Select relevant moderators in meta-regression using Bayesian penalization

Caspar J. Van Lissa[1,2] & Sara van Erp[1]

[1] Utrecht University, dept. Methodology & Statistics

[2] Open Science Community Utrecht

Author Note

Add complete departmental affiliations for each author here. Each new line herein must be indented, like this line.

Enter author note here.

The authors made the following contributions. Caspar J. Van Lissa: Conceptualization, Writing - Original Draft Preparation, Programming front-end; Sara van Erp: Writing - Contributions and feedback, Programming back-end.

Correspondence concerning this article should be addressed to Caspar J. Van Lissa, Padualaan 14, 3584CH Utrecht, The Netherlands. E-mail: c.j.vanlissa@uu.nl

Abstract

One or two sentences providing a **basic introduction** to the field, comprehensible to a scientist in any discipline.

Two to three sentences of **more detailed background**, comprehensible to scientists in related disciplines.

One sentence clearly stating the **general problem** being addressed by this particular study.

One sentence summarizing the main result (with the words "**here we show**" or their equivalent).

Two or three sentences explaining what the **main result** reveals in direct comparison to what was thought to be the case previously, or how the main result adds to previous knowledge.

One or two sentences to put the results into a more **general context**.

Two or three sentences to provide a **broader perspective**, readily comprehensible to a scientist in any discipline.

Keywords: keywords

Word count: X

Select relevant moderators in meta-regression using Bayesian penalization

Meta-analysis is a quantitative form of evidence synthesis, whereby effect sizes from multiple similar studies are aggregated. In its simplest form, this aggregation consists of the computation of a summary effect as a weighted average of the observed effect sizes. This average is weighted to account for the fact that some observed effect sizes are assumed to be more informative about the underlying population effect. Each effect size is assigned a weight that determines how influential it is in calculating the summary effect. This weight is based on specific assumptions; for example, the *fixed effect* model assumes that all observed effect sizes reflect one underlying true population effect size. This assumption is well-suited to the situation where effect sizes from close replication studies are meta-analyzed (**higgins_re-evaluation_2009?**, fabrigar_conceptualizing_2016, maxwell_is_2015). The *random effects* model, by contrast, assumes that population effect sizes follow a normal distribution. Each observed effect size provides information about the mean and standard deviation of this distribution of population effect sizes. This assumption is more appropriate when studies are conceptually similar and differences between them are random (**higgins_re-evaluation_2009?**, fabrigar_conceptualizing_2016, maxwell_is_2015).

Not all heterogeneity in effect sizes is random, however. Quantifiable between-study differences may introduce systematic heterogeneity. Such between-study differences are known as "moderators." For example, if studies have been replicated in Europe and the Americas, this difference can be captured by a binary moderator called "continent." Alternatively, if studies have used different dosages of the same drug, this may be captured by a continuous moderator called "dosage." Systematic heterogeneity in the observed effect sizes can be accounted for using *meta-regression* (Viechtbauer & López-López, 2015). This technique provides estimates of the effect of one or more study characteristics on the overall effect size, as well as of the overall effect size and residual heterogeneity after

controlling for their influence.

One common application of meta-analysis is to summarize existing bodies of literature. In such situations, the number of moderators is often relatively high because similar research questions have been studied in different laboratories, using different methods, instruments, and samples. Each of these between-study differences could be coded as a moderator, and some of these moderators may explain systematic heterogeneity.

It is theoretically possible to account for the influence of multiple moderators using meta-regression. However, like any regression-based approach, meta-regression requires a relatively high number of cases (studies) per parameter obtain sufficient power to examine heterogeneity. In practice the number of available studies is often too low to examine heterogeneity reliably (Riley, Higgins, & Deeks, 2011). At the same time, there are many potential sources of heterogeneity, as similar research questions are studied in different laboratories, using different methods, instruments, and samples. This leads to a problem known as the "curse of dimensionality": the number of candidate moderators is large relative to the number of cases in the data. Such cases do not fit comfortably into the classic meta-analysis paradigm, which, like any regression-based approach, requires a high number of cases per parameter. Between-studies thus presents a non-trivial challenge to data aggregation using classic meta-analytic methods. At the same time, it also offers an unexploited opportunity to learn which differences between studies have an impact on the effect size found, if adequate exploratory techniques can be developed.

Addressing the curse of dimensionality necessitates *variable selection*: the selection of a smaller subset of relevant moderators from a larger number of candidate moderators. One way to perform variable selection is by relying on theory. However, in many fields of science, theories exist at the individual level of analysis (e.g., in social science, at the level of individual people). These theories do not necessarily generalize to the study level of analysis. Using theories at the individual level for moderator selection at the study level

amounts to committing the ecological fallacy: generalizing inferences across levels of analysis (**jargowskyEcologicalFallacy2004?**). To illustrate what theory at the study level of analysis might look like, consider the so-called *decline effect*. It is a phenomenon whereby effect sizes in a particular tranche of the literature seem to diminish over time (**schoolerUnpublishedResultsHide2011?**). It has been theorized that the decline effect can been attributed to regression to the mean: A finding initially draws attention from the research community because an anomalously large effect size has been published, and subsequent replications find smaller effect sizes. Based on the decline effect, we might thus expect the variable "year of publication" to be a relevant moderator of study effect sizes. Note that this prediction is valid even if year is orthogonal to the outcome of interest within each study. Until more theory about the drivers of between-study heterogeneity is developed, however, this approach will have limited utility for variable selection.

An alternative solution is to rely on statistical methods for variable selection. This is a focal issue in the discipline of machine learning (**hastieElementsStatisticalLearning2009?**). One technique that facilitates variable selection is *regularization*: shrinking model parameters towards zero, such that only larger parameters remain. Although this technique biases the parameter estimates, it also reduces their variance, which has the advantage of producing more generalizable results that make better predictions for new data (see **hastieElementsStatisticalLearning2009?**). This paper introduces *Bayesian regularized meta-regression* (BRMA), an algorithm that uses Bayesian estimation with regularizing priors to perform variable selection in meta-analysis. The algorithm is implemented in the function `brma()` in the R-package `pema`.

Statistical underpinnings

To understand how BRMA estimates the relevant parameters and performs variable selection, it is instructional to first review the statistical underpinnings of the aforementioned classic approaches to meta-analysis. First is the fixed-effect model, which

assumes that each observed effect size T_i is an estimate of an underlying true effect size Θ (**hedgesFixedRandomeffectsModels1998?**). The only cause of heterogeneity in observed effect sizes is presumed to be effect size-specific sampling variance, v_i, which is treated as known, and computed as the square of the standard error of the effect size. Thus, for a collection of k studies, the observed effects sizes of individual studies i (for i = 1,2, . . . k) are given by:

$$T_i = \Theta + \epsilon_i \tag{1}$$

$$\text{where } \epsilon_i \sim N(0, v_i) \tag{2}$$

Under the fixed effect model, the estimated population effect size $\hat{\theta}$ is obtained by computing a weighted average of the observed effect sizes. If sampling error is assumed to be the only source of variance in observed effect size, then it follows that studies with smaller standard errors estimate the underlying true effect size more precisely. The fixed-effect weights are thus simply the reciprocal of the sampling variance, $w_i = \frac{1}{v_i}$. The estimate of the true effect is a weighted average across observed effect sizes:

$$\hat{\theta} = \frac{\sum_{i=1}^{k} w_i T_i}{\sum_{i=1}^{k} w_i} \tag{3}$$

–>

Whereas the fixed-effect model assumes that only one true population effect exists, the random-effects model assumes that true effects may vary for unknown reasons, and thus follow a (normal) distribution of their own (Hedges & Vevea, 1998). This heterogeneity of the true effects is represented by their variance, τ^2. The random effect model thus assumes that the heterogeneity in observed effects can be decomposed into sampling error and between-studies heterogeneity, resulting in the following equation for the observed effect sizes:

$$T_i = \Theta + \zeta_i + \epsilon_i \tag{4}$$

$$\text{where } \zeta_i \sim N(0, \tau^2) \tag{5}$$

$$\text{and } \epsilon_i \sim N(0, v_i) \tag{6}$$

In this model, Θ is the mean of the distribution of true effect sizes, and τ^2 is its variance, which can be interpreted as the variance between studies.

If the true effect sizes follow a distribution, then even less precise studies (with larger sampling errors) may provide some information about this distribution. Like fixed-effect weights, random effects weights are still influenced by sampling error, but this influence is attenuated by the estimated variance of the true effect sizes. The random-effects weights are thus given by $w_i = \frac{1}{v_i + \hat{\tau}^2}$. It is important to note that, whereas the sampling error for each individual effect size is treated as known, between-study heterogeneity τ^2 must be estimated. This estimate is represented by $\hat{\tau}^2$.

Meta-regression. The random effects model assumes that causes of heterogeneity in the true effect sizes are unknown, and that their influence is random. Oftentimes, however, there are systematic sources of heterogeneity in true effect sizes. These between-study differences can be coded as moderators, and their influence can be estimated and controlled for using meta-regression. Meta-regression with p moderators can be expressed with the following equation, where $x_{1...p}$ represent the moderators, and $\beta_{1...p}$ the regression coefficients:

$$T_i = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + \zeta_i + \epsilon_i \tag{7}$$

$$\tag{8}$$

Note that β_0 represents the intercept of the distribution of true effect sizes after controlling for the moderators and the error term ζ_i represents residual between-studies

heterogeneity. This term is still included because unexplained heterogeneity often remains after accounting for the moderators (Thompson & Sharp, 1999). This is a mixed-effects model; the intercept and effects of moderators are treated as fixed and the residual heterogeneity as random (Viechtbauer & López-López, 2015).

To solve this model, the regression coefficients and residual heterogeneity must be estimated simultaneously. Numerous methods have been proposed to estimate meta-regression models, the most commonly used of which is restricted maximum likelihood (REML). REML is an iterative method, meaning it performs the same calculations repeatedly, updating the estimated regression coefficients and residual heterogeneity until these estimates stabilize. In contrast to a regularized analysis technique, this estimator produces low bias, which means that the average value of the estimated regression coefficients and residual heterogeneity is close to their true values (Panityakul et al., 2013; Hardy & Thompson, 1996).

Regularized regression. Regularized regression biases parameter estimates towards zero by including a shrinkage penalty in the estimation process. Before examining the Bayesian case, we will explain the principle using frequentist OLS regression as an example. OLS regression estimates the model parameters by minimizing the Residual Sum of Squares (RSS) of the dependent variable, which is given by:

$$RSS = \sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2$$

The resulting parameter estimates are those that give the best predictions of the dependent variable in the present dataset. Penalized regression, by contrast, adds a penalty term to the quantity to be minimized. One commonly used penalty is the L1-norm of the regression coefficients, or LASSO penalty, which corresponds to the sum of their absolute values. This gives the penalized residual sum of squares:

$$PRSS = RSS + \lambda \sum_{j=1}^{p} |\beta_j|$$

Where λ is a tuning parameter that determines how influential the penalty term will be. If λ is zero, the shrinkage penalty has no impact at all and the penalized regression will produce the OLS estimates. If $\lambda \to \infty$, all coefficient shrink towards zero, producing the null model. Because the penalty term is a function of the regression coefficients, the optimizer has an incentive to keep the regression coefficients as small as possible. Note that theLASSO penalty is but one example of a shrinkage penalty; other penalties exist, with some unique properties.

Bayesian estimation.

Simulation study

The present study set out to validate the BRMA algorithm using a simulation study. As a benchmark for comparison, we used restricted maximum likelihood meta-regression, which is the standard in the field. The algorithms are evaluated on three different criteria: The algorithms' predictive performance in new data, their ability to perform variable selection, and their ability to recover population parameters.

Performance indicators

Predictive performance reflects how well the algorithm is able to predict data not used to estimate the model parameters, in other words, it indicates the generalizability of the model. To compute it, for each iteration of the simulation both a training dataset and a testing dataset are generated. The model is estimated on the training data, which has a varying number of cases according to the simulation conditions. Predictive performance is then operationalized as the explained variance in the testing data, R^2_{test}. The testing data has 100 cases in all simulation conditions. The R^2_{test} reflects the fraction of variance in the

testing data explained by the model, relative to the variance explained by the mean of the training data. For a predictive performance measure, it is necessary to use the mean of the training data, as the mean of the testing is a descriptive statistic of that sample. The resulting metric R^2_{test} is expressed by the following equation:

$$R^2_{cv} = 1 - \frac{\sum_{i=1}^{k}(y_{i-test} - \hat{y}_{i-test})^2}{\sum_{i=1}^{k}(y_{i-test} - \bar{y}_{train})^2}$$

With k being the number of studies in the testing dataset, $\hat{y}_{i-test}$ being the predicted effect size for study i, and $\bar{y}_{train}$ being the mean of the training dataset.

The algorithms' ability to perform variable selection was evaluated by three indices: sensitivity, specificity, and the fraction of the product over the sum (FPS). These indices are based on proportions computed across replications within each condition. Sensitivity P is the ability to select true positives, or the probability that a variable is selected, $S = 1$, given that it has a non-zero population effect: $P = p(S = 1||\beta| > 0)$. Specificity is the ability to identify true negatives, or the probability that a variable is not selected given that it has a zero population effect: $N = p(S = 0|\beta = 0)$. It is possible for a poorly-fitted model to score highly on specificity by predicting all null effects while detecting no true positives. Therefore, we additionally computed a the fraction of the product over the sum (FPS) as an overall performance criterion, which is given by $FPS = 2(\frac{P*N}{P+N})$.

The ability to recover population parameters β and τ^2 was examined in terms of bias and variance of these estimates. The bias is given by the mean deviation of the estimate from the population value, and the variance is given by the variance of this deviation.

Design factors

The simulation conditions consist of all unique combinations of the following design factors: the number of studies in the training data $k \in (22, 40, 80)$, average within-study sample size $\bar{n} \in (40, 100, 200)$, the population effect size of relevant moderators

$\beta \in (0, .2, .5, .8)$, the number of skewed normally distributed moderators $p \in (2, 3, 6)$, and residual heterogeneity $\tau^2 \in (.01, .04, .1)$. The scale parameters of the skewed normal moderators were selected from $\omega \in (0, 2, 10)$, where $\omega = 0$ corresponds to a standard normal distribution. The true effect size was simulated using two models: one with a linear effect of one moderator, $T_i = \beta x_{1i} + \epsilon_i$, and one with a non-linear (cubic) effect of one moderator, $T_i = \beta x_{1i} + \beta x_{1i}^2 + \beta x_{1i}^3 + \epsilon_i$, where $\epsilon_i \sim N(0, \tau^2)$. The observed effect size y_i was then simulated as a standardized mean difference (SMD), sampled from a non-central t-distribution. The design factors combined to produce unique conditions. For all simulation conditions, 100 datasets were generated.

Results

The RMA algorithm failed to converge for replications, all of which were characterized by low number of cases ($k \leq 40$) and high effect sizes $\beta \geq .5$. These cases were omitted from further analysis.

Predictive performance

Results indicated that the overall R^2_{test} was highest for BRMA with a Horseshoe prior and lowest for RMA, see 1. This difference was driven in part by the fact that explained variance was somewhat higher for the BRMA models when the true effect was not null, and by the fact that RMA had larger negative explained variance when the true effect was equal to zero (i.e., there was no effect to detect).

To determine the effect of the design factors on R^2_{test}, ANOVA's were performed; one per algorithm. Note that p-values are likely not informative due to the large sample size and violation of the assumptions of normality and homoscedasticity. The results should thus be interpreted as descriptive, not inferential, statistics. Table 2 reports the effect size η^2 of main effects and two-way interactions between simulation conditions. As many effects

of design factors were monotonously positive or negative for all algorithms, we provide an overview of the interpretation of those effects in the Table. Interpretation was less straightforward for a number of interactions; for these, the median R^2_{test} is graphed in Figure 1.

–>

Estimating residual heterogeneity

The ability of the algorithms to correctly estimate τ^2 was operationalized by subtracting the true value for τ^2 from the τ^2 estimated by the algorithms. Again, the median and Mean Absolute Deviation were used as metrics for performance. The RMA algorithm showed the best results, $\Delta\tau^2_{RMA} = 0.02 \pm 0.06$, followed by the MetaForest algorithm $\Delta\tau^2_{Mf} = 0.09 \pm 0.13$. The Pema algorithms performed worst $\Delta\tau^2_{Lasso} = 0.23 \pm 0.18$; $\Delta\tau^2_{Hs} = 0.23 \pm 0.17$. The finding that all medians are positive implies that all algorithms have a bigger tendency to overestimate τ^2 than underestimate it. One comment to make is that uncertainty of the estimates generally increased as $\Delta\tau^2$ also increased. This implies that there was more variation in performance as median performance worsened.

To determine the effect of the design factors on $\Delta\tau^2$ for all algorithms, four separate ANOVA's were performed. The effect size η^2 per condition per algorithm, including η^2 for all two-way interactions can be found in table 2. Again, the assumption of normality was violated.

The biggest predictor on the correct estimation of τ^2 was the estimated model. This was mainly because the algorithms overestimated τ^2 most when the model contained cubic terms. Image 5A shows the marginal relationship of the estimated model on $\Delta\tau^2$. It

becomes more clear why this overestimation occurred when showing the interaction between β and the model estimated on $\Delta\tau^2$, shown in image 6. First note the general trend that during estimation of all models τ^2 got more overestimated as β increased, except during estimation of the linear model, where the effect of β on $\Delta\tau^2$ was close to zero, except for MetaForest. However, note the scales for the y-axes. While estimating the two-way interaction, linear and exponential model, $\Delta\tau^2$ stayed well within a confined interval. However, the algorithms severely overestimated τ^2 when the model contained cubic terms. Especially MetaForest overestimated τ^2 substantially when $\beta = 0.8$ and the estimated model is cubic: $\Delta\tau^2_{MF} = 2.92 \pm 2.37$. The other algorithms also had a $\Delta\tau^2 > 1$ in these conditions, but the results were not as severe. Interestingly, the Pema algorithms even outperformed the RMA algorithm in these conditions.

The marginal effects of β on $\Delta\tau^2$ are shown in image 5B. MetaForest was affected most by the increase in β, but in general performed better than the Pema algorithms when $\beta < 0.8$. The RMA algorithm performed best overall.

The marginal effect of α on $\Delta\tau^2$ was rather minimal, although there was a slight decrease in $\Delta\tau^2$ as α increased. However, the decrease is more explicit when the interaction of α with the estimated model is added. Image 7 shows this interaction. The algorithms were rather unaffected by α for the linear model, and a small decrease in $\Delta\tau^2$ as α increased can be seen in the exponential model. When the two-way interaction model was estimated however, the algorithms benefitted as α increased, while for the cubic model, $\Delta\tau^2$ first increased as α increased from 0 to 2, but decreased as α increased from 2 to 10. RMA performed best, followed by MetaForest. The Pema algorithms performed similarly, but worst.

The effect of the true τ^2 on $\Delta\tau^2$ was rather unnoticeable for the RMA and MetaForest algorithms. The tendency for the Pema algorithms on the other hand, was to overestimate τ^2 more as the true τ^2 increased. Image 5C shows the marginal relationship.

The effect of the number of moderators on $\Delta\tau^2$ was not that large either. A small increase in $\Delta\tau^2$ can be seen in the RMA and MetaForest algorithm as the number of moderators increased which was not found for the Pema algorithms. However, A small note is that MetaForest did substantially increase in $\Delta\tau^2$ as more moderators were added and the estimated model is cubic. Image 8 shows the interaction between the number of moderators and the estimated model.

κ only had a substantial effect for MetaForest; the $\Delta\tau^2$ decreased quite rapidly if κ increased, especially when the cubic model was estimated. For the other algorithms, decreasing κ had little to no effect on correctly estimating the residual heterogeneity. Image 9 shows the interaction of κ with the estimated model.

Finally, the average number of observations in the studies did not have a substantial effect on $\Delta\tau^2$. Image 5D shows the marginal relationship.

Variable selection

To determine the extent to which the algorithms could perform variable selection correctly, the proportion true positives $[TP]$ and true negatives $[TN]$ were calculated. The TP and TN reflect how well the algorithms accredit importance to relevant moderators and discredit importance to irrelevant moderators respectively. It should be noted that TP could only take on values 1 or 0 per simulated iteration, because in all models where $\beta > 0$, only one moderator was simulated to be relevant. As β increased, the already relevant moderator increased in relevance, rather than spreading the relevance over the other moderators. TN had a bigger range and could take on values dependent on how many moderators were taken up in the model. E.g. when $\beta = 0$ and $n_{mods} = n$, $n + 1$ different proportions were possible, $\frac{0}{n}$ up until $\frac{n}{n}$.

There were no differences in variables selected by Highest Density Intervals or Confidence Intervals for both Lasso and Horshoe and so for both algorithms it did not

matter which interval type was analyzed. It was found that MetaForest had the highest proportion true positives: $TP_{Mf} = 0.98$, closely followed by RMA: $TP_{RMA} = 0.96$. Horshoe performed slightly better than Lasso; $TP_{Hs} = 0.91$; $TP_{Lasso} = 0.89$. As for TN, it was found that the pema algorithms performed best: TN_{Hs} and $TN_{Lasso} = 0.93$, followed by RMA: $TN_{RMA} = 0.89$. MetaForest performed worst by a large margin: $TN_{Mf} = 0.50$. The Mean Absolute Deviation for all algorithms on both TP and TN was 0, except for MetaForests performance on TN, where the Mean Absolute Deviation was 0.44.

Perfomance on TP and TN were very high for all algorithms, with all mean proportions, except MetaForests performance on TN, exceeding .89. This implies that MetaForest had issues excluding irrelevant moderators from the models. Plots were inspected to determine the effects of the design factors on the proportions. While inspecting the plots, there were found to be little marginal effects of the design factors on TN, while TP was more affected.

Firstly, κ only had a positive effect on TP. As κ increased, TP also increased. MetaForest had the highest TP, followed by RMA and, lastly, Horsehoe and Lasso. The increase in TP for higher values of κ was steeper for the Pema algorithms, however. There was also an interaction of κ with the estimated model shown in image 10. During estimation of the linear and two-way interaction model, the relationship of κ looked relatively linear. At $\kappa = 20$ the TP was relatively low for the algorithms, compared to the cubic model where TP starts at .98 and converges to 1 as κ increased. This latter relationship was also found for the exponential model, although the TP at $\kappa = 20$ was lower.

$\bar{n}$ had a positive and roughly linear relationship with TP for all algorithms. MetaForest performed best, followed by RMA, while Horseshoe and Lasso performed worst. Image 11A shows the relationship.

β had an interaction effect with the model estimated on TN. Only during estimation

of the two-way interaction model, TN decreased as β increased, For the other models, TN remained stable. This could be because the interaction model was only fitted when there were 3,4 or 7 moderators, while for the other models, only 2,3 or 6 moderators were used. Image 12 shows the relationship. The effect of β on TP was positive; as β increased, TP increased too.

The true τ^2 had a negative effect on TP, while α seemed to have little effect on both TP and TN. Image 11B shows the marginal relationship of τ^2 on TP.

Finally, there was an effect of number of moderators on TN, but only for the two-way interaction model. The TN increased as the number of moderators did. Image 13 shows the interaction. This relationship was reversed for TP and was found during estimation of all models, i.e. TP decreased as the number of moderators increased. Image 14 shows the relationship.

Discussion

353 ## References

Table 1

Mean and SD of predictive R2 for BRMA with a horseshoe (HS) and LASSO prior, and for RMA, for models with a true effect (ES != 0) and without (ES = 0).

	$\bar{R}^2_{HS}$	CI_{95}	$\bar{R}^2_{LASSO}$	CI_{95}	$\bar{R}^2_{RMA}$	CI_{95}
Overall	0.42	[-0.03, 0.87]	0.42	[-0.01, 0.87]	0.39	[-0.30, 0.87]
ES = 0	0.57	[0.04, 0.89]	0.56	[0.03, 0.88]	0.55	[-0.01, 0.88]
ES != 0	-0.01	[-0.04, -0.00]	-0.01	[-0.02, 0.00]	-0.10	[-0.40, -0.01]

Table 2

Effect size of design factors and their two-way interactions on R2 the different algorithms, and of the difference between algorithms. The comparison between HS and LASSO was zero in the second decimal for all conditions. Interpretation indicates whether the effect was uniformly positive or negative (for interactions, the effect of the second design factor in the interaction) for all algorithms.

Factor	HS	LASSO	RMA	HS vs. RMA	LASSO vs. RMA	Interpretation
ω	0.04	0.04	0.02	0.00	0.00	negative
ω :Model	0.18	0.18	0.09	0.00	0.00	positive
ω :M	0.00	0.00	0.00	0.00	0.00	other
β	0.89	0.89	0.80	0.02	0.03	positive
β:ω	0.02	0.02	0.01	0.00	0.00	other
β:Model	0.51	0.51	0.32	0.00	0.00	other
β:M	0.00	0.00	0.02	0.01	0.01	other
β:τ^2	0.04	0.04	0.02	0.00	0.00	other
k	0.04	0.05	0.11	0.02	0.01	positive
k :ω	0.00	0.00	0.00	0.00	0.00	negative
k :β	0.00	0.01	0.03	0.01	0.01	positive
k :n	0.00	0.00	0.00	0.00	0.00	positive
k :Model	0.00	0.00	0.00	0.00	0.00	positive
k :M	0.00	0.00	0.04	0.01	0.01	other
k :τ^2	0.00	0.00	0.00	0.00	0.00	negative
n	0.11	0.10	0.04	0.00	0.00	positive
n:ω	0.00	0.00	0.00	0.00	0.00	other
n:β	0.04	0.04	0.02	0.00	0.00	positive
n:Model	0.04	0.04	0.02	0.00	0.00	positive

Table 2 continued

Factor	HS	LASSO	RMA	HS vs. RMA	LASSO vs. RMA	Interpretation
n:M	0.00	0.00	0.00	0.00	0.00	negative
n:τ^2	0.01	0.01	0.01	0.00	0.00	negative
Model	0.33	0.34	0.18	0.00	0.00	positive
M	0.00	0.01	0.06	0.02	0.01	negative
M:Model	0.00	0.00	0.00	0.00	0.00	positive
τ^2	0.12	0.12	0.06	0.00	0.00	negative
τ^2:ω	0.00	0.00	0.00	0.00	0.00	other
τ^2:Model	0.07	0.07	0.03	0.00	0.00	positive
τ^2:M	0.00	0.00	0.00	0.00	0.00	other

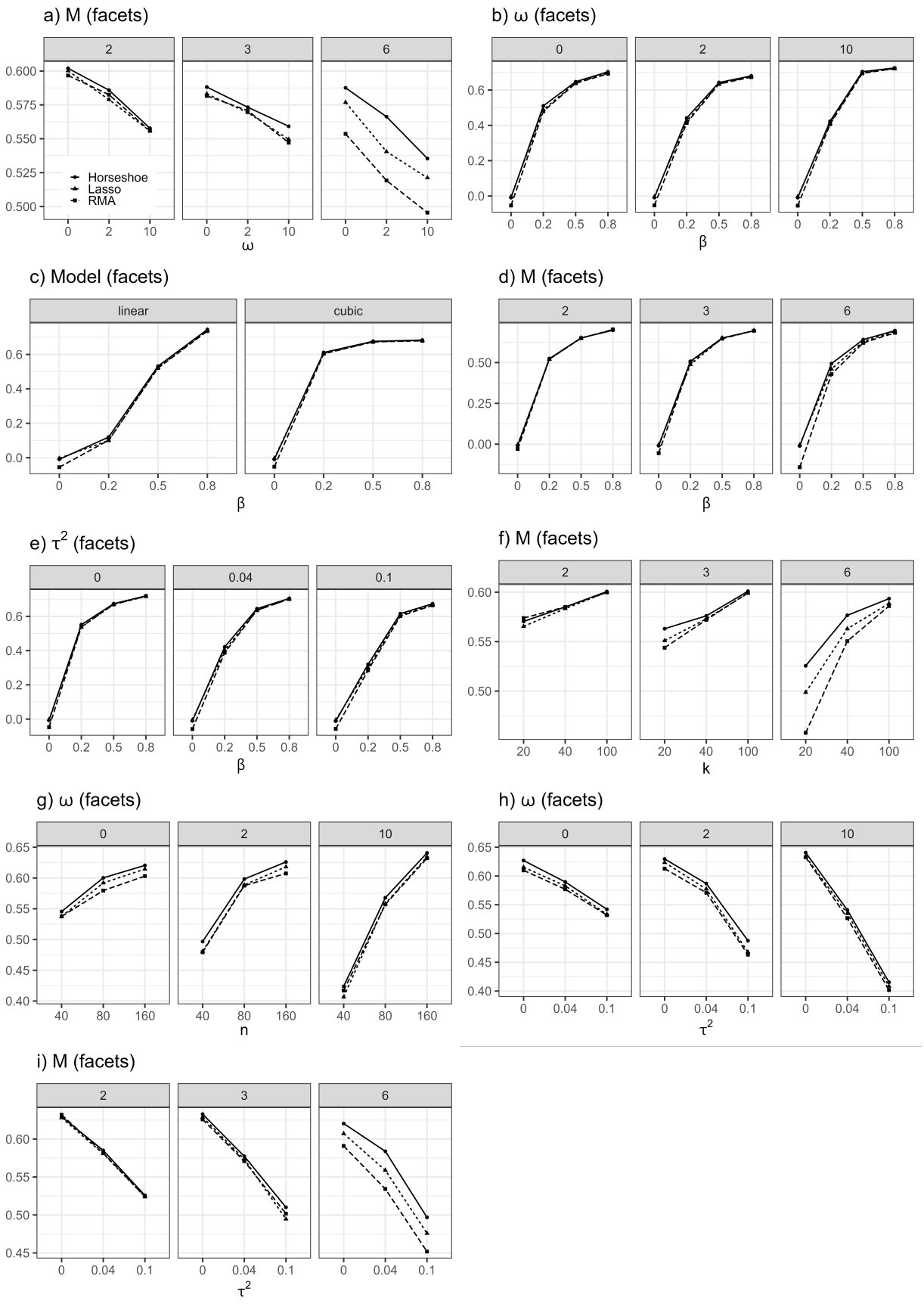

Figure 1. Predictive R2 for HS (circle, solid line), LASSO(triangle, dotted line) and RMA (square, dashed line) for interactions. One design factor is displayed in different panels, the