University of Amsterdam
Psychology

Graduate School of Psychology

Form 2A - Research Master's Psychology: Thesis Research Proposal

1. General Information

1.1 Student information

Student name: David Coba

Student Id card number: 12439665

Address: -

Postal code and residence: -

Telephone number: -

Email address: coba@cobac.eu

Major: Psychological methods

1.2 Supervisor information

Supervisor name: Maarten Marsman

Second assesor name: Jonas Haslbeck

Specialization: Psychological Methods

1.3 Other information

Date: 1.04.2022

Status: First draft

Number of ECs for the thesis: 32EC

Ethics Review Board (ERB) code: -

2. Title and Summary of the Research Project

2.1 Title: Assessing the performance Occam's window for Bayesian model averaging

2.2 Summary of proposal

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Word count: /150

3. Project description

3.1 Prior research

When we perform statistical inferences, such as hypotheses tests about the inclusion of a parameter in a model or whether a parameter lays within an interval, we typically select a statistical model and then use that model to perform the inference. However, this approach underestimates the total uncertainty in our inferences, since it essentially ignores the uncertainty derived from the model selection process and produces overconfident conclusions (Leamer, 1978; Draper et al., 1987; Hoeting et al., 1999; for a recent review of the issue see Kaplan, 2021). There are different approaches under a Bayesian framework that allow to model the uncertainty of the model selection process, and address the issue of using a single model for our statistical inferences. These approaches can be categorized into two groups. The first group is using mixture models that encompass all possible models. To estimate the joint posterior distribution of all possible models researchers usually employ samplers like Markov chain Monte Carlo model composition (MC^3, Madigan & York, 1995), which is based on reversible jump Markov chain Monte Carlo (Green, 1995). However, these samplers are hard to implement, are computationally demanding and tend to have instability issues; which is why the second group of methods is generally preferred (Yao et al., 2018). The second group of approaches to multiple model inference is to combine the information of a set of candidate models $\mathcal{A}$. With these methods, the final inference $p(\Delta|D)$ is the weighted average of that inference across all candidate models $p(\Delta|M_k, D)$ shown in Equation 1. This approach allows to separate the use of multiple models into two steps: identifying a set of candidate models $\mathcal{A}$ and combining the uncertainty from those models.

$$p(\Delta|D) = \sum_{\forall k: M_k \in \mathcal{A}} p(\Delta|\mathcal{M}_k, D) w_k \tag{1}$$

There are two main methods to combine multiple models and not ignore the uncertainty of the model-selection process. The first method is Bayesian model averaging (BMA, Hinne et al., 2020; Hoeting et al., 1999; Leamer, 1978) and uses the posterior probability of candidate models as the model weights of Equation 1. This posterior probability

$$p(M_k|D) = \frac{p(D|\mathcal{M}_k)p(M_k)}{\sum_{\forall l: M_l \in \mathcal{A}} p(D|M_l)p(M_l)}$$

depends on the marginal likelihood of the data under each model

$$p(D|M_k) = \int p(D|\theta_k, M_k)p(\theta_k|M_k)d\theta_k$$

and their prior probability $p(M_k)$. To calculate the marginal likelihoods we need to integrate the product of the likelihood function of each model $p(D|\theta_k, M_k)$ and the prior distribution of the model parameters $p(\theta_k|M_k)$ over the whole parameter space. In most cases it is not possible to calculate the marginal likelihoods analytically, and we require of approximate solutions. The second method is model stacking, which minimizes the leave-one-out cross-validation (LOOCV) estimate of a loss function to assign weights to different models (Wolpert, 1992). Stacking is a common technique to aggregate point estimations from different models, but Yao et al. (2018) extend the method to combine Bayesian predictive distributions, producing combined uncertainty distributions similarly to BMA. It is possible to calculate LOOCV estimates from samples of the posterior distribution (Vehtari et al., 2016), which makes it convenient if one is using methods such as Markov chain Monte Carlo to estimate the posterior distributions in the first place.

The main difference between BMA and model stacking is their asymptotic behavior when the data-generating model is not in the set of candidate models $\mathcal{A}$. In this scenario, BMA will select the single model that minimizes the Kullback-Leibler divergence from the data-generating process, while model stacking will select the mixture of models that minimizes the loss function that was used to find the model weights (Yao et al., 2018). The literature is divided between proponents of marginal likelihood based methods, such as Bayes factors and BMA, and proponents of methods based on the posterior predictive distributions, such as LOOCV and model stacking. The disagreements seem to be rooted on differences in philosophical positions and scientific goals (Gronau & Wagenmakers, 2018, 2019; Lotfi et al., 2022; Vehtari et al., 2018).

When we do not have strong theoretical arguments to pre-select a set of candidate models $\mathcal{A}$, we can use a model search algorithm. One possible algorithm is Occam's window (Madigan & Raftery, 1994; A. E. Raftery et al., 1997), which is based on Occam's razor principle. Occam's razor (also known as the law of parsimony) states than when one is presented with competing hypotheses that explain equally well a particular phenomena, one should choose the simplest one. In general terms, Occam's window algorithm first selects a set of models that fit the data reasonably well, and then discards all models that have simpler counterparts that fit the data equally well. Formally, the first step equals constructing the set of models

$$\mathcal{A}' = \left\{ M_k : \frac{\max\{p(M_l|D)\}}{p(M_k|D)} \leq c \right\}$$

with posterior probabilities $p(M_k|D)$ not significantly lower than the model with highest posterior probability of all models $M_l \in \mathcal{A}'$. The constant c specifies the range of posterior probabilities—the size of the window—that fit the data reasonably well. For second step the algorithm identifies the set of models

$$\mathcal{B} = \left\{ M_k : \exists M_l \in \mathcal{A}', M_l \subset M_k, \frac{p(M_l|D)}{p(M_k|D)} > 1 \right\}$$

that have at least one submodel M_l in $\mathcal{A}'$ with greater posterior probability. The final set of candidate models is $\mathcal{A} = \mathcal{A}' \setminus \mathcal{B}$. Computationally the algorithm is a deterministic greedy search that performs two passes over the model space. The first pass goes from the bottom to the top (i.e. comparing the simplest models with p parameters to models with $p + 1$ parameters and so on), and the second pass starts from the most complex models and compares all the way to the simplest. To calculate posterior model probabilities $p(M_k|D)$ we need to compute the marginal likelihood $p(D|M_k)$ of each model, similarly to BMA. One of the drawbacks of Occam's window is that it overestimates the posterior probability of the selected "best" candidate models and it underestimates —essentially nullifies—the posterior probability of the rest of the models. This is by design and acknowledged by Madigan and Raftery (1994), and it is a trade-off we have to make to avoid having to combine information from the complete model space. Occam's window is implemented for linear regression models using priors that allow to analytically calculate the marginal likelihoods (A. E. Raftery et al., 1997) in the R package BMA (A. Raftery et al., 2015). There is also an extension of Occam's window to allows to model streams of data that become available sequentially (Onorante & Raftery, 2016).

Alternative model search algorithms include Bayesian adaptive sampling (BAS) and birth-death Markov chain Monte Carlo (BDMCMC). BAS samples without replacement from the space of possible models and uses the marginal likelihoods of the sampled models to iteratively estimate the marginal likelihoods of the models that remain unsampled (M. A. Clyde et al., 2011). BAS is available for (generalized) linear models as an R package (M. Clyde, 2021). BDMCMC (A. Mohammadi & Wit, 2015) samples from the joint posterior space of all possible models, and uses a Poisson process to model the rate at which the Markov chains jump from one model to another. BDMCMC is available in the R package BDGraph (R. Mohammadi & Wit, 2019) for graphical models, which uses a pseudo-likelihood funciton (Pensar et al., 2017) and an analytical approximation to the ratio of marginal likelihoods (R. Mohammadi et al., 2017).

Finally, we also want to give an overview of possible ways of approximating the marginal likelihoods that are required for BMA and Occam's window. The first and crudest one is to use the Bayesian information criterion (BIC, Schwarz, 1978) as an approximation. The BIC of a model M_k is defined as

$$\text{BIC}(M_k) = -2p\left(D|\widehat{\theta}, M_k\right) + d_{M_k} \log n,$$

where $p\left(D|\widehat{\theta}, M_k\right)$ is the likelihood of the maximum likelihood estimate for the parameter values under that model, d_{Mi} is the number of parameters of the model and n is the sample size. The logarithm of the marginal likelihood of a model can be approximated as

$$\log p\left(D|M_k\right) \approx p\left(D|\widehat{\theta}, M_k\right) - \frac{1}{2} d_{M_k} \log n$$

if we assume an unit information prior, which means that

$$\log p\left(D|M_k\right) \approx \frac{\mathrm{BIC}(M_k)}{-2}$$

and that the ratio of marginal likelihoods—the Bayes factor—between two models is

$$2 \log B_{ij} = -\mathrm{BIC}(M_i) + \mathrm{BIC}(M_k).$$

Another method to approximate the marginal likelihood is to use bridge sampling (Bennett, 1976; Gronau et al., 2017), which uses samples from the posterior distribution. Bridge sampling generally provides accurate approximations of the marginal likelihoods, but is also very computationally demanding. A method between BIC and bridge sampling in terms of accuracy and computational demands is the Laplace approximation (De Bruijn, 1970; Kass & Raftery, 1995). This method approximates the posterior distribution with a normal distribution centered around the posterior mode, which can be estimated using a expectation-maximization algorithm. The Laplace approximation is accurate to the second moment of the posterior distribution, but it is possible to extend it and include more accurate approximations at the cost of more computational resources (Ruli et al., 2016). Lastly, note that in the context of Occam's window, it is possible to use a faster but less accurate approximation during model search, and use a slower but more accurate approximation during the model combination step.

3.2 Key questions

The main goal of this project is to assess how Occam's window model search algorithm performs in general terms. To our knowledge there are no simulation studies evaluating its performance under different conditions. We want to explore the possible trade-offs between accuracy and computational speed and how it compares to alternative model search algorithms. Specifically, we are motivated by the issue of deciding whether to include or not particular edges in graphical models. The number of possible graphical models grows exponentially with the number of variables, and we believe that Occam's window is a promising alternative to sampling from the joint posterior distribution of all possible models.

Word count: 1427/1200

4. Procedure

4.1 Operationalization

To address our research questions we will first implement Occam's window model search algorithm and then conduct a simulation study. We plan on implementing our algorithm and running our simulations in the Julia programming language (Bezanson et al., 2017).

There are multiple simulation conditions that we could consider. We have identified four categories of conditions, each with multiple potential conditions.

The choice of model: Linear regression models, logistic regression models, Gausian graphical models and Ising models.

The choice of model-search algorithm: Occam's window, BAS and BDgraph approach with BDMCMC.

The choice of model combination: BMA and model stacking.

Which approximation to the marginal likelihood: BIC, Laplace approximation, analytical approximations and bridge sampling.

- Possibly using different approximations during the model search than during the model combination step.

Testing all combinations of conditions is unrealistic, so we have ordered what we believe are the most interesting ones in order of priority. We intend on running simulations under the four proposed models. Linear regression is the obvious simplest choice to start developing the algorithms, logistic regression is a next step that increases the complexity of the procedure, and both kinds of graphical models are the ones that motivate this project. Regarding the choice of model search algorithm we will only implement Occam's window algorithm, and rely on the implementations of BAS for linear models and BDgraph for graphical models as benchmarks. Also, since Occam's window algorithm uses marginal likelihoods during model search, it is most practical to use BMA to combine the candidate models, which is also how the algorithm was originally conceived. We will not use model stacking during our simulations. Finally, regarding the choice of approximations of the marginal likelihood, we will only implement the BIC and Laplace approximations during Occam's window model search, since we believe they are the simplest and more likely to be optimal approximations respectively. To arrive at this order we have considered what are the simplest implementations that are pre-requisites to the more complex ones.

1. Occam's window with linear regression models and BIC approximation.
2. Occam's window with linear regression models and Laplace approximation.
3. Occam's window with logistic regression models and Laplace approximation.
4. Occam's window with Gaussian graphical models and BIC approximation.
5. Occam's window with Gaussian graphical models and Laplace approximation.
6. BAS with its current implementation in R.
7. BDgraph with its current implementation in R.
8. Occam's window with Ising models and BIC approximation.
9. Occam's window with Ising models and Laplace approximation.
10. Using Occam's window model search with BIC, re-run BMA but using the Laplace approximation.

11. Using Occam's window model search with BIC, re-run BMA but using bridge sampling.

We believe that it is realistic to complete up to step no. 9 in this project. Evaluating the performance of conditions no. 10 and no. 11 will most likely remain open questions for future research.

4.2 Sample characteristics

We plan on generating data from a set of models and evaluate how well each simulation condition recovers the characteristics of the true data-generating models. However, we do not think it makes sense to commit to specific data-generating processes at this stage of the project.

4.4 Data analysis

This project is inherently exploratory and, similarly to the last section, we do not think it makes sense to commit at this stage to a specific analysis plan. In general terms, to assess how well each model-search algorithm performs we will compare the posterior probabilities of the true data-generating model, and the posterior probabilities of including specific edges that are present on the data-generating model. To assess computational costs we will use real runtime in order to not penalize algorithms that benefit from parallel computations. If instead we used CPU time, we would be penalizing all parallelizable algorithms by a factor of the number of parallel processes or threads.

4.4 Modifiability of procedure

In section 4.1 we have ordered some possible simulation conditions in order of priority and we have estimated how many are realistic to complete during this thesis project. If our estimations prove to be overconfident, we can choose to exclude additional conditions, starting with the ones with lowest priority. Similarly, if everything goes smoother than planned, we can choose to simulate and analyze additional conditions.

Word count: 685/1000

5. Intended results

The main goal of this project is to assess in general terms how Occam's window performs. If our analysis concludes that the algorithm compares favorably against alternative methods, we will show that Occam's window can be a useful tool to supplement the use of BMA to avoid the problem of single model inference. We are motivated specially by the case of graphical models, where the space of possible models grows exponentially with the number of variables. Current approaches to sampling from the complete model space have limitations, and we anticipate that Occam's window can be a useful tool that is currently underused. In case that our results show that the performance of Occam's window does not compensate for its shortcomings, we would have provided an updated assessment of its performance that is currently lacking in the literature. Moreover, we expect to contribute software that implements BMA and Occam's window, and that integrates with the rest of the Julia ecosystem.

Word count: 163/250

6. Work plan

6.1 Time schedule

This thesis project consists of 28 EC, excluding the thesis proposal. This is equivalent to approximately 18 weeks working full time. We aim to complete and present the project by the 15th of July 2022. In broad terms we plan to achieve the following milestones each month:

April
- Week 1/2: Address feedback on the proposal and implement Occam's window algorithm for linear regression models using BIC as an approximation to the marginal likelihood.
- Week 3: Implement the Laplace approximation to the marginal likelihood and test its performance with linear regression models.
- Week 4: Implement the Laplace approximation for logistic models and buffer time.

May
- Week 1: Buffer time and hopefully enjoy the UvA teaching-free days.
- Week 2: Implement the Laplace approximation for graphical Gausian models and start running simulations.
- Week 3: Buffer time and start running simulations with BAS and BDGraph.
- Week 4: Continue running simulations and buffer time.

June
- Week 1: Implement the Laplace approximation for Ising models. Continue running simulations.
- Week 2: Continue running simulations and start analyzing results. Start writing the thesis.
- Week 3/4: Analyze results and thesis writing. Complete a first draft of the full thesis.

July
- Weeks 1/2: Complete writing the thesis and prepare the presentation.

As detailed in section 4.4 ”Modifiability of procedure”, the scope of this project is highly flexible, and we can adapt which conditions to include or exclude in our simulation study depending on how fast we progress.

6.2 Infrastructure

No special infrastructure is required to complete this project.

6.3 Data storage

We will keep the results of all our simulations under version control and with remote backups. We do not plan on collecting any data, and in the case we end up deciding to use empirical data we would use publicly available datasets.

6.3 Budget

In principle we will not require extra funds to complete this project. In the case that the computational resources that we have access to prove insufficient to conduct the simulations, we might consider using cloud computing services. In any case, such costs would not exceed the maximum budget.

Word count: 333/500

7. References

Bennett, C. H. (1976). Efficient estimation of free energy differences from monte carlo data. Journal of Computational Physics, 22(2), 245–268.

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. SIAM Review, 59(1), 65–98. https://doi.org/10.1137/141000671

Clyde, M. (2021). BAS: Bayesian variable selection and model averaging using bayesian adaptive sampling. https://cran.r-project.org/package=BAS

Clyde, M. A., Ghosh, J., & Littman, M. L. (2011). Bayesian adaptive sampling for variable selection and model averaging. Journal of Computational and Graphical Statistics, 20(1), 80–101.

De Bruijn, N. G. (1970). Asymptotic methods in analysis.

Draper, D., Hodges, J. S., Leamer, E. E., Morris, C. N., & Rubin, D. B. (1987). A research agenda for assessment and propagation of model uncertainty. Rand Corporation, Report N-2683-RC.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and bayesian model determination. Biometrika, 82(4), 711–732. https://doi.org/10.1093/biomet/82.4.711

Gronau, Q. F., Sarafoglou, A., Matzke, D., Ly, A., Boehm, U., Marsman, M., Leslie, D. S., Forster, J. J., Wagenmakers, E.-J., & Steingroever, H. (2017). A tutorial on bridge sampling. Journal of Mathematical Psychology, 81(nil), 80–97. https://doi.org/10.1016/j.jmp.2017.09.005

Gronau, Q. F., & Wagenmakers, E.-J. (2018). Limitations of bayesian leave-one-out cross-validation for model selection. Computational Brain & Behavior, 2(1), 1–11. https://doi.org/10.1007/s42113-018-0011-7

Gronau, Q. F., & Wagenmakers, E.-J. (2019). Rejoinder: More limitations of bayesian leave-one-out cross-validation. Computational Brain & Behavior, 2(1), 35–47. https://doi.org/10.1007/s42113-018-0022-4

Hinne, M., Gronau, Q. F., van den Bergh, D., & Wagenmakers, E.-J. (2020). A conceptual introduction to bayesian model averaging. Advances in Methods and Practices in Psychological Science, 3(2), 200–215. https://doi.org/10.1177/2515245919898657

Hoeting, J. A., Madigan, D., Raftery, A. E., & Volinsky, C. T. (1999). Bayesian model averaging: A tutorial [with comments by M. Clyde, David Draper and EI George, and a rejoinder by the authors]. Statistical science, 14(4), 382–417.

Kaplan, D. (2021). On the quantification of model uncertainty: A bayesian perspective. Psychometrika, 86(1), 215–238. https://doi.org/10.1007/s11336-021-09754-5

Kass, R. E., & Raftery, A. E. (1995). Bayes factors. Journal of the american statistical association, 90(430), 773–795.

Leamer, E. E. (1978). Specification searches: Ad hoc inference with nonexperimental data. Wiley.

Lotfi, S., Izmailov, P., Benton, G., Goldblum, M., & Wilson, A. G. (2022). Bayesian model selection, the marginal likelihood, and generalization. CoRR. http://arxiv.org/abs/2202.11678v1

Madigan, D., & Raftery, A. E. (1994). Model selection and accounting for model uncertainty in graphical models using occam's window. Journal of the American Statistical Association, 89(428), 1535–1546. https://doi.org/10.1080/01621459.1994.10476894

Madigan, D., & York, J. (1995). Bayesian graphical models for discrete data. International Statistical Review, 63(2), 215–232.

Mohammadi, A., & Wit, E. C. (2015). Bayesian Structure Learning in Sparse Gaussian Graphical Models. Bayesian Analysis, 10(1), 109–138. https://doi.org/10.1214/14-BA889

Mohammadi, R., Massam, H., & Letac, G. (2017). Accelerating bayesian structure learning in sparse gaussian graphical models. CoRR. http://arxiv.org/abs/1706.04416v3

Mohammadi, R., & Wit, E. C. (2019). BDgraph: An R package for Bayesian structure learning in graphical models. Journal of Statistical Software, 89(3), 1–30. https://doi.org/10.18637/jss.v089.i03

Onorante, L., & Raftery, A. E. (2016). Dynamic model averaging in large model spaces using dynamic occam s window. European Economic Review, 81, 2–14. https://doi.org/https://doi.org/10.1016/j.euroecorev.2015.07.013

Pensar, J., Nyman, H., Niiranen, J., & Corander, J. (2017). Marginal pseudo-likelihood learning of discrete markov network structures. Bayesian Analysis, 12(4), nil. https://doi.org/10.1214/16-ba1032

Raftery, A., Hoeting, J., Volinsky, C., Painter, I., & Yeung, K. Y. Y. (2015). Bma: Bayesian model averaging. https://cran.r-project.org/package=BMA

Raftery, A. E., Madigan, D., & Hoeting, J. A. (1997). Bayesian model averaging for linear regression models. Journal of the American Statistical Association, 92(437), 179–191. https://doi.org/10.1080/01621459.1997.10473615

Ruli, E., Sartori, N., & Ventura, L. (2016). Improved laplace approximation for marginal likelihoods. Electronic Journal of Statistics, 10(2), 3986–4009.

Schwarz, G. (1978). Estimating the dimension of a model. The annals of statistics, 461–464.

Vehtari, A., Gelman, A., & Gabry, J. (2016). Practical bayesian model evaluation using leave-one-out cross-validation and waic. Statistics and Computing, 27(5), 1413–1432. https://doi.org/10.1007/s11222-016-9696-4

Vehtari, A., Simpson, D. P., Yao, Y., & Gelman, A. (2018). Limitations of ”limitations of bayesian leave-one-out cross-validation for model selection”. Computational Brain & Behavior, 2(1), 22–27. https://doi.org/10.1007/s42113-018-0020-6

Wolpert, D. H. (1992). Stacked generalization. Neural networks, 5(2), 241–259.

Yao, Y., Vehtari, A., Simpson, D., & Gelman, A. (2018). Using Stacking to Average Bayesian Predictive Distributions (with Discussion). Bayesian Analysis, 13(3), 917–1007. https://doi.org/10.1214/17-BA1091

8. Further steps

Make sure your supervisor submits an Ethics Checklist for your intended research to the Ethics Review Board of the Department of Psychology at https://www.lab.uva.nl/lab/ethics/

7. Signatures

- ☐ I hereby declare that both this proposal, and its resulting thesis, will only contain original material and is free of plagiarism (cf. Teaching and Examination

Regulation in the research master's course catalogue).

☐ I hereby declare that the result section of the thesis will consist of two subsections, one entitled "confirmatory analyses" and one entitled "exploratory analyses" (one of the two subsections may be empty):

1. The confirmatory analysis section reports exactly the analyses proposed in Section 4 of this proposal.
2. The exploratory analysis section contains not previously specified, and thus exploratory, proposal analyses.

Location:	Student's signature:	Supervisor's signature:
Amsterdam		