BENCHMARKING UNCERTAINTY QUANTIFICATION FOR PROTEIN ENGINEERING

Kevin P. Greenman
Department of Chemical Engineering
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
kpg@mit.edu

Ava P. Soleimany & Kevin K. Yang
Microsoft Research New England
Cambridge, MA 02142, USA
{avasoleimany,yang.kevin}@microsoft.com

ABSTRACT

Machine learning sequence-function models for proteins could enable significant advances in protein engineering, especially when paired with state-of-the-art methods to select new sequences for property optimization and/or model improvement. Such methods (Bayesian optimization and active learning) require calibrated estimations of model uncertainty. While studies have benchmarked a variety of deep learning uncertainty quantification (UQ) methods on standard and molecular machine-learning datasets, it is not clear how well these results extend to protein datasets. In this work, we implement a panel of deep learning UQ methods on the Fitness Landscape Inference for Proteins (FLIP) benchmark regression tasks. We compare results across different degrees of distributional shift using metrics that assess each UQ method's accuracy, calibration, coverage, width, and rank correlation to provide recommendations for the effective design of biological sequences.

1 INTRODUCTION

Machine learning (ML) has already begun to accelerate the field of protein engineering by providing low-cost predictions of phenomena that require time- and resource-intensive labeling by experiments or physics-based simulations (Yang et al., 2019). It is often necessary to have an estimate of model uncertainty in addition to the property prediction, as the performance of an ML model can be highly dependent on the domain shift between its training and testing data. Because protein engineering data is often collected in a biased manner (Dallago et al., 2021), tailored ML methods are required to guide the selection of new experiments from a protein landscape. Uncertainty quantification (UQ) can inform the selection of experiments in order to improve a ML model or optimize protein function through active learning or Bayesian optimization.

In chemistry and materials science, several studies have benchmarked common UQ methods against one another on standard datasets and have used or developed appropriate metrics to quantify the quality of these uncertainty estimates (Scalia et al., 2020; Tran et al., 2020; Hirschfeld et al., 2020; Nigam et al., 2021; Soleimany et al., 2021). This work has illustrated that the best choice of UQ method can be dependent on the dataset and other considerations such as scaling.

While some protein engineering work has leveraged uncertainty estimates, these studies have been limited to single UQ methods such as ensembles (Mariet et al., 2020) or Gaussian processes (GPs) (Hie et al., 2020). In this work, we use a group of standardized, public protein datasets to benchmark a panel of UQ methods. Our chosen datasets include splits with varied degrees of domain extrapolation, enabling us to evaluate the methods in a setting similar to what might be experienced while collecting new experimental data. We assess each model using a variety of metrics that capture different aspects of desired performance, including accuracy, calibration, coverage, width, and rank correlation.

2 METHODS

This work uses several dataset splits to evaluate the performance of a panel of uncertainty methods across various evaluation metrics (Figure 1).

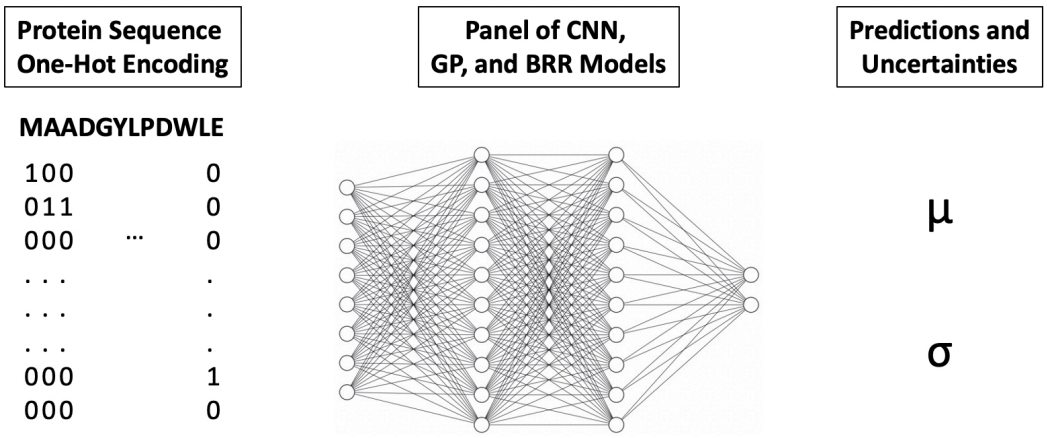

Figure 1: Overall schematic.

2.1 DATASETS AND TASKS

The landscapes used in this work were taken from the Fitness Landscape Inference for Proteins (FLIP) benchmark (Dallago et al., 2021). These included the binding domain of an immunoglobulin binding protein (GB1), adeno-associated virus stability (AAV), and thermostability (Meltome), which cover a large sequence space and a broad range of protein families. The FLIP benchmark includes several train-test splits, or tasks, for each landscape; most of these tasks are designed to mimic common, real-world data collection scenarios and are thus a more realistic assessment of generalizability than random train-test splits. However, random splits are also included as a point of reference. We chose 8 of the 15 FLIP tasks to benchmark the panel of uncertainty methods. The selected tasks include random sampling with no domain shift (AAV/sampled, Meltome/mixed, and GB1/sampled); the most relevant domain-shift regimes (AAV/mut_des, AAV/seven_vs_many, and GB1/one_vs_rest); and less aggressive domain shifts (GB1/two_vs_rest and GB1/three_vs_rest).

2.2 EVALUATION METRICS

To give a comprehensive report of model accuracy, we compute the following metrics on the test sets: root mean square error (RMSE), mean absolute error (MAE), coefficient of determination (R^2), and Spearman rank correlation (ρ). RMSE is more sensitive to outliers than MAE, so while both are informative independently, the combination of the two gives additional information about the distribution of errors. R^2 and ρ are both unitless and are thus more easily interpreted and compared across datasets.

We evaluated the quality of the uncertainty estimates using four metrics. First, ρ_{unc} is the the Spearman rank correlation between uncertainty and absolute prediction error. Following Kompa et al. (2021), we measure the coverage as the percentage of true values that fall within the 95% confidence interval ($\pm 2\sigma$) of each prediction. Kompa et al. (2021) define the width as the size of the 95% confidence region (4σ), but we calculate the width relative to the range of the training set as $4\sigma/R$ to make these values more interpretable across datasets. Finally, the miscalibration area (also called the area under the calibration error curve or AUCE) quantifies the absolute difference between the calibration plot and perfect calibration in a single number (Gustafsson et al., 2020).

2.3 Uncertainty Methods

We implemented eight uncertainty methods for this benchmark: linear Bayesian ridge regression (BRR), Gaussian processes (GPs), and six methods using variations on a convolutional neural network (CNN) architecture. We used a `scikit-learn` implementation of BRR (Pedregosa et al., 2011) and a GPyTorch implementation of continuous-kernel GPs (Gardner et al., 2018). The CNN implementation from FLIP provided the core architecture used by our dropout (Gal & Ghahramani, 2016), ensemble (Lakshminarayanan et al., 2017), evidential (Amini et al., 2020), mean-variance estimation (MVE) (Nix & Weigend, 1994), and last-layer stochastic variational inference (SVI) (Hoffman et al., 2013) models. For each dropout model, we tested dropout fractions of 0.1, 0.2, 0.3, 0.4, and 0.5 and reported the model with the lowest miscalibration area. The output of the evidential uncertainty can be divided into an epistemic and aleatoric component following the analysis of Amini et al. (2020), so we report these uncertainties separately along with their sum as the total uncertainty. For all models and landscapes, the sequences were featurized as one-hot encodings.

3 Results and Discussion

We trained each of the seven models described in Section 2.3 on each of the eight tasks described in Section 2.1 and evaluated their performance on the test set using the metrics described in Section 2.2. We compare model calibration and accuracy in Figure 3 using the legend in Figure 2 and the percent coverage vs. average width relative to range in Figure 4 using the same legend.

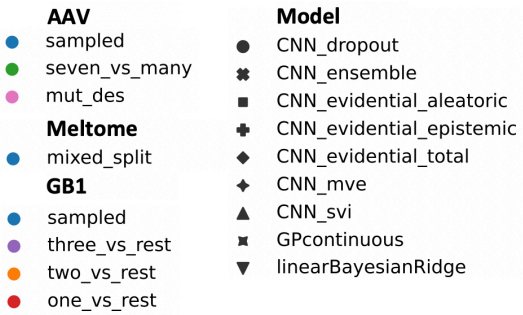

Figure 2: Legend for Figures 3 and 4. Splits are ordered by qualitative degree of domain extrapolation within each landscape (e.g. "sampled" requires the least extrapolation of the GB1 sets, while "one_vs_rest" requires the most).

As expected, the splits with the least required domain extrapolation tend to have more accurate models (lower RMSE). However, the relationship between miscalibration area and extrapolation is less clear; some models were highly calibrated on the most rigorous spits, while others were poorly calibrated even on random splits. There is no single method that performs consistently well across splits and landscapes, but some trends can be observed. For example, the CNN ensemble is often one of the highest accuracy models, but also one of the most poorly calibrated.

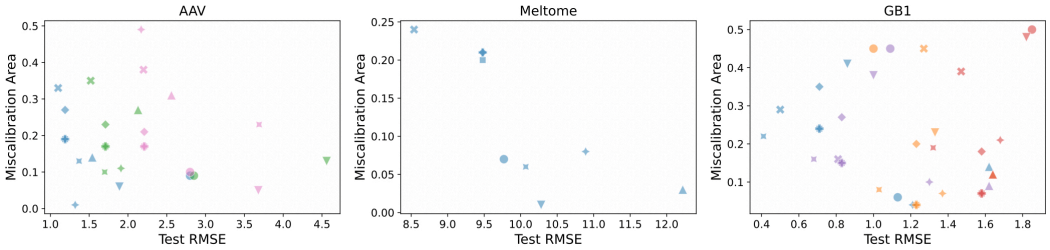

Figure 3: Miscalibration area vs. RMSE. Colors and shapes are as defined in Figure 2. It is desirable to have a model that is both accurate and well-calibrated, so the best performing points are those closest to the lower left corner of the plots.

Figure 4 illustrates that many methods perform relatively well in either coverage or width (corresponding to the the top and left limits of the plot, respectively), but few methods perform well in both. Among those that do, the GP model is the only method that is among the best across all landscapes and most splits. Similarly to Figure 3, there is some observable trend that more challenging splits are further from the optimal part (upper left) of the plot; this trend is more clear for the GB1 splits than for the AAV splits.

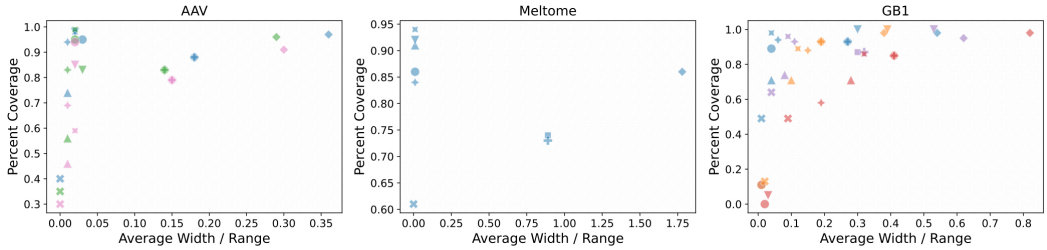

Figure 4: Coverage vs. average width / range. Colors and shapes are as defined in Figure 2. A good model exhibits high coverage and low width, which corresponds to the upper left of each plot.

Figure 5 compares the ranking performance of each method in terms of predictions relative to true values and uncertainty estimates relative to true errors. The splits are ordered according to domain shift within their respective landscapes, and the rank correlation of the predictions to the true labels generally decreases moving from left to right within a landscape. Ensembling was often among the best performance in ρ, and the GP and BRR models performed better on ρ on the most rigorous splits. Performance on ρ_{unc} is generally much worse than that on ρ, with a large number of results showing negative correlation. Dropout has one of the highest and most consistent ρ_{unc} across splits. All methods have ρ_{unc} near zero for the most challenging splits.

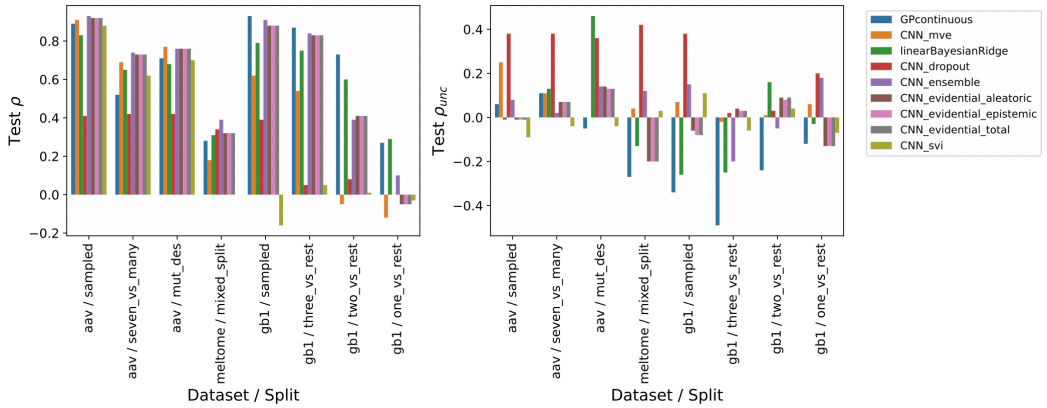

Figure 5: Spearman rank correlations of predictions and uncertainties vs. extrapolation. Within each landscape (AAV, Meltome, and GB1), splits are qualitatively ordered by the amount of domain shift between train and test sets, with the lowest domain shift on the left and highest shift on the right.

4 Conclusion

Calibrated uncertainty estimations are necessary for effective property optimization or model improvement using Bayesian optimization or active learning. In this work, we have benchmarked a panel of uncertainty quantification (UQ) methods on protein datasets, including on train-test splits that are representative of real-world data collection practices. After evaluating each method based on accuracy, calibration, coverage, width, and rank correlation, there is no method that performs consistently well across all metrics or all landscapes and splits. However, this study only examines models using one-hot-encoding representations of sequences; further research is needed to understand how more informative and generalizable representations such as the pretrained embeddings

from (Dallago et al., 2021) may impact performance, particularly on more rigorous splits. Additionally, while the evaluation metrics used herein provide valuable information, they are ultimately only a proxy for expected performance in Bayesian optimization and active learning. Retrospective studies using holdouts of the training sets would be useful for evaluating this performance more directly. A more thorough understanding of how to best apply UQ to sequence-function models will ultimately enable more effective protein engineering.

AUTHOR CONTRIBUTIONS

A.P.S. and K.K.Y. conceived the project. K. P. G. wrote the computer code, analyzed the data, and wrote the first manuscript draft. A.P.S. and K.K.Y. supervised the research and edited the manuscript.

ACKNOWLEDGMENTS

K.P.G. was supported by a Microsoft Research micro-internship and by the National Science Foundation Graduate Research Fellowship Program under Grant No. 1745302. We acknowledge the MIT Lincoln Laboratory Supercloud cluster (Reuther et al., 2018) at the Massachusetts Green High Performance Computing Center (MGHPCC) for providing high-performance computing resources to train our machine learning models.

REFERENCES

Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. Deep evidential regression. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 14927–14937. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/aab085461de182608ee9f607f3f7d18f-Paper.pdf.

Christian Dallago, Jody Mou, Kadina E Johnston, Bruce Wittmann, Nick Bhattacharya, Samuel Goldman, Ali Madani, and Kevin K Yang. FLIP: Benchmark tasks in fitness landscape inference for proteins. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=p2dMLEwL8tF.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/gal16.html.

Jacob Gardner, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. *Advances in neural information processing systems*, 31, 2018.

Fredrik K Gustafsson, Martin Danelljan, and Thomas B Schon. Evaluating scalable bayesian deep learning methods for robust computer vision. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 318–319, 2020.

Brian Hie, Bryan D. Bryson, and Bonnie Berger. Leveraging uncertainty in machine learning accelerates biological discovery and design. *Cell Systems*, 11(5):461–477.e9, 2020. ISSN 2405-4712. doi: https://doi.org/10.1016/j.cels.2020.09.007. URL https://www.sciencedirect.com/science/article/pii/S2405471220303641.

Lior Hirschfeld, Kyle Swanson, Kevin Yang, Regina Barzilay, and Connor W Coley. Uncertainty quantification using neural networks for molecular property prediction. *Journal of Chemical Information and Modeling*, 60(8):3770–3780, 2020.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 2013.

Benjamin Kompa, Jasper Snoek, and Andrew L. Beam. Empirical frequentist coverage of deep learning uncertainty quantification procedures. *Entropy*, 23(12), 2021. ISSN 1099-4300. doi: 10.3390/e23121608. URL https://www.mdpi.com/1099-4300/23/12/1608.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf.

Zelda Mariet, Ghassen Jerfel, Zi Wang, Christof Angermüller, David Belanger, Suhani Vora, Maxwell Bileschi, Lucy Colwell, D Sculley, Dustin Tran, et al. Deep uncertainty and the search for proteins. In *Workshop: Machine Learning for Molecules*, 2020.

AkshatKumar Nigam, Robert Pollice, Matthew FD Hurley, Riley J Hickman, Matteo Aldeghi, Naruki Yoshikawa, Seyone Chithrananda, Vincent A Voelz, and Alán Aspuru-Guzik. Assigning confidence to molecular property prediction. *Expert opinion on drug discovery*, 16(9):1009–1023, 2021.

David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, volume 1, pp. 55–60. IEEE, 1994.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Albert Reuther, Jeremy Kepner, Chansup Byun, Siddharth Samsi, William Arcand, David Bestor, Bill Bergeron, Vijay Gadepally, Michael Houle, Matthew Hubbell, Michael Jones, Anna Klein, Lauren Milechin, Julia Mullen, Andrew Prout, Antonio Rosa, Charles Yee, and Peter Michaleas. Interactive supercomputing on 40,000 cores for machine learning and data analysis. In *2018 IEEE High Performance extreme Computing Conference (HPEC)*, pp. 1–6. IEEE, 2018.

Gabriele Scalia, Colin A Grambow, Barbara Pernici, Yi-Pei Li, and William H Green. Evaluating scalable uncertainty estimation methods for deep learning-based molecular property prediction. *Journal of chemical information and modeling*, 60(6):2697–2717, 2020.

Ava P Soleimany, Alexander Amini, Samuel Goldman, Daniela Rus, Sangeeta N Bhatia, and Connor W Coley. Evidential deep learning for guided molecular property prediction and discovery. *ACS central science*, 7(8):1356–1367, 2021.

Kevin Tran, Willie Neiswanger, Junwoong Yoon, Qingyang Zhang, Eric Xing, and Zachary W Ulissi. Methods for comparing uncertainty quantifications for material property predictions. *Machine Learning: Science and Technology*, 1(2):025006, may 2020. doi: 10.1088/2632-2153/ab7e1a. URL https://doi.org/10.1088/2632-2153/ab7e1a.

Kevin K Yang, Zachary Wu, and Frances H Arnold. Machine-learning-guided directed evolution for protein engineering. *Nature methods*, 16(8):687–694, 2019.

A Appendix

Table 1: RMSE (↓) / MAE (↓) / R^2 (↑)

Dataset (Split)	BRR	Ens.	MVE	Dropout	GP	Evi. (a)	Evi. (e)	SVI
AAV (sampled)	1.89 1.50 0.62	1.10 0.83 0.87	1.32 1.01 0.82	2.80 2.37 0.18	1.37 1.03 0.80	1.19 0.89 0.85	1.19 0.89 0.85	1.54 1.18 0.75
AAV (7-vs-many)	4.56 3.30 -4.02	1.52 1.19 0.44	1.91 1.53 0.12	2.85 2.40 0.17	1.70 1.33 0.30	1.71 1.33 0.29	1.71 1.33 0.29	2.13 1.76 -0.10
AAV (mut-des)	3.68 2.58 -0.12	2.20 1.68 0.60	2.17 1.68 0.61	2.80 2.38 0.17	3.69 3.15 -0.13	2.21 1.69 0.60	2.21 1.69 0.60	2.56 2.11 0.46
Meltome (mixed)	10.28 7.88 0.22	8.54 6.55 0.46	10.89 7.88 0.12	9.77 6.89 0.15	10.07 7.49 0.25	9.48 7.12 0.34	9.48 7.12 0.34	12.22 8.79 -0.11
GB1 (sampled)	0.86 0.65 0.50	0.50 0.34 0.83	1.21 0.92 0.00	1.13 0.84 0.15	0.41 0.27 0.88	0.71 0.43 0.65	0.71 0.43 0.65	1.62 1.15 -0.80
GB1 (3-vs-rest)	1.00 0.77 0.38	0.81 0.58 0.60	1.30 1.06 -0.03	1.09 0.83 0.00	0.68 0.45 0.71	0.83 0.54 0.57	0.83 0.54 0.57	1.62 1.08 -0.61
GB1 (2-vs-rest)	1.33 1.04 -0.17	1.27 1.05 -0.07	1.37 0.94 -0.24	1.00 0.75 0.00	1.03 0.68 0.30	1.23 0.93 0.01	1.23 0.93 0.01	1.64 1.13 -0.78
GB1 (1-vs-rest)	1.82 1.53 -1.23	1.47 1.26 -0.17	1.68 1.16 -0.90	1.85 1.61 -3.10	1.32 1.13 -0.17	1.58 1.07 -0.67	1.58 1.07 -0.67	1.64 1.15 -0.80

Table 2: ρ (↑) / ρ_{unc} (↑)

Dataset (Split)	BRR	Ens.	MVE	Dropout	GP	Evi. (a)	Evi. (e)	SVI
AAV (sampled)	0.83 -0.01	0.93 0.08	0.91 0.25	0.41 0.38	0.89 0.06	0.92 -0.01	0.92 -0.01	0.88 -0.09
AAV (7-vs-many)	0.65 0.13	0.74 0.02	0.69 0.11	0.42 0.38	0.52 0.11	0.73 0.07	0.73 0.07	0.62 -0.04
AAV (mut-des)	0.68 0.46	0.76 0.14	0.77 -	0.42 0.36	0.71 -0.05	0.76 0.14	0.76 0.13	0.70 -0.04
Meltome (mixed)	0.31 -0.13	0.39 0.12	0.18 0.04	0.34 0.42	0.28 -0.27	0.32 -0.20	0.32 -0.20	0.00 0.03
GB1 (sampled)	0.79 -0.26	0.91 0.15	0.62 0.07	0.39 0.38	0.93 -0.34	0.88 -0.06	0.88 -0.08	-0.16 0.11
GB1 (3-vs-rest)	0.75 -0.25	0.84 -0.20	0.54 -0.02	0.05 0.02	0.87 -0.49	0.83 0.04	0.83 0.03	0.05 -0.06
GB1 (2-vs-rest)	0.60 0.16	0.39 -0.05	-0.05 0.01	0.08 0.03	0.73 -0.24	0.41 0.09	0.41 0.08	0.01 0.04
GB1 (1-vs-rest)	0.29 -0.03	0.10 0.18	-0.12 0.06	0.00 0.20	0.27 -0.12	-0.05 -0.13	-0.05 -0.13	-0.03 -0.07

Table 3: Coverage (↑) / $(4\sigma/R)$ (↓)

Dataset (Split)	**BRR**	**Ens.**	**MVE**	**Dropout**	**GP**	**Evi. (a)**	**Evi. (e)**	**SVI**
AAV (sampled)	0.98 0.02	0.40 0.00	0.94 0.01	0.95 0.03	0.98 0.02	0.88 0.18	0.88 0.18	0.74 0.01
AAV (7-vs-many)	0.83 0.03	0.35 0.00	0.83 0.01	0.95 0.02	0.99 0.02	0.83 0.14	0.83 0.14	0.56 0.01
AAV (mut-des)	0.85 0.02	0.30 0.00	0.69 0.01	0.94 0.02	0.59 0.02	0.79 0.15	0.79 0.15	0.46 0.01
Meltome (mixed)	0.92 0.01	0.61 0.00	0.84 0.01	0.86 0.01	0.94 0.01	0.74 0.89	0.73 0.89	0.91 0.01
GB1 (sampled)	1.00 0.30	0.49 0.01	0.94 0.06	0.89 0.04	0.98 0.04	0.93 0.27	0.93 0.27	0.71 0.04
GB1 (3-vs-rest)	1.00 0.53	0.64 0.04	0.93 0.11	0.11 0.01	0.96 0.09	0.87 0.30	0.87 0.32	0.74 0.08
GB1 (2-vs-rest)	1.00 0.39	0.13 0.02	0.88 0.15	0.11 0.01	0.89 0.12	0.93 0.19	0.93 0.19	0.71 0.10
GB1 (1-vs-rest)	0.05 0.03	0.49 0.09	0.58 0.19	0.00 0.02	0.86 0.32	0.85 0.41	0.85 0.41	0.71 0.28

Table 4: Miscalibration Area (↓)

Dataset (Split)	**BRR**	**Ens.**	**MVE**	**Dropout**	**GP**	**Evi. (a)**	**Evi. (e)**	**SVI**
AAV (sampled)	0.06	0.33	0.01	0.09	0.13	0.19	0.19	0.14
AAV (7-vs-many)	0.13	0.35	0.11	0.09	0.10	0.17	0.17	0.27
AAV (mut-des)	0.05	0.38	0.49	0.10	0.23	0.17	0.17	0.31
Meltome (mixed)	0.01	0.24	0.08	0.07	0.06	0.20	0.21	0.03
GB1 (sampled)	0.41	0.29	0.04	0.06	0.22	0.24	0.24	0.14
GB1 (3-vs-rest)	0.38	0.16	0.10	0.45	0.16	0.15	0.15	0.09
GB1 (2-vs-rest)	0.23	0.45	0.07	0.45	0.08	0.04	0.04	0.12
GB1 (1-vs-rest)	0.48	0.39	0.21	0.50	0.19	0.07	0.07	0.12