CHAPTER

TWO

REGRESSION TABLES WITH MULTI-STAGE METHODS

In this chapter we discuss how to create regression tables for high dimensional models. At stake is the creation of p-values, confidence intervals, and corresponding point estimators. Our method for the creation of p-values is theoretically the same as given by two previous works, but we make three important contributions in that (i) the original authors never, no public code was provided to test the results, and (ii) non-ideal default `options' were suggested in the original work which lead to poor simulation results (which can be easily fixed). Our method of constructing confidence intervals, is an original procedure and more than simple a natural extension of the p-value results; we provide corresponding theory asserting the coverage probability of the resulting intervals. Point estimators are obtained via a sensible interpolation of the confidence intervals.

For discussion of the implementation of our results, see chapter 5 where the **R** package `hdlm' is explored in detail.

2.1 Multi-Stage Methods and Theory

Recent advancements in the theory of sparse high dimensional regression have largely concerned the convergence rates of point estimation procedures, leaving open important issues in other aspects of statistical inference. Confidence intervals for high dimensional linear models, for instance, remain largely unstudied. The lack of research in this area is an important omission as confidence intervals remain incredibly useful for the purpose of data analysis. Without a concrete measurement of the accuracy of an estimator, it is difficult to draw useful conclusions from data. The convergence rates given in theory are often not acceptable for understanding estimator error in a finite setting since they correspond only to overall convergence, rather than a componentwise rate, of an estimator $\widehat{\beta}$ to β and typically include constants which are hard to compute.

A large part of the deficiency in p-values, standard errors and other measures of certainty in high dimensional learning stems from the fact that it is provenly hard to analyse model selection and inference which have been done either simultaneously or sequentially on the same dataset. Leeb,[1] Pötscher,[2] and Yang[3] have shown that there is in fact no generic procedure which uniformly estimates the conditional or unconditional distribution of post-model selection inference procedures.[4] A way to avoid these impossibility results, however, is to use part of the data for model selection and an independent part for parameter estimation.

[1] H. Leeb and B.M. Pötscher. "Model selection and inference: Facts and fiction". In: *Econometric Theory* 21.1 (2005), pp. 21–59.

[2] H. Leeb and B.M. Poetscher. "The finite-sample distribution of post-model-selection estimators and uniform versus nonuniform approximations". In: *Econometric Theory* 19.1 (2003), pp. 100–142; H. Leeb and B.M. Pötscher. "Can one estimate the conditional distribution of post-model-selection estimators?" In: *The Annals of Statistics* 34.5 (2006), pp. 2554–2591; H. Leeb and B.M. Pötscher. "Can One Estimate the Unconditional Distribution of Post-Model-Selection Estimators?" In: *Econometric Theory* 24.02 (2008), pp. 338–376.

[3] Y. Yang. "Can the strengths of AIC and BIC be shared? A conflict between model identification and regression estimation". In: *Biometrika* 92.4 (2005), pp. 937–950.

[4] R. Berk, L. Brown, and L. Zhao. "Statistical inference after model selection". In: *Journal of Quantitative Criminology* 26.2 (2010), pp. 217–236, For a practical discussion of what this means in data analysis, see.

A proposal by Wasserman and Roeder to construct p-values for high dimensional regression consists in utilizing the model selection property to conduct a multi-stage hybrid method.[5] The stated goal of their work is, with a fixed true support T of β, to derive a procedure which gives:

$$\limsup_{n\to\infty} \mathbb{P}\left[\widehat{T}_n \subset T\right] \geq 1 - \alpha. \tag{2.1}$$

In other words, they wish to have the probability of a false detection be at most α. Not worrying about perfect detection of T is nice, as it is impossible to detect very small signals and an upper bound on the signal strength can be awkward to deal with. While, given this, it is true that choosing $\widehat{T}_n = \emptyset$ technically satisfies equation 2.1, it is implicit that one wants to greatest power possible under the given false detection constraints.

In order to construct such an estimator, the observations of data are first randomly split into three groups: D_1, D_2, and D_3. Using the first subset, a series of methods are used to fit a number of sparse models:

$$b_\lambda := \phi_\lambda(D_1),\ \lambda \in \Lambda \tag{2.2}$$

This might be, for instance, a finite set of lasso solutions with varying tuning parameters. Once these models have been fitted, the following prediction error is calculated over the second subset of data:

$$e_\lambda := \sum_{i\in D_2} (y_i - x_i^t b_\lambda)^2 \tag{2.3}$$

The goal is to determine which of the models fit in the first stage best predict the second set

[5] L. Wasserman and K. Roeder. "High dimensional variable selection". In: *Annals of statistics* 37.5A (2009), p. 2178.

of data. This splitting methodology is a very simple version of cross validation. Using these estimates, an initial conservative guess of the support T is made:

$$\widehat{S}_n := \text{support}\left\{\arg\min_{\lambda\in\Lambda}(e_\lambda)\right\} \tag{2.4}$$

It is assumed, at this point, that there is a very high probability that $\widehat{S_n}$ contains the true model; in fact, this probability should tend to zero as the sample size limits to infinity[6]. Under reasonable assumptions, this rate will fall off exponentially for the methods we discussed in chapter 2.

Now, with the final set of data D_3 ordinary least squares regression is run on the variables contained in $\widehat{S_n}$. The final trimming is done by screening p-values, and creating $\widehat{T_n}$ from those variables with a p-value less than the desired level α. If the model selection in the first two steps is truely conservative, this should correctly control the component-wise probability of a Type-I Error.

The three methods suggested by Wasserman and Roeder for the first step of their procedure are the lasso with various tuning parameters, forward step-wise regression with λ steps, and trimmed marginal regression with various cutoffs.[7]

2.2 Bootstrapping Multistage p-values

It has noticed that the resulting p-values of the two-stage method of Wasserman and Roeder can be quite sensitive to the choice of the random splitting of data into the first and second stages of the method. Figure 2.1 shows the distribution of p-values for one variable given

[6] The conservativeness of cross validation using the lasso is discussed in the appendix of Wasserman and Roeder. The result may actually be the most original and theoretically interesting piece of the paper.

[7] Here, Y is regressed individually on each X_i. The trimming involves setting any value less than λ equal to zero. While generally it is the T-values which should be used for trimming, the standardization of the columns of X makes these equivalent.

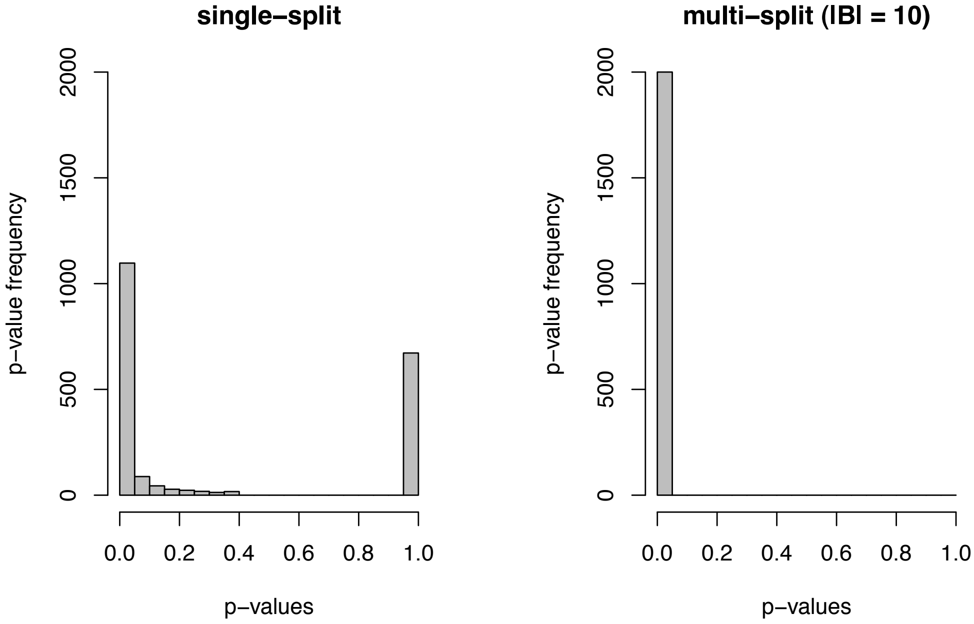

Figure 2.1: Distribution of p-values in a simple simulation with $p = 50$, $n = 25$, and $\beta^t = (1, 0, \ldots, 0)$, over 2000 trials. The left plot shows the p-values resulting from the single-split method, whereas the right plot shows p-values resulting from the multi-split method with 10 bootstrap runs and p-values combined via the discussed FDR procedure.

different partitions of the dataset. The random nature of the output make it at best difficult to analyse and at worse totally useless as a statistical method. Fortunately, a proposal by Meinshausen et al. provides a method for obtaining p-values which bootstraps over many splits of the dataset and intelligently pastes the results together.[8] We refer to their method as the `multi-split method', and the former by the 'single-split method'.

The major difficulty of the multi-step method is determining how to combine the p-values from each run into a single set of p-values. Consider just one column of the data matrix X and let $\{P_b\}_{b \in B}$ be the set of p-values for this one variable across all bootstrap runs b in B. This set can be viewed as a multiple hypothesis testing problem, with the

[8] N. Meinshausen, L. Meier, and P. Bühlmann. "P-values for high-dimensional regression". In: *Journal of the American Statistical Association* 104.488 (2009), pp. 1671–1681.

strange property that the null distribution in each test is the same: $H_0 : \beta_j \neq 0$. In multiple hypothesis testing there are essentially two forms of error which can be controlled. The family-wise error rate (FWER) concerns the chance of making any Type-II errors, whereas the false discovery rate (FDR) concerns the proportion of Type-II errors to rejected hypotheses. While not stressed in the Meinshausen paper, for our setting where all of the hypothesis tests have the same null hypothesis, these two rates will be exactly the same. What differs is the power of the generic methods for controlling these two error rates, when applied to our specific situation.

The classic method for control the family-wise error rate is the Bonferroni correction. Here p-values are `adjusted' by multiplying by the total number of hypothesis tests. Null hypotheses are rejected if the corresponding adjusted value is less than the desired family-wise error rate. A modification by Sture Holm, which is uniformly more powerful and still valid for any pattern of independence or dependence amongst the hypothesis, is to sort the p-values from smallest $P_{(1)}$ to largest $P_{(|B|)}$ and then adjust as:

$$\widetilde{P}_{(k)} := P_{(k)} \cdot (|B| - k + 1), \tag{2.5}$$

Where $|B|$ is the total number of hypothesis tests. The adjusted values are again compared to the maximum desired error rate and rejected accordingly.

The control of the false discovery rate is done in a similar fashion, as proven by Yosef Hochberg, Yoav Benjamini, and (in the case of dependent hypotheses) Daniel Yekutieli. Ordering again the p-values, find the smallest k such that:

$$P_{(k)} \leq \frac{\alpha k}{|B| \cdot c(|B|)} \tag{2.6}$$

Where α is the desired maximum false discovery rate and $c(|B|)$ is one in the case of independent or positively correlated tests and $\sum_i^{|B|} i^{-1} \approx \log(|B|) + 0.57721$ in the case of any other dependence structure.footnoteThe constant is the famous Euler-Mascheroni constant, originally defined in fact as the limitting difference between the natural logarithm and the harmonic series. All tests corresponding to p-values less than this $P_{(k)}$ have their null hypotheses rejected at the given level. Notice that the above equation doesn't not necessarily behave monotonically, and it is possible for instance to have $P_{(2)}$ not follow the above inequality even when $P_{(3)}$ does. In this case both null hypotheses are still rejected. It is possible to also write the FDR procedure in equivalent terms using adjusted p-values. In our case, the hypothesis tests can certainly be negatively correlated and therefore we need to use the most conservative formulation.

The proposal by Meinshausen et al. is a slight modification of FWER and FDR rates. For some $\gamma \in (0, 1)$ the define:

$$Q(\gamma) = \min\left\{1,\, q_\gamma\left\{P_{(j)}/\gamma;\, j = 1, \ldots, |B|\right\}\right\} \tag{2.7}$$

Where q_γ is the empirical γ-quantile function. For a fixed value of γ this value serves as a valid adjusted p-value; as the power of this procedure depends greatly on the choice of the quantile, a great improvement can be given by adaptively searching over a range of quantiles. It has been shown that this gives valid p-values with the addition of a extra constant. Specifically, by fixing a minimum value γ_{min} define the following adjusted p-value:

$$Q' = \min\left\{1,\, (1 - \log \gamma_{min}) \min_{\gamma \in (\gamma_{min}, 1)} Q(\gamma)\right\}. \tag{2.8}$$

We refer to this as the QA (quantile adjusted) p-value. The authors suggest setting γ_{min} equal to 0.05, which gives a multiple of just less then 4. An easy way to relate this procedure to the others, is to consider a set of possible values of γ consisting of $\{j/|B|,\, j = 1, \ldots, |B|\}$.

Then, a slightly less powerful version of the above Q' is given as:

$$Q'' = \min \left\{ 1,\ (1 + \log|B|) \min_j \frac{P_{(j)}}{k} \right\} \tag{2.9}$$

In this form, the QA proposal appears extremely similar to that of the FDR rate. The former has a slightly higher constant multiple, but benefits from continuously moving between observed p-values. A true difference between these two methods comes when the number of bootstraps is large, so that one may reasonably pick γ_{min} to be larger than $|B|^{-1}$. The authors suggest a minimum value of gamma around 0.05, so presumably this becomes a true factor when using more than a few dozen bootstrap replicates. Given the similarities, however, we lump FDR and QA together in the following discussion.

In our setting we care only if we choose to reject at least one or none of the hypotheses, as the null hypothesis are all the same. Notice that there is not a uniformly better choice between FWER and FDR/QA methods in this case. Consider for instance testing m hypotheses and getting the first p-value to be some small value q and the other $m - 1$ tests give a value of 1. We will reject the first hypothesis using FWER at the α level if $q \leq \alpha/m$ and using FDR/QA if $q \leq \alpha/(m\, c(m))$. Obviously the FWER will have a higher power (the same power is given by the FDR method if we were able to assume the tests were independent) in this case. In contrast, consider having all m of the p-values being equal to some value q. The FWER will again only reject if $q \leq \alpha/m$ whereas FDR/QA rejects as long as $q \leq \alpha/c(m)$. Given that the function $c(m)$ can be approximated by $log(m) + 0.577$, and therefore the FDR/QA test will have a higher power; the difference will be quite drastic when the number of hypothesis tests is large.

In general, the false discovery rate and quantile adjusted methods are more powerful when there is a large number of tests and a large number of relatively small p-values. Conversely, the family-wise error rate is more powerful when there is a small number of tests or one very small p-value but a large number of bigger values. In general, we suggest using at least 20 bootstraps and the the FDR method. The FWER is a good alternative for quick simulation runs when the number of bootstrap trials is significantly reduced. The QA proposal is only suggested only when one desires to determine a truly stable p-value by using a very high number of bootstrap samples; its limiting behaviour with a fixed γ_{min} is a bit more stable than the FDR procedure, however for smaller runs it has a tendency to be marginally less powerful and additionally requiring far more computational time.

Two properties are needed for the corrected p-values above to actually control the associated error rate, both concerning the statistical properties of the first stage model selection algorithm. Using our previous notation we have:

$$\lim_{n\to\infty} \mathbb{P}\left[\widehat{S}_n \subset T\right] = 0 \qquad \text{(Screening Property)}$$

$$|\widehat{S}_n| < \frac{n}{2} \qquad \text{(Sparsity Property)}$$

These properties are needed in order to run valid regression analyses in the second stage of the inference procedure. Once the individual p-values are shown to be valid, the prior theory of multiple hypothesis error control shows that our reported p-values are themselves asymptotically valid with either family-wise or false discovery error control methods. While both properties are necessary for asymptotically consistent p-values in the single split method, it is quite possible to have consistency in the multi-split method without these holding exactly.

The original work of Meinshausen et al. actually proposes an additional variant of that presented here, where all p-values are further adjusted globally with error correction done across all bootstrap runs and variables at the same time. This correction may actually make more sense in terms of pure model selection, where p-values are used primarily to conduct a secondary trimming of the model. As we are trying to produce an analogue to a traditional regression table, where p-values are not adjusted in any way, we suggest only working within each particular variable.

The literature on both single and multi-split two-stage methods has generally left the choice of the first stage model selector as generic as possible. While options such as the lasso and step-wise regression are suggested, the theory allows for any model selector which satisfies the screening and sparsity assumptions. The second stage procedure in comparison is always assumed to be standard least squares estimator; the actual underlying theory however does not rely on any particular theory of least squares. Any procedure which gives consistent p-values can be used in place of least squares. Options include robust alternatives such as MM-estimators and quantile regression, as well as generalised linear model methods such as logistic regression. The latter can be paired with a generalised linear model model selection procedure to create p-values and regression tables for generalised linear models.

2.3 Obtaining Confidence Intervals and Point Estimators

Typically, regression tables give either confidence intervals or standard errors in addition to p-values. In ordinary linear least squares regression these quantities all give equivalent information, albeit with a different focus; in other situations such as robust or quantile regression where the distribution of estimated coefficients is not assumed to be normal, these can actually depend on different methods. In the single-split variant of two-stage high dimensional regression, the second stage is and ordinary linear regression and therefore reporting either of both of these quantities is not terribly difficult. With the multi-split method, the particulars of how to combine different makes calculating either for the final regression table somewhat non-trivial. Here we will concentrate on a method for constructing asymptotically valid confidence intervals; standard errors are not calculated as they are not a particularly helpful quantity to estimate when using biased point estimators.

We wish to construct confidence intervals, individually for each variable, using a similar procedure as used to construct p-values in the previous section. Benjamini and Yekutieli[9]

[9] Y. Benjamini and D. Yekutieli. “False discovery rate-adjusted multiple confidence intervals for selected parameters”. In: *Journal of the American Statistical Association* 100.469 (2005), pp. 71–81.

have written about a multiple confidence interval generalisation to their FDR multiple hypothesis procedure. Unfortunately, unlike in the hypothesis setting, we find that having all of the confidence intervals correspond to the same quantity changes the natural method for combining the separate intervals and therefore have used some new theory. The main difficulty lies in the fact that confidence intervals from alternate bootstrap runs may in fact be disjoint; considering that the distribution of our multistage method is multi-modal, such a situation in fact arises quite frequently. Additionally, the proposed generalisation of FDR to confidence intervals can occasionally yield no solution; that is none of the confidence intervals are chosen, leaving us with no estimated interval. Such a situation seems inadequate given that a trivial conservative confidence interval can be constructed by a simple union across all bootstrap runs.

In order to address the issue with FDR confidence intervals, we propose an alternative procedure for aggregating confidence intervals across simulation runs. We do not suppose a particular process for the construction of confidence intervals, but rather allow for the use of any valid confidence interval construction procedure.

Theorem 2. *Let $(L_{j,b}, U_{j,b})$ be the confidence interval for the coefficient β_j from the b-th bootstrap replicate. For a fixed $\alpha \in (0,1)$, $k \in \{1, 2, \ldots \lfloor (m+1)/2 \rfloor\}$ and co-ordinate $j \in \{1, 2, \ldots p\}$, define the confidence intervals*

$$CI_j(\alpha) := \left(L_{j,(k)}(\alpha),\, U_{j,(m-k+1)}(\alpha)\right), \tag{2.10}$$

Where $L_{j,(k)}$ and $U_{j,(m-k+1)}$ are the k-th and $(m-k+1)$-th order statistics, respectively. If the original confidence intervals are valid, then we have

$$\mathbb{P}\left[\beta_j \in CI_j(\alpha)\right] \geq 1 - \alpha \cdot (2k-1), \tag{2.11}$$

Individually for each coefficient β_j.

Before proceeding to the proof of theorem 2, we note a few interesting properties of the proposed confidence intervals. Notice that if $k = 1$, the resulting confidence interval is simply the convex hull of the intersection of all bootstrapped confidence intervals. Consequently, the corresponding significance of the resulting interval CI_j is $1 - \alpha$; therefore, we see that when $k = 1$ our method collapses to the union bound. In contrast, consider for the moment setting $k = (m + 1)/2$, The resulting confidence interval corresponds to taking the maximum lower bound together with the minimum upper bound; the resulting confidence interval should have a significance level of $1 - \alpha \cdot m$. In other words, when $k = m$ we are using a Bonferroni correction and (assuming all of the lower bounds are less than all of the upper bounds) taking the intersection of the resulting intervals. We see then that our method allows for choosing an interval somewhere between the extremes of the Bonferroni correction and the union bound.

The size of k is restricted due to the fact that for larger k it is possible that the resulting confidence interval is in fact an empty set. The following lemma shows that this will not be true, regardless of the validity of the bootstrapped intervals, when k is properly bounded from above:

Lemma 1. *Regardless of the validity of the confidence intervals given in theorem 2, the resulting $CI_j(\alpha)$ is non-empty; that is, we have the right end-point of the interval no larger than the left end-point.*

Proof. The proof follows almost immediately from the definition of the order statistics; given that $m - k + 1 \geq k$ we have:

$$L_{j,(k)}(\alpha) \leq L_{j,(m-k+1)}(\alpha) \leq U_{j,(m-k+1)}(\alpha) \tag{2.12}$$

Therefore the collapsed confidence intervals $CI_j(\alpha)$, which are given in the the form of $\left(L_{j,(k)}(\alpha),\, U_{j,(m-k+1)}(\alpha)\right)$, must necessarily be non-empty. □

Notice that this lemma does not hold uniformly for higher values of k; as an example take a set of mutually disjoint confidence intervals.

Now that we have a basic understanding of the construction given in theorem 2, we proceed to a proof. We first construct a hierarchical sequence of confidence intervals, and then use a Holm-Bonferroni type correction to bound the coverage probability.

Proof. For a fixed j, let $\mathcal{C} = \{(L_{j,b}, U_{j,b}), b = 1, \ldots, m\}$ be the collection of all bootstrapped confidence intervals. Define for every $h = 1, 2, \ldots, 2k$ the following intervals A_h:

$$A_h = \begin{cases} \left(L_{(\frac{h+1}{2})},\, U_{(m+1-\frac{h+1}{2})}\right) & : h \,\text{odd} \\ \left(L_{(\frac{h}{2})},\, U_{(m-\frac{h+1}{2})}\right) & : h \,\text{even} \end{cases} \tag{2.13}$$

By this construction, A_{2h-1} is the confidence interval $CI_j(\alpha)$ given in theorem 2 with the parameter k equal to h. We now have the hierarchy of confidence intervals given by:

$$A_1 \supseteq A_2 \supseteq \cdots \supseteq A_{2k} \tag{2.14}$$

And furthermore that:

$$|\left\{c \in \mathcal{C}\ : c \in A_h\right\}| \geq |\left\{c \in \mathcal{C}\ : c \in A_(h+1)\right\}| + 1 \tag{2.15}$$

We can obtain this relationship by noticing that the difference between A_h and $A_{(h+1)}$ is the removal of one extremal endpoint, which at most can effect one confidence interval. Note also that this inequality is an equality when all of the confidence intervals have unique endpoints. Assume now that the confidence intervals have some random ordering $\{(L_i, U_i)\}_{i=1,\ldots m}$, independent of their actual values. Construct the following sequence of sets for all $h = 1, 2, \ldots, 2k$:

$$\widetilde{A}_h = \begin{cases} \bigcap_{j=1}^{h} \left(L_{\frac{j+1}{2}},\, U_{(m+1-\frac{j+1}{2})} \right) & : \; h\,\text{odd} \\ \bigcap_{j=1}^{h} \left(L_{\frac{j}{2}},\, U_{m-\frac{h+1}{2}} \right) & : \; h\,\text{even} \end{cases} \tag{2.16}$$

This construction is parallel to that of A_h, except that the indices are used instead of the order statistics; as successive confidence intervals do not contains one another, we need to use the slightly more verbose set intersection notation. Note that $\widetilde{A}_h \subseteq A_h$ for all h; also note that this set containment holds under arbitrary re-ordering of the indices. Also note, that by the Bonferroni confidence interval correction, we have (as long as the confidence intervals are valid) that $\beta \in A_h$ with probability $1 - \alpha h$ as this is simply an intersection of h randomly chosen confidence intervals. Now, we can use a simple union bound to finish off the proof. For a fixed k we have:

$$\begin{aligned} \mathbb{P}\left[\beta_j \notin A_{2k-1}\right] &\leq \mathbb{P}\left[\beta_j \notin \bigcup_{h=1}^{2k-1} \widetilde{A}_h\right] && (2.17) \\ &\leq \sum_{h=1}^{2k-1} \mathbb{P}\left[\beta_j \notin \widetilde{A}_h\right] && (2.18) \\ &\leq \alpha \cdot (2k-1) && (2.19) \end{aligned}$$

Taking the inverse of this equation, we see that the constructed confidence interval has coverage probability of no less than $1 - \alpha(2k-1)$, as stated. □

As can be seen in the proof, our method does not typically provide the smallest possible intervals; using $\widetilde{A}_h$ for a particular order of the indices would, for instance, typically give smaller intervals. However, our construction is a conservative choice, but should not be overly conservative, and has the nice property of always constructing non-empty confidence intervals.

Finally, we conclude by suggesting reasonable point estimators to correspond with the confidence intervals. The following definition works well given that the point estimator is assured to fall inside the resulting confidence interval:

Definition 1. *Given a confidence interval $CI_j(\alpha)$ as in theorem 2, define the point estimator $\widehat{\beta}_j$ as:*

$$\widehat{\beta} := median\{\beta_{j,b} : \beta_{j,b} \in CI_j(\alpha)\} \tag{2.20}$$

Use of the median rather than mean or other measure comes from the special status of coefficient estimates which are set exactly the zero. If a majority of the time, a coefficient is not included in the final model it makes sense to set it equal to exactly zero. On the other hand, if a coefficient is usually estimated to be around 1 but not included in about 20% of the model selection steps, a point estimator around 1 is better than one around 0.8. The median assures that both of these properties hold. We avoid giving an in depth in a convergence result regarding either point estimator, as convergence depends in a complex way on the underling data matrix as well as on the chosen model selection and confidence interval procedures. Obviously, however, since we are using a sample median, if the single-split procedure yields asymptotically consistent point estimators and the number of bootstrap trials is fixed, our resulting point estimator be consistent as well.

We note that a possible alternative point estimator would be to use the average of $L_{(\lfloor(m+1)/2\rfloor)}$, and $U_{(\lfloor(m+1)/2\rfloor)+1}$. Such a method is not likely to have the property of setting additional coefficients to zero, but does have the nice property that the point estimator does not depend on the confidence level α, will still having the point estimator always contained in the given confidence interval.

2.4 Generalisation to Learning Gaussian Graphical Models

Learning a Gaussian graphical model involves observing data from a multivariate normal distribution and identifying pairs of dimensions which are not conditionally independent given all other dimensions. As discussed in Chapter 2, this problem can be framed as a series of local model selection routines where each coordinate is sequentially used as the dependent variable in a high dimensional regression with the other coordinates as co-variates. Given the duality between these two procedures, it seems easy to amend the two-stage procedure in the regression framework to graphical models.

The first stage model selection procedure in learning a graphical model should generally be done using a global optimisation procedure, such as the graphical lasso, rather than a local method which is cycled over each node of the graph. Reasons for doing this include having typically better convergence results as well as generally having significant computational advantages in terms of both speed and memory usage. Out of necessity, the second stage p-values must be computed locally but should run quickly given the reduced number of variable following the model selection step. Even when model selection is done locally (due to a non-globalised minimisation routine), the same subset of the observations should be used in each step. Notice that for each bootstrap run, every selected edge will have two corresponding p-values; conveniently, the geometry of least squares guarantees that these will be exactly the same value as long as the same dataset is used in each stage.

A final interesting point regarding a two-stage graphical model learning procedure, are possible methods for displaying the results. Often in classical model selection procedures, the resultant model estimate is nicely laid out by way of graph drawing algorithm. Here, using the resultant p-values, we have a weighted form of graphical model where some edges have a stronger statistical significance than others. We discuss these issues in detail in our implementation of the two stage graphical model learning procedure in Chapter 6.

2.5 Model Selection Comparison

The main impetus to creating regression tables of high dimensional data was in order to provide a more complete picture of data to researchers studying such datasets. However, we can show that the resulting p-values can be used as a model selection scheme which is quite competitive in comparison to methods such as the traditional lasso. This is done by simply selecting for the model any variable which has a p-value smaller than some threshold; we used 0.001 in this section though the particular cut-off seems reasonably robust.

	(a)	(b)	(c)	(d)	(e)	(f)
hdlm (pilot)	0.1312	0.0558	0.1320	0.1956	1.0950	0.1526
hdlm (cv)	0.1433	0.0544	0.1849	0.1644	0.4620	0.1333
mc+	0.0293	0.0348	0.7157	0.0468	0.2140	0.0523
lasso (cv)	0.0233	0.0161	0.0136	0.0277	0.2266	0.0338
elastic net (cv)	0.0511	0.0394	0.0145	0.0902	0.3334	0.0513
lasso (oracle)	0.0228	0.0170	0.0099	0.0252	0.2147	0.0337
ols (oracle)	0.0008	0.0023	0.0041	0.0016	0.0071	·

Table 2.1: Simulated Median Mean Square Error. The baseline simulation (a) has $p = 1000$, $n = 200$, $\sigma = 0.5$ and $\beta = (1, 1, 0, \ldots, 0)$. Simulation (b) changes $p = 200$, (c) adds moderate correlation between X_1 and X_2, (d) adds moderate correlation between X_1 and X_3, (e) changes σ to $3/2$, and (f) changes $\beta_k \propto \exp(-k)$. Lasso oracle uses known σ and ols oracle uses known support of β; the weak sparsity of simulation (f) cannot run with least squares and hence is not included in the table.

We ran six simulation ensembles to capture typical variants of the model selection problem. The output of these simulations in terms of mean square error as well as model selection are shown in tables 2.1 and 2.2. A related set of simulations with slightly larger true models is given in 2.3.

We note that the two-stage method is comparable but slightly less accurate in terms of mean square error to other methods, however the model selection property is far better. The increase in mean square error is because there are many variables not weeded out in the first step (on purpose) and these exhibit high variation from the unbiased second stage ols procedure. Essentially, we are trading higher variance for lower bias. Importantly, however, the two stage method gives much more information; also, if one really wanted good parameter estimation it can be followed by a re-fit ols on the reduced model. While model selection is not always perfect, see table 2.3, it usually contains at the very least the true model.

In the end, the real benefit of the two-stage method is once again the ability to add some nuance to the model selection procedure. By using p-values we can determine which variables are clearly in the model, which have no evidence of being in the model, and which have only borderline signals.

	(a)	(b)	(c)	(d)	(e)
hdlm (pilot)	0.0	0.0	0.0	0.0	0.1
hdlm (cv)	0.0	0.2	0.1	0.0	0.0
mc+	3.4	7.9	2.9	3.6	2.7
lasso (cv)	37.0	8.9	6.8	17.9	15.9
elastic net (cv)	39.8	17.5	11.5	39.9	20.2
lasso (oracle)	10.5	4.9	7.5	9.3	8.0

Table 2.2: Simulated Mean Number of 'Mistakes' (added variables + missing variables) for model selection. The baseline simulation (a) has $p = 1000$, $n = 200$, $\sigma = 0.5$ and $\beta = (1, 1, 0, \ldots, 0)$. Simulation (b) changes $p = 200$, (c) adds moderate correlation between X_1 and X_2, (d) adds moderate correlation between X_1 and X_3, and (e) changes σ to 3/2. Lasso oracle uses known σ and hdlm uses cutoff of 0.001; stand-alone ols is not useful for high dimensional model selection and hence is not included.

	(a)	(b)	(c)	(d)	(e)
hdlm (pilot)	0.1	0.0	0.3	0.8	5.2
hdlm (cv)	9.4	0.0	5.6	6.0	5.4
mc+	3.8	7.3	4.5	3.6	4.1
lasso (cv)	58.4	32.9	47.8	54.6	50.7
elastic net (cv)	84.6	49.4	69.8	66.1	74.5
lasso (oracle)	16.5	7.9	16.1	16.1	15.6

Table 2.3: Simulated Mean Number of 'Mistakes' (added variables + missing variables) for model selection. Same as table 2.2, but larger model throughout with $\beta_1 = \beta_2 = \cdots = \beta_{10} = 0$.