, keep = TRUE)

  return(cv.model)

}

#Empirical data

#import dataset
emp_data <- read.spss(file="TOTAAL.SAV", use.value.labels = FALSE,
                      to.data.frame = TRUE)


# 1991
# test.items
Life.satisfaction91<- c("SS281", "SS282", "SS283","SS284","SS285")
Dysphoria91<- c("SS236","SS238","SS239","SS240", "SS242", "SS243",
               "SS244","SS246","SS247","SS249","SS250","SS254",
               "SS255","SS257","SS258","SS259","SS261","SS262",
               "SS264","SS265","SS267","SS268" )
Euphoria91<-  c("SS237","SS241","SS245","SS248","SS251","SS252",
               "SS253","SS256","SS260","SS263","SS266","SS269")
Positive.self.esteem91<- c("SS224","SS225","SS226")
Negative.self.esteem91<- c("SS227","SS228","SS229","SS230","SS231",
                           "SS232","SS233","SS234")
Neuroticism91<-  c("SS196","SS197","SS198", "SS200","SS207",
                   "SS214","SS222")
Disinhibition91<-  c("SS201", "SS205","SS208","SS212","SS216","SS220")
Need.for.change91<-  c("SS202","SS206","SS209","SS213","SS217","SS221")
Impulsivity91<-  c("SS204","SS211","SS215","SS219","SS223","SS235")
General.psychological.health91<-  c("SS270","SS271","SS272","SS273",
                                    "SS274","SS275","SS276","SS277",
                                    "SS278","SS27", "SS280")
Work.wish91<-c( "MM384")
Workless91<-c("MM386")
```

```
#create te dataset
emp_data_91 <- emp_data[, c(Disinhibition91, Dysphoria91, Euphoria91,
                            General.psychological.health91,
                            Impulsivity91, Life.satisfaction91,
                            Need.for.change91, Negative.self.esteem91,
                            Neuroticism91, Positive.self.esteem91)]

emp_data_91_complete <- emp_data_91[ complete.cases(emp_data_91), ]

#create list of datasets
raw_testdata <- list(Disinhibition =
                       emp_data_91_complete[ , Disinhibition91],
                     Dysphoria = emp_data_91_complete[ , Dysphoria91],
                     Euphoria = emp_data_91_complete[, Euphoria91],
                   Psychhealth =
                   emp_data_91_complete[, General.psychological.health91],
                     Impulsivity = emp_data_91_complete[, Impulsivity91],
                     Life.satisfaction =
                       emp_data_91_complete[, Life.satisfaction91],
                     Need.for.change =
                       emp_data_91_complete[, Need.for.change91],
                     Negative.self.esteem =
                       emp_data_91_complete[,  Negative.self.esteem91],
                     Neuroticism =
                       emp_data_91_complete[,  Neuroticism91],
                     Positive.self.esteem =
                       emp_data_91_complete[, Positive.self.esteem91]
)

#create dataset of test scores
testdata <- sapply(raw_testdata, function(x) rowSums(x, na.rm = TRUE))



set.seed(12345)
K <- 10
index <- rep(1:K, floor(nrow(testdata)/K)+1)[1:nrow(testdata)]
fold.index <- sample(index)

pred.error <- sapply(1:K, function(k){
  training <- testdata[fold.index!=k, ]
  validation <- testdata[fold.index==k, ]

  fit.os <- OS_LASSO(data = training[, c(1:3, 5:10)],
                     output = training[, 4])
  fit.ts <- TS_LASSO(data = training[, c(1:3, 5:10)],
                     rawdata = raw_testdata, output = training[, 4])
```

```
  fit.ls <- LS_LASSO(data = training[, c(1:3, 5:10)],
                     output = training[, 4])

  y.pred.os <- glmnet::predict.cv.glmnet(fit.os,
               newx = validation[, c(1:3, 5:10)], type = "response",
               s = "lambda.1se")
  y.pred.ts <- glmnet::predict.cv.glmnet(fit.ts,
               newx = validation[, c(1:3, 5:10)], type = "response",
               s = "lambda.1se")
  y.pred.ls <- glmnet::predict.cv.glmnet(fit.ls,
               newx = validation[, c(1:3, 5:10)], type = "response",
               s = "lambda.1se")

  pred.error.os <- sqrt(sum((validation[, 4] - y.pred.os)^2)
                        /nrow(validation))
  pred.error.ts <- sqrt(sum((validation[, 4] - y.pred.ts)^2)
                        /nrow(validation))
  pred.error.ls <- sqrt(sum((validation[, 4] - y.pred.ls)^2)
                        /nrow(validation))

  #also want to know the beta's
  beta_os <- coef(fit.os, s = "lambda.1se")
  beta_ts <- coef(fit.ts, s = "lambda.1se")
  beta_ls <- coef(fit.ls, s = "lambda.1se")

  return(c(pred.error.os, pred.error.ts, pred.error.sl, pred.cor.os,
           pred.cor.ts, pred.cor.sl, as.numeric(beta_os),
           as.numeric(beta_ts), as.numeric(beta_ls)))
})
```