Building Conformal Prediction Intervals with Approximate Message Passing

Lucas Clarté[1] **Lenka Zdeborová**[1]

[1]Statistical Physics of Computation laboratory, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

Abstract

Conformal prediction has emerged as a powerful tool for building prediction intervals that are valid in a distribution-free way. However, its evaluation may be computationally costly, especially in the high-dimensional setting where the dimensionality and sample sizes are both large and of comparable magnitudes. To address this challenge in the context of generalized linear regression, we propose a novel algorithm based on Approximate Message Passing (AMP) to accelerate the computation of prediction intervals using full conformal prediction, by approximating the computation of conformity scores. Our work bridges a gap between modern uncertainty quantification techniques and tools for high-dimensional problems involving the AMP algorithm. We evaluate our method on both synthetic and real data, and show that it produces prediction intervals that are close to the baseline methods, while being orders of magnitude faster. Additionally, in the high-dimensional limit and under assumptions on the data distribution, the conformity scores computed by AMP converge to the one computed exactly, which allows theoretical study and benchmarking of conformal methods in high dimensions.

1 INTRODUCTION

Quantifying uncertainty is a central task in statistics, especially in sensitive applications. For regression tasks, the goal is to produce prediction sets instead of point estimates: consider here a dataset $\mathcal{D} = ((\boldsymbol{x}_1, y_1), \cdots, (\boldsymbol{x}_n, y_n))$ with independent samples of the same distribution, with $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \mathbb{R}$. Given a new input $\boldsymbol{x}$, we aim to produce a set of prediction $\mathcal{S}(\boldsymbol{x})$ that contains the observed label y with probability $1 - \kappa$ for $\kappa \in (0, 1)$. Conformal methods constitute a general framework used to produce such prediction sets with guarantees on their coverage. Among these methods, we can cite full and split conformal prediction (FCP and SCP) Vovk et al. [2005], Shafer and Vovk [2007] and Jackknife+ Barber et al. [2019]. In full conformal prediction, the prediction set of $\boldsymbol{x}$ is the set of labels y whose *typicalness* is sufficiently high. The computation of this typicalness is based on leave-one-out residuals that are computed on an augmented dataset that includes the test data. Full conformal prediction has been shown to provide the correct coverage under the exchangeability of the data samples and symmetry of the scoring function under the permutation of the data. However, the computation cost of FCP is proportional to the number of training samples and the number of possible labels, making it computationally very heavy in practice. Split conformal prediction (SCP) Shafer and Vovk [2007], Jing et al. [2018] is an efficient alternative to FCP, in which data is split between training and validation sets, the latter being used to calibrate the model after training. SCP is much more efficient than FCP, at the expense of statistical efficiency. Indeed, because the model is fitted on a lower amount of data than in FCP, the intervals of SCP are wider and thus less informative than FCP, as illustrated in Jing et al. [2018]. Similarly to SCP, the Jackknife+Barber et al. [2019] does not require to iterate over possible labels, but still requires to compute-leave-one-out residuals. It provides weaker coverage guarantees than FCP in exchange for faster computations. Finally, other works are concerned with accelerating full conformal prediction [Lei, 2019, Ndiaye and Takeuchi, 2019]. While the work of [Lei, 2019] focuses to the Lasso and ElasticNet, the method introduced in [Ndiaye and Takeuchi, 2019] is applicable to general convex empirical risks. Additionally, the work of Cherubin et al. [2021] leverages incremental learning in the context of classification, kernel density estimation and k-NN regression while Martinez et al. [2023] approximates FCP in the context of classification.

Uncertainty quantification in high dimensions – In this work, we will focus our attention on the *high-dimensional*

regime, where the number of samples n and the dimension d are both large with a fixed ratio $\alpha = {}^{n}/_{d}$. In this regime, many common uncertainty quantification methods are not applicable or quantify the true uncertainty wrongly. Full conformal prediction is computationally demanding as it needs to fit n estimators for each possible label. Alternatives, such as split conformal prediction or the Jackknife+ Barber et al. [2019] are more tractable, at the expense of statistical efficiency. On the other hand, the bootstrap Davison and Hinkley [1997] has been shown to fail in high-dimensional linear regression Clarté et al. [2024], Karoui and Purdom [2018] and with deep neural networks Nixon and Tran [2020]. Other methods based on ensembling, like the jackknife Quenouille [1956] or Adaboost Zhu et al. [2006], have been analyzed in high-dimension Takahashi [2024], Clarté et al. [2024], Loureiro et al. [2023], Liang and Sur [2022] and have been shown to be problematic in that setting as well. Authors of Bai et al. [2021b] have shown that unpenalized quantile regression achieves under-coverage in high dimensions.

High-dimensional inference with AMP – Approximate message passing (AMP) algorithms are a class of iterative equations used to solve inference problems in high-dimension under certain distributional assumptions Donoho et al. [2009], Zdeborová and Krzakala [2016]. These equations are usually derived by relaxing belief propagation equations in a graphical model Pearl [1988]. A central property of AMP algorithms is their state-evolution equations that track their behaviour in high dimensions. Thanks to these state-evolution equations, AMP has been used as an analytical tool to tackle a wide range of problems in high-dimensional statistics Sur and Candès [2019], Donoho et al. [2009], Bayati and Montanari [2010]. In the context of uncertainty quantification, AMP has been used to study the calibration of frequentist and Bayesian classifiers Bai et al. [2021a], Clarté et al. [2023b,a] and for change point detection Arpino et al. [2024]. Additionally to these analyses, AMP algorithms have also been used in practical scenarios, such as compressed sensing Donoho et al. [2009], genomics [Depope et al., 2024], to accelerate cross-validation [Obuchi and Kabashima, 2016] or for change point detection [Arpino et al., 2024]. Finally, in Bayesian learning, AMP can be used to compute marginals of the posterior distributions faster than with Monte-Carlo methods [Clarté et al., 2023b], or it can be used to establish fast sampling rigorously [El Alaoui et al., 2022]. However, to our knowledge, no work has applied AMP to accelerate the computation of full conformal prediction.

Contributions – Our contributions are four-fold:

- First, we apply the AMP algorithm on generalized linear regression to compute the prediction intervals of full conformal prediction. AMP accelerates FCP by approximating the n leave-one-out estimators simultaneously. We show that it still provides coverage guarantees under the standard assumption that the data is exchangeable.

- Second, we introduce the Taylor-AMP algorithm, which further accelerates the computations by removing the need to fit an estimator for each possible label. We claim that Taylor-AMP is a good approximation of AMP if the empirical risk minimizer only weakly depends on each sample.

- Third, we show that in a teacher-student model with Gaussian data and in the high-dimensional limit, AMP recovers the prediction intervals obtained by computing the leave-one-out scores exactly. As a consequence, our algorithm allows the study of conformal prediction in high dimensions and provides a non-trivial benchmark for other methods in this regime. We also leverage the state-evolution equations of AMP in the teacher-student model to predict sharply the performance of conformal prediction and benchmark it against Bayes-optimal estimation.

- Finally, we demonstrate the performance of Taylor-AMP on real data and benchmark it against other algorithms. We show that it provides the correct coverage and tight prediction intervals, thus demonstrating its practical interest.

To our knowledge, our work is the first to apply ideas from the area of approximate message-passing algorithms to full conformal prediction and opens the door to a new research direction in which methods from high-dimensional statistics can be used practically for uncertainty quantification. The AMP-based method has the coverage guarantees celebrated in conformal prediction, with possible wide prediction intervals if the scores are estimated inacurately. The method can be used with practical advantages in scenarios where the AMP is usable for estimation, for instance, genomics [Depope et al., 2024] or MRI reconstruction [Millard et al., 2020]. Another practical interest of our work stems from the utility of having non-trivial high-dimensional settings where FCP can be evaluated rapidly, as this may be useful for theoretical research and benchmarking of other more general speed-up methods.

Notation – For a set of real values $\boldsymbol{z} = z_1, \cdots, z_n$ we will write $\hat{q}_\kappa(\boldsymbol{z})$ the κ quantile of $\boldsymbol{z}$ (i.e the $\kappa \times n$ largest value). The normal distribution of mean μ and variance σ^2 will be noted $\mathcal{N}(\mu, \sigma^2)$ while we will denote by $\mathcal{L}(\mu, b)$ the Laplace distribution with density $p(x) = \frac{1}{2b} e^{-\frac{|x-\mu|}{b}}$. The element-wise product between two vectors or matrices A, B will be written $A \otimes B$. Jac denotes the Jacobian of a vector-valued function.

2 SETTING

We consider here the framework of generalized linear models for regression. Assume a training set $\mathcal{D} = (\boldsymbol{x}_i, y_i)_{i=1}^n$ with $\boldsymbol{x}_i, y_i \in \mathbb{R}^d \times \mathbb{R}$. Given a test sample $\boldsymbol{x}$, we want to build a prediction set $\mathcal{S}(\boldsymbol{x})$ that contains the true label y with probability $1 - \kappa$

$$\mathbb{P}_{\mathcal{D},\boldsymbol{x}} \left(y \in \mathcal{S}(\boldsymbol{x}) \right) \geqslant 1 - \kappa \,. \tag{1}$$

In (1), the randomness is on the training data and the test sample. We are interested in methods that provide the correct coverage with prediction sets of minimal size. In this work, we will focus on generalized linear models trained using empirical risk minimization

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathcal{R}\left(\boldsymbol{\theta}\right) = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \ell\left(y_i, \boldsymbol{\theta}^\top \boldsymbol{x}_i\right) + \sum_{\mu=1}^{d} r(\theta_\mu) \tag{2}$$

where ℓ is a convex loss and r is a convex regularizer. For concreteness, we will consider the cases of Ridge ($r(\theta) = \frac{\lambda}{2}\theta^2$) and Lasso ($r(\theta) = \lambda|\theta|$) regression, but our results apply to other problems such as quantile regression. Because the algorithms that we introduce rely on the computation of leave-one-out residuals, we introduce the leave-one out estimators $\hat{\boldsymbol{\theta}}_{-i}$ that are learned on the whole dataset except sample i.

2.1 FULL CONFORMAL PREDICTION

The basic procedure of full conformal prediction is to iterate over any possible label y, for which we define the augmented dataset $\mathcal{D}^+\left(y\right) = \mathcal{D} \cup (\boldsymbol{x}, y)$. We then compute the $n+1$ leave-one-out estimators $\hat{\boldsymbol{\theta}}_{-i}$ trained on $\mathcal{D}^+\left(y\right)$ from which we compute the conformity scores $\sigma_i\left(y\right)$. These scores will be used to compute test statistics that will determine the inclusion y in the prediction set $\mathcal{S}\left(\boldsymbol{x}\right)$. We first define

$$\hat{\boldsymbol{\theta}}_{-i}\left(y\right) = \arg\min_{\boldsymbol{\theta}} \sum_{j \neq i} \ell\left(y_j, \boldsymbol{\theta}^\top \boldsymbol{x}_j\right) + \ell\left(y, \boldsymbol{\theta}^\top \boldsymbol{x}\right) + \sum_{\mu} r\left(\theta_\mu\right) \tag{3}$$

that minimizes the empirical risk on $\mathcal{D}^+\left(y\right)$. We then define the conformity scores as the leave-one-out residuals:

$$\sigma_i(y) = |\hat{\boldsymbol{\theta}}_{-i}\left(y\right)^\top \boldsymbol{x}_i - y_i| \tag{4}$$

From these scores, the prediction set $\mathcal{S}_{\text{fcp}}\left(\boldsymbol{x}\right)$ is defined by

$$y \in \mathcal{S}_{\text{fcp}}\left(\boldsymbol{x}\right) \Leftrightarrow \sigma_{n+1}\left(y\right) \leqslant \hat{q}_{\lceil (1-\kappa)(n+1) \rceil / n}(\boldsymbol{\sigma}(y)) \tag{5}$$

in other words, a label y is included in the prediction set if the conformity score of the test sample, when using the $y_{n+1} = y$, is lower than the $\lceil (1-\kappa)\,(n+1) \rceil / n$ quantile of the scores $\sigma_1(y), \cdots, \sigma_{n+1}(y)$ [Vovk et al., 2005, Angelopoulos and Bates, 2022].

In what follows, we will refer as *exact LOO* the computation of the conformity scores (4) by solving the minimization problems (3) exactly. The prediction set $\mathcal{S}_{\text{fcp}}$ achieves the desired coverage on average under the assumption that the data is exchangeable and the regression function used to produce the conformity scores is symmetric [Vovk et al., 2005]. However, as noted before, fitting a model for all possible labels and computing the residuals by solving the minimization problem (3) is computationally heavy in practice. Methods have been developed to accelerate the computation of full conformal prediction, and in this paper, we introduce two algorithms that leverage tools from high-dimensional statistics, namely the AMP and Taylor-AMP algorithms. Contrary to exact LOO, our methods approximate the computation of the leave-one-out estimators (3) used to build prediction intervals. Note that other works[Lei, 2019, Ndiaye and Takeuchi, 2019] use $\sigma_i\left(y\right) = |\hat{\boldsymbol{\theta}}\left(y\right)^\top \boldsymbol{x}_i - y_i|$ for the conformity scores. While this definition does not require to compute leave-one-out estimators, this leads to issues if $\hat{\boldsymbol{\theta}}$ overfits the training data, which typically happens in the overparametrized regime. In this work, we will thus focus on the scores defined in Eq. (4).

2.2 SPLIT CONFORMAL PREDICTION

Split conformal prediction (SCP, also known as inductive conformal prediction) [Papadopoulos et al., 2002, Vovk et al., 2005] is an alternative to FCP that is computationally much cheaper. In the simplest form of SCP, $\mathcal{D}$ is split between the training and calibration sets $\mathcal{D}_{\text{train}}, \mathcal{D}_{\text{cal}}$. An estimator $\hat{\boldsymbol{\theta}}$ will be fit using $\mathcal{D}_{\text{train}}$, and the conformity scores $(\sigma_i)_{i=1}^{|\mathcal{D}_{\text{cal}}|}$ are computed on the calibration set. We then extract the $\lceil (1-\kappa) \times (n+1) \rceil$ quantile of the scores.

$$\sigma_i = |y_i - \hat{\boldsymbol{\theta}}^\top \boldsymbol{x}_i|, \qquad Q = \hat{q}_{\lceil (1-\kappa) \times (n+1) \rceil / n}\left(\sigma_i\right) \tag{6}$$

$$\mathcal{S}_{\text{SCP}}\left(\boldsymbol{x}\right) = \left[\hat{\boldsymbol{\theta}}^\top \boldsymbol{x} - Q, \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} + Q\right] \tag{7}$$

One drawback of (7) is that its prediction intervals are of the same size for all test samples. In this context, Romano et al. [2019] introduced conformal quantile regression, which combines split conformal prediction and quantile regression to accommodate potential heteroskedasticity and produce intervals with data-dependent length.

2.3 BAYES-OPTIMAL ESTIMATOR

Consider the Bayesian setting where the parameter to infer $\boldsymbol{\theta}_\star$ is sampled from a prior $p_{\boldsymbol{\theta}_\star}$ and the labels are generated by the likelihood distribution $p(y|\boldsymbol{\theta}_\star{}^\top \boldsymbol{x})$. One can then

compute the Bayes posterior

$$\boldsymbol{\theta} \sim p(\boldsymbol{\theta}|\mathcal{D}) \propto \prod_{i=1}^{n} p\left(y_i|{\boldsymbol{\theta}_\star}^\top \boldsymbol{x}_i\right) p_{\boldsymbol{\theta}_\star}\left(\boldsymbol{\theta}_\star\right) \tag{8}$$

which yields the *Bayes-optimal* estimator, with the lowest generalisation error. This posterior distribution yields the predictive posterior distribution

$$p(y|\mathcal{D}, \boldsymbol{x}) = \int \mathrm{d}\boldsymbol{\theta} p(y|\boldsymbol{\theta}^\top \boldsymbol{x}) p(\boldsymbol{\theta}|\mathcal{D}) \tag{9}$$

One can then build a prediction interval $\mathcal{S}_{\mathrm{bo}}(\boldsymbol{x})$ for the Bayes-optimal estimator using the *highest density interval*, which for a coverage $1-\kappa$ is the smallest set with measure $1-\kappa$.

Bayes posterior and maximum a posteriori In some settings, the empirical risk (2) corresponds to the logarithm of the Bayes-posterior. For instance, Ridge regression with $\lambda = 1$ corresponds to the log-posterior for the Gaussian prior $p_{\boldsymbol{\theta}_\star} = \mathcal{N}(0,1)$ while Lasso with $\lambda = 1$ matches the log posterior for the Laplace prior $p_{\boldsymbol{\theta}_\star} = \mathcal{L}(0,1)$.

3 APPROXIMATE MESSAGE PASSING FOR UNCERTAINTY QUANTIFICATION

3.1 COMPUTING RESIDUALS USING AMP

We first introduce the AMP algorithm, stated in Algorithm 1. Given the regression problem (2), AMP approximates $\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}$ of the empirical risk minimizer $\hat{\boldsymbol{\theta}}$. As we will show later, using AMP to solve Eq. (2) will allow us to simultaneously compute all the leave-one-out estimators instead of fitting the model n times, thus dramatically accelerating the computations. While AMP has been discussed extensively in the literature, for example, in Donoho et al. [2009], Zdeborová and Krzakala [2016], Mézard and Montanari [2009], we point the reader to Appendix A for its derivation.

Algorithm 1 requires to define a *channel* and *denoising* functions, respectively noted as g_{out} and f_w and defined as follows depending on the choice of loss and regularization:

$$g_{\mathrm{out}}(y, \omega, V) = \arg\min_z \ell(z, y) + \frac{1}{2V}(z-\omega)^2 \tag{10}$$

$$f_w(b, A) = \arg\min_z r(z) + \frac{1}{2A}(z - Ab)^2 \tag{11}$$

Above, g_{out} and f_w take scalar arguments but are applied on vectors in Algorithm 1 by applying the functions componentwise.

Channel and denoiser for Ridge and Lasso – In the general setting, computing g_{out} and f_w requires minimizing a scalar function. For concreteness, for Ridge regression and the Lasso these functions have a closed-form expression

$$\begin{cases} g_{\mathrm{out}}^{\mathrm{Ridge}}(y, \omega, V) &= \frac{y-\omega}{1+V} \\ f_w^{\mathrm{Ridge}}(b, A) &= \frac{b}{\lambda + A} \end{cases}, \tag{12}$$

$$\begin{cases} g_{\mathrm{out}}^{\mathrm{Lasso}}(y, \omega, V) &= \frac{y-\omega}{1+V} \\ f_w^{\mathrm{Lasso}}(b, A) &= \frac{b-\lambda}{A} \text{ if } b > \lambda, \frac{b+\lambda}{A} \text{ if } b < -\lambda \text{ else } 0 \end{cases}$$

but we provide in Appendix E examples of channels for other losses such as the pinball loss.

Algorithm 1 AMP

Input: Data $\mathrm{X} \in \mathbb{R}^{n \times d}$, $\boldsymbol{y} \in \mathbb{R}^n$

Define $\mathrm{X}^2 = \mathrm{X} \otimes \mathrm{X} \in \mathbb{R}^{n \times d}$ and initialize $\hat{\boldsymbol{\theta}}^{t=0} = \mathcal{N}(\mathbf{0}, \mathrm{I}_d)$, $\hat{\boldsymbol{v}}^{t=0} = \mathbf{1}_d$, $\boldsymbol{g}^{t=0} = \mathbf{0}_n$.

for $t \leq t_{\max}$ or until convergence **do**

/* *Update channel mean and variance*

$\boldsymbol{V}^t = \mathrm{X}^2 \hat{\boldsymbol{v}}^t$; $\boldsymbol{\omega}^t = \mathrm{X}\hat{\boldsymbol{\theta}}^t - \boldsymbol{V}^t \otimes \boldsymbol{g}^{t-1}$;

/* *Update channel*

$\boldsymbol{g}^t = g_{\mathrm{out}}(\boldsymbol{y}, \boldsymbol{\omega}^t, \boldsymbol{V}^t)$; $\partial \boldsymbol{g}^t = \partial_\omega g_{\mathrm{out}}(\boldsymbol{y}, \boldsymbol{\omega}^t, \boldsymbol{V}^t)$;

/* *Update prior mean and variance*

$\boldsymbol{A}^t = -{\mathrm{X}^2}^\top \partial \boldsymbol{g}^t$; $\boldsymbol{b}^t = \mathrm{X}^\top \boldsymbol{g}^t + \boldsymbol{A}^t \otimes \hat{\boldsymbol{\theta}}^t$;

/* *Update marginals* */

$\hat{\boldsymbol{\theta}}^{t+1} = f_w(\boldsymbol{b}^t, \boldsymbol{A}^t)$; $\quad \hat{\boldsymbol{v}}^{t+1} = \partial_b f_w(\boldsymbol{b}^t, \boldsymbol{A}^t)$

end for

/* *Compute the leave-one-out estimators with Eq.* (13)

for $1 \leqslant i \leqslant n$ **do**

$\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}} = \hat{\boldsymbol{\theta}}_{\mathrm{gamp}} - g_{\mathrm{gamp},i} \boldsymbol{x}_i \otimes \hat{\boldsymbol{v}}_{\mathrm{gamp}}$

end for

Return: $\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}, (\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}})_{i=1}^n$

Leave-one-out estimation – Using AMP, one can approximate the leave-one-out-estimators (3) and the associated residuals (4) with a single fit of the algorithm: for any sample i, an approximation of the $\hat{\boldsymbol{\theta}}_{-i}$ is given by the following expression

$$\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}}(y) = \hat{\boldsymbol{\theta}}_{\mathrm{gamp}}(y) - g_{i,\mathrm{gamp}}(y) \times \boldsymbol{x}_i^\top \otimes \hat{\boldsymbol{v}}_{\mathrm{gamp}}(y) \tag{13}$$

where all the vectors $\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}, \hat{\boldsymbol{v}}_{\mathrm{gamp}}, \boldsymbol{g}_{\mathrm{gamp}}$ are computed in Algorithm 1, and the dependency on the last label y is made explicit. We refer the reader to Appendix A for a justification of the above expression. The derivation is based on a close cousin of AMP, relaxed Belief Propagation (rBP), which is equivalent in the high-dimensional limit under Gaussianity assumptions on the data distribution, which we discuss in Section 3.3. At finite dimensions d the leave-one-out estimators $\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}}$ from (13) are only approximations of the solutions of (3) and may not be very good approximations. However, they still provide valid coverage guarantees, as essential in the conformal prediction.

Coverage guarantees for AMP – A central property of conformal prediction is that under very weak assumptions,

one get prediction sets that have the correct coverage. Indeed, a standard property of FCP is that if the data is exchangeable and the score function f, which maps samples to confirmity scores, is symmetric, then the prediction intervals given by f satisfy Eq. (1), as shown in Vovk et al. [2005]. Recall that *symmetric* means here that for any permutation $s : [1, n] \to [1, n]$, then $\hat{f}(\left(\boldsymbol{x}_{s(i)}, y_{s(i)}\right))_{i=1}^{n} = \left(\sigma_{s(i)}\right)_{i=1}^{n}$. We show in Appendix Appendix C that AMP is symmetric, which leads to the following property:

Property 1 *Consider training data $\mathcal{D} = (\boldsymbol{x}_i, y_i)_{i=1}^{n}$ and a test sample $\boldsymbol{x}$, assuming that the data is exchangeable. Consider the conformity scores $(\sigma_{i,\mathrm{gamp}})_i = |y_i - \hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}}^{\top} \boldsymbol{x}_i|$ where the leave-one-out estimators are computed using AMP:*

$$\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}} = \hat{\boldsymbol{\theta}}_{\mathrm{gamp}} - g_{i,\mathrm{gamp}} \boldsymbol{x}_i^{\top} \otimes \hat{\boldsymbol{v}}_{\mathrm{gamp}}$$

and the confidence set with target coverage $1 - \kappa$, defined as

$$\mathcal{S}_{\mathrm{fcp}}(\boldsymbol{x}) = \left\{ y | \sigma_{n+1} \leqslant \hat{q}_{\lceil (1-\kappa)(n+1) \rceil / n} \left(\sigma_i \right) \right\}$$

then, $\mathcal{S}_{\mathrm{fcp}}$ achieves coverage at $1 - \kappa$ on average

$$\mathbb{P}_{\mathcal{D}, \boldsymbol{x}} \left(y \in \mathcal{S}_{\mathrm{fcp}} \left(\boldsymbol{x} \right) \right) \geqslant 1 - \kappa \tag{14}$$

Note that Property 1 is valid at finite dimension and independently of the data distribution : AMP needs not to approximate precisely the leave-one-out residuals to achieve the correct coverage. In particular, we only require the data to be exchangeable for the property to hold.

3.2 TAYLOR-AMP

In the previous paragraphs, we saw that AMP can be used to accelerate the computation of the conformity scores $\sigma_i \left(y \right)$ by computing the n leave-one-out estimators simultaneously for a fixed label y of the test data. In this section, we present a variant of AMP called Taylor-AMP and described in Algorithm 2, whose goal is to further accelerate AMP by approximating the iteration over the set of possible labels: Taylor-AMP will compute the leave-one out estimators $\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}} \left(y \right)$ without fitting the model for each label y. The general idea is to approximate the quantities $\hat{\boldsymbol{\theta}}_{-i}^{\top} \boldsymbol{x}_i$ by an affine function around a reference value $\hat{y}$. To do so, we will compute the derivative of the estimators $\hat{\boldsymbol{\theta}}_{-i}(y)$ with respect to y, around $\hat{y}$. Then, for any possible label y, the corresponding scores will be approximated with

$$\begin{aligned}
\sigma_i \left(y \right) &= |y_i - \hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}} \left(y \right)^{\top} \boldsymbol{x}_i| \\
&= |y_i - \left(\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}} \left(\hat{y} \right) + (y - \hat{y}) \frac{\partial \hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}}}{\partial y} \left(\hat{y} \right) \right)^{\top} \boldsymbol{x}_i|
\end{aligned}$$

The central part is the estimation of $\frac{\partial \hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}}}{\partial y}$ using AMP. Indeed, $\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}$ solves a fixed point equation of the form

$$\boldsymbol{f}_{\mathrm{gamp}} \left(\hat{\boldsymbol{\theta}}_{\mathrm{gamp}} \left(y_{n+1} \right), y_{n+1} \right) = \hat{\boldsymbol{\theta}}_{\mathrm{gamp}} \left(y_{n+1} \right)$$

where we only make explicit its dependency y_{n+1} as the rest of the training data is fixed. Using the implicit function theorem, one can compute the derivative $\frac{\partial \hat{\boldsymbol{\theta}}_{\mathrm{gamp}}}{\partial y_{n+1}}$ from the implicit equation

$$\frac{\partial \hat{\boldsymbol{\theta}}_{\mathrm{gamp}}}{\partial y} \left(\hat{y} \right) = \left(\mathbf{I} - \mathrm{Jac} \left(\boldsymbol{f}_{\mathrm{gamp}} \right) \right)^{-1} \frac{\partial \boldsymbol{f}_{\mathrm{gamp}}}{\partial y} \left(\hat{y} \right) \tag{15}$$

which can be solved iteratively:

$$\Delta \hat{\boldsymbol{\theta}}^{t+1} = \mathrm{Jac} \left(\boldsymbol{f}_{\mathrm{gamp}} \right) \left(\Delta \hat{\boldsymbol{\theta}}^{t} \right) + \frac{\partial \boldsymbol{f}_{\mathrm{gamp}}}{\partial y} \left(\hat{y} \right) . \tag{16}$$

In Algorithm 2, we iterate Eq. (16) until convergence, at which point $\left(\Delta \hat{\boldsymbol{\theta}}, \Delta \hat{\boldsymbol{v}}, \Delta \boldsymbol{g} \right) = \left(\frac{\partial \hat{\boldsymbol{\theta}}}{\partial y}, \frac{\partial \hat{\boldsymbol{v}}}{\partial y}, \frac{\partial \boldsymbol{g}}{\partial y} \right)$. We provide more details, in particular the explicit form of the function $\boldsymbol{f}_{\mathrm{gamp}}$ in Appendix B.

To summarize, Algorithm 2 computes the derivatives $\Delta \hat{\boldsymbol{\theta}}_{\mathrm{gamp}}, \Delta \hat{\boldsymbol{v}}_{\mathrm{gamp}}, \Delta \boldsymbol{g}_{\mathrm{gamp}}$ of $\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}, \hat{\boldsymbol{v}}_{\mathrm{gamp}}, \boldsymbol{g}_{\mathrm{gamp}}$ around some value $\hat{y} = \hat{\boldsymbol{\theta}}^{\top} \boldsymbol{x}_n$ where $\hat{\boldsymbol{\theta}}$ minimizes (2) on $\mathcal{D}$. We can then approximate the leave-one-out estimators $\hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}} \left(y \right)$ by differentiating the expression of the leave-one-out estimators (13), which yields

$$\begin{aligned}
\frac{\partial \hat{\boldsymbol{\theta}}_{-i,\mathrm{gamp}}}{\partial y} \left(y \right) = \Delta \hat{\boldsymbol{\theta}} - g_{i,\mathrm{gamp}} \left(\hat{y} \right) \times \boldsymbol{x}_i \otimes \Delta \hat{\boldsymbol{v}}_{\mathrm{gamp}} \\
- \Delta g_{i,\mathrm{gamp}} \boldsymbol{x}_i \otimes \hat{\boldsymbol{v}}_{\mathrm{gamp}} \left(\hat{y} \right)
\end{aligned}$$

which allows us to compute the conformity scores of FCP in Eq. (4).

Justification of Taylor-AMP – Taylor-AMP is based on the idea that the value of the last sample only weakly affects the value of the estimator $\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}$. More precisely, in high-dimensions as $n, d \to \infty$, $\frac{\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}}{\partial y} \to 0$. This implies for instance that the data contains no outliers, whose value would induce a significant change in $\hat{\boldsymbol{\theta}}_{\mathrm{gamp}}$. We refer the reader to Appendix B.1 for more details: we numerically observe for synthetic Gaussian data that Taylor-AMP accurately approximates the leave-one-out predictions $\hat{\boldsymbol{\theta}}_{-i}^{\top} \boldsymbol{x}_i$ in high dimensions.

3.3 EXACTNESS IN HIGH DIMENSIONS FOR GAUSSIAN DATA

In this section, we provide guarantees on the size of the prediction intervals using conformity scores produced by AMP in high dimensions. Suppose that the samples $(\boldsymbol{x}_i, y_i)_{i=1}$ are i.i.d and follow the distribution

$$y_i \sim p(\cdot | \boldsymbol{\theta}_{\star}^{\top} \boldsymbol{x}_i), \qquad \boldsymbol{x}_i \sim \mathcal{N}(0, {}^{I_d}/_{d}) \tag{17}$$

Algorithm 2 Taylor-AMP

Input: Data $\mathrm{X} \in \mathbb{R}^{n \times d}$, $\boldsymbol{y} \in \mathbb{R}^n$

Compute $\left(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{v}}, \boldsymbol{\omega}, \boldsymbol{V}, \boldsymbol{A}, \boldsymbol{b}, \boldsymbol{g}, \partial \boldsymbol{g}\right)$ using Algorithm 1

Initialize $\Delta\hat{\boldsymbol{\theta}}^0 = \mathbf{0}, \Delta\hat{\boldsymbol{v}}^0 = \mathbf{0}, \Delta\boldsymbol{V}^0 = 0, \Delta\boldsymbol{\omega}^0 = \mathbf{0}$

for $t \leq t_{\text{max}}$ or until convergence **do**

$\Delta \boldsymbol{V}^t = \mathrm{X}^2 \Delta \hat{\boldsymbol{v}}^{t-1}$

$\Delta \boldsymbol{\omega}^t = X \Delta \hat{\boldsymbol{\theta}}^{t-1} - \Delta V \otimes \boldsymbol{g}^{t-1} - V \otimes \Delta \boldsymbol{g}^{t-1}$

$\Delta \boldsymbol{g}^t = \partial_{\boldsymbol{\omega}} g_{\text{out}} \Delta \boldsymbol{\omega}^t + \partial_{\boldsymbol{V}} g_{\text{out}} \Delta \boldsymbol{V}^t + \left(\partial_y g_{\text{out}|n}\right) \boldsymbol{e}_n$

$\Delta \partial \boldsymbol{g}^t = \partial^2_{\boldsymbol{\omega}^2} g_{\text{out}} \Delta \boldsymbol{\omega}^t + \partial_{\boldsymbol{V}} \partial_{\boldsymbol{\omega}} g_{\text{out}} \Delta \boldsymbol{V}^t + \left(\partial_y \partial_\omega g_{\text{out}|n}\right) \boldsymbol{e}_n$

$\Delta \boldsymbol{A}^t = -X^{2\top} \Delta \partial \boldsymbol{g}^t$

$\Delta \boldsymbol{b}^t = X^\top \Delta \boldsymbol{g}^t$

$\Delta \hat{\boldsymbol{\theta}}^t = \partial_b f_w \Delta \boldsymbol{b}^t + \partial_A f_w \Delta \boldsymbol{A}^t$

$\Delta \hat{\boldsymbol{v}}^t = \partial_b \left(\partial_b f_w\right) \Delta \boldsymbol{b}^t + \partial_A \left(\partial_b f_w\right) \Delta \boldsymbol{A}^t$

end for

Return: Derivatives $\left(\Delta\hat{\boldsymbol{\theta}}_{\text{gamp}}, \Delta\hat{\boldsymbol{v}}_{\text{gamp}}, \Delta \boldsymbol{g}, \right)$

for $\boldsymbol{\theta}_\star$ *teacher* vector that is to be recovered from the training data and with a likelihood function $p(\cdot|z)$ that is not known to the statistician e.g. $y = \boldsymbol{\theta}_\star{}^\top \boldsymbol{x} + \varepsilon$, with $\varepsilon \sim \mathcal{N}(0,1)$. Assume also that $\boldsymbol{\theta}_\star$ is random and its components are independently sampled from the same distribution $p_{\boldsymbol{\theta}_\star}$. In what follow we will assume that $p_{\boldsymbol{\theta}_\star}$ is either the standard normal $p_{\boldsymbol{\theta}_\star} = \mathcal{N}(0,1)$ or the Laplace distribution $p_{\boldsymbol{\theta}_\star}(z) = \frac{1}{2}e^{-|z|}$. Then, under these assumptions on $\boldsymbol{\theta}_\star$ and the data, in the high-dimensional limit where $n, d \to \infty$ with n/d fixed, the estimator $\hat{\boldsymbol{\theta}}_{\text{gamp}}$ converges to the true empirical risk minimizer, provided the samples $\boldsymbol{x}_i, y_i$ come from the distribution (17) as shown in Zdeborová and Krzakala [2016], Mézard and Montanari [2009], Donoho et al. [2009]. Thus, for any test sample $\boldsymbol{x}$ and any $\varepsilon > 0$

$$\mathbb{P}_{\mathcal{D},\boldsymbol{x}}\left(|\hat{\boldsymbol{\theta}}_{\text{gamp}}^\top \boldsymbol{x} - \hat{\boldsymbol{\theta}}^\top \boldsymbol{x}| < \varepsilon\right) \xrightarrow[n,d\to\infty, n/d=\alpha]{} 1 \tag{18}$$

Moreover, we show in Appendix A that in this high-dimensional limit, the estimators $\hat{\boldsymbol{\theta}}_{i,\text{gamp}}$ of Eq. (13) converge to the true leave-one-out estimators Eq. (3).

Exact distribution of the prediction intervals in high-dimensions Under the assumption in Eq. (17) on the data, we can leverage the *state-evolution equations* of AMP to compute exactly the distribution of the prediction interval $\mathcal{S}(\boldsymbol{x})$ for a random test vectors $\boldsymbol{x}$. We illustrate this asymptotic behaviour for Ridge regression in the following property :

Property 2 *Consider a training data $\mathcal{D} = (\boldsymbol{x}_i, y_i)_{i=1}^n$ and a test sample $\boldsymbol{x}$ following* (17) *with $y \sim \mathcal{N}\left(\theta_\star^\top \boldsymbol{x}, \Delta\right)$ and the estimator is Ridge regression with penalty λ. Then, in the limit $n, d \to \infty$ with $n/d = \alpha$, the prediction set $\mathcal{S}(\boldsymbol{x})$ is an interval of width*

$$2 \times q_{1-\kappa/2}(Z) \times \sqrt{\rho - 2 \times m + q + \Delta} \tag{19}$$

where $q_{1-\kappa/2}(Z)$ denotes the $1-\kappa/2$ quantile of the standard normal distribution $Z \sim \mathcal{N}(0,1)$, $\rho = \frac{1}{d}\|\boldsymbol{\theta}_\star\|^2$ and the overlaps m, q are the solutions of the following equations

$$\hat{m} = \frac{\alpha}{1+v}, \hat{q} = \frac{\alpha(\rho + q - 2m + \Delta)}{(1+v)^2}, \tag{20}$$

$$m = \frac{\rho\hat{m}}{\lambda + \hat{m}}, q = \frac{(\hat{m}^2\rho + \hat{q})}{(\lambda+\hat{m})^2}, v = \frac{1}{\lambda+\hat{m}} \tag{21}$$

We refer to Appendix F for a more general statement and the derivation of Property 2.

4 NUMERICAL EXPERIMENTS

In this section, we first show that on synthetic Gaussian data, our method correctly approximates the conformity scores while accelerating their computations by orders of magnitude. This allows us to compare FCP to other methods such as split conformal prediction and the Bayes-optimal estimator in a non-trivial high-dimensional setting. We then evaluate the methods on real datasets, showing the usefulness of AMP for uncertainty quantification beyond synthetic data with no distributional assumptions. In all of our numerical experiments, the prediction intervals will have a target coverage of 90%. For the sake of completeness, we provide experiments at other target coverages in Appendix D.1.

4.1 SYNTHETIC HIGH-DIMENSIONAL BENCHMARK

Coverage and size of prediction intervals – In this section, we consider synthetic data generated by the model described in Eq. (17). In Table 1, we first compute the coverage of Taylor-AMP for the Ridge and Lasso regressions at different values of λ. We see in the right-most column that our method provides the desired coverage. Moreover, on this synthetic data we compare the size of prediction intervals produced by exact LOO and observe that the average length are almost equal. This numerically validates the statement of Section 3.3 and shows that with Gaussian data, even at moderate dimension, Taylor-AMP is very close to exact LOO.

We also compute the similarity between the prediction intervals produced by Taylor-AMP with those returned by exact LOO, to show that both methods return the same intervals. To this end, we compute the *Jaccard index* between the exact and approximate intervals. Recall that the Jaccard index between two sets $\mathcal{S}_1, \mathcal{S}_2$ is defined as

$$\mathcal{J}\left(\mathcal{S}_1, \mathcal{S}_2\right) = \frac{|\mathcal{S}_1 \cap \mathcal{S}_2|}{|\mathcal{S}_1 \cup \mathcal{S}_2|} \in [0,1]$$

Problem	exact LOO	Taylor-AMP	SCP	CQP	Coverage of Taylor-AMP
Lasso ($\lambda = 1$)	3.9 ± 0.45	4.2 ± 0.8	4.3 ± 0.9	4.7 ± 0.9	0.9
Ridge ($\lambda = 1$)	3.7 ± 0.34	3.9 ± 0.4	4.4 ± 0.8	4.7 ± 0.9	0.89
Ridge ($\lambda = 0.01$)	4.4 ± 0.5	4.7 ± 0.7	5.7 ± 1.2	4.8 ± 0.9	0.91

Table 1: Mean and standard deviation, of the size of prediction intervals at coverage $q = 0.9$, with random data at $n = 100, d = 50$ generated from a Gaussian teacher. For all methods except exact LOO, values are averaged over 1000 test samples.

Problem	JI (Taylor-AMP)	JI (SCP)
Ridge ($\lambda = 0.01$)	0.93 ± 0.04	0.80 ± 0.12
Ridge ($\lambda = 0.1$)	0.95 ± 0.04	0.83 ± 0.1
Ridge ($\lambda = 1$)	0.98 ± 0.02	0.84 ± 0.04
Lasso ($\lambda = 0.01$)	0.90 ± 0.06	0.86 ± 0.11
Lasso ($\lambda = 0.1$)	0.92 ± 0.05	0.87 ± 0.09
Lasso ($\lambda = 1$)	0.97 ± 0.03	0.88 ± 0.08

Table 2: Jaccard index (JI) between exact LOO and Taylor-AMP and SCP for different estimators, with data generated from a Gaussian teacher, and $d = 50, n = 100$. We report the averages and standard deviation over 20 test samples.

values closer to 1 indicate more precise approximations. We report our findings in Table 2, where we evaluate the Jaccard index $\mathcal{J}\left(\mathcal{S}_{\text{fcp}}(\boldsymbol{x}), \mathcal{S}_{\text{Taylor-AMP}}(\boldsymbol{x})\right)$ and $\mathcal{J}\left(\mathcal{S}_{\text{fcp}}(\boldsymbol{x}), \mathcal{S}_{SCP}(\boldsymbol{x})\right)$. Taylor-AMP has a higher similarity to FCP than SCP, confirming that even though our method is approximate, it provides intervals that are very close to the exact ones even at moderate dimensions.

Computation speed – In Figure 1, we compare the time to compute $\mathcal{S}(\boldsymbol{x})$ for a single test sample $\boldsymbol{x}$, as a function of the dimension for a fixed sampling ratio $\alpha = {}^{n}/_{d}$. Our method provides a speed-up over exact LOO by more than two orders of magnitude, and allows us to quantify the uncertainty for dimensions about 10 times higher for the same amount of time. With the Taylor-AMP algorithm, we can readily treat problems of dimension 10^4. So far our numerical results show that our algorithm approximates precisely exact LOO, while being order of magnitudes faster. This allows to benchmark FCP against other methods in large dimensions, as we do in the following paragraphs.

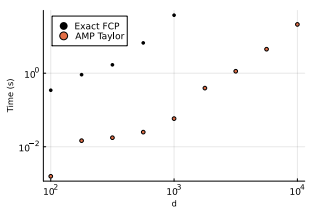

Figure 1: Computation time to produce a single prediction interval, for exact LOO and Taylor-AMP , for Lasso at $\lambda = 1$ and ${}^{n}/_{d} = 0.5$.

Comparison with Bayes posterior – We compare the prediction intervals of conformal prediction with those of the Bayes-optimal estimator as defined in Section 2.3. Recall that the Bayes-optimal estimator has the lowest generalisation error when the data-generating process is known. When the prior $p_{\boldsymbol{\theta}_\star}$ is Gaussian, the log-posterior exactly corresponds to Ridge regression with $\lambda = 1$. Likewise, for a Laplace prior on $\boldsymbol{\theta}_\star$, the log-posterior is exactly the empirical risk of Lasso, with $\lambda = 1$. In Table 3, we compare the average length of the prediction intervals provided by FCP with the highest density intervals of the Bayes posterior. Note that for a Gaussian prior, the posterior distribution is also Gaussian and can be easily sampled. However, this is not the case for a Laplace prior. In general, one would sample the posterior using Monte-Carlo methods. However, within our synthetic data setting, we can leverage the AMP algorithm 1 to sample the posterior [Clarté et al., 2023b]. AMP is much faster than costly Monte-Carlo sampling, while being exact in the high-dimensional limit. Lines in bold represent the matched settings where the minimized empirical risk matches the true log posterior. In these settings, FCP has almost optimal length, as it is very close to those of the Bayes-optimal estimator. On the other hand, when λ has a value that does not match the true prior, then the intervals obtained with Taylor-AMP are significantly larger than those of Bayes, for instance with $\lambda = 0.1$. Finally, we show in italic the theoretical predictions using Eq. (19) and observe a good match with the empirical values.

Comparison with split conformal prediction – In Table 1, we compare the length of the prediction intervals of Taylor-AMP with SCP described in 7, and to conformalized quantile regression (CQP) [Romano et al., 2019], where split conformal prediction is applied on two estimators of the quantile functions of the likelihood $p(y|\boldsymbol{x})$. We observe that as expected, our method provides tighter intervals while having the correct coverage.

Comparison on real data– In this section, we compare the performance of Taylor-AMP with other methods in the literature : *exact homotopy*[Lei, 2019], *approximate homotopy*[Ndiaye and Takeuchi, 2019] and the *Jackknife+*[Barber et al., 2019]. For a fair comparison, the experiments are done for the Lasso since Lei [2019] focuses on the Lasso and

Teacher	Regularization	Bayes	Taylor-AMP
Gaussian	L_2 ($\lambda = 0.1$)	4.4	4.8 ± 0.6 (*4.8*)
	$\mathbf{L_2}(\lambda = \mathbf{1.0})$		$\mathbf{4.4 \pm 0.4}$ (***4.4***)
	L_1 ($\lambda = 1.0$)		5.0 ± 1.2 (*4.6*)
Laplace	L_1 ($\lambda = 0.1$)	5.1	7.6 ± 2.1 (*6.0*)
	$\mathbf{L_1}(\lambda = \mathbf{1.0})$		$\mathbf{5.8 \pm 1.2}$ (***5.3***)
	L_2 ($\lambda = 1.0$)		5.2 ± 0.4 (*5.2*)

Table 3: Average and standard deviation of length of prediction intervals of FCP with Taylor-AMP , at $d = 250, n = 125$ compared with the Bayes optimal estimator. Measures are averaged over 1000 samples of both $\mathcal{D}$ and the single test sample. Bold lines correspond to the matched setting where the empirical risk corresponds to the log-posterior of the data-generating process. Values in italic are the theoretical predictions using state-evolution equations 19.

ElasticNet. Note however that Taylor-AMP is extendable to other empirical risks as we detail in Appendix E. We evaluate the methods on three datasets : the wine quality[Cortez et al., 2009], the Boston housing and the Riboflavin production rate[Bühlmann et al., 2014] datasets. We evaluate the coverage, the mean size of the prediction intervals and the computation time of the four methods. We observe that Taylor-AMP has the correct coverage and comparable sizes as Lei [2019] and Ndiaye and Takeuchi [2019]. Moreover, for the Riboflavin dataset, at $d = 4088$, approximate homotopy becomes overly conservative and is significantly slower than our method, while we perform similarly as Lei [2019]. Note that the Jackknife+ is overly conservative across all datasets. We refer the reader to Appendix D for more details on the datasets and the methods.

Beyond Ridge and Lasso regression– While the comparison Table 4 was only done for the Lasso, our method is very generic and is applicable to any generalized linear model whose loss and regularization are convex. For instance, one can apply AMP and Taylor-AMP to classification tasks, robust regression or quantile regression. We refer to Appendix E for more details.

5 DISCUSSION

In this paper, we introduce a method to accelerate the computations of full conformal prediction while guaranteeing confidence sets with the correct coverage. Our method leverages methods stemming from high-dimensional statistics literature, namely the approximate message passing (AMP) algorithm. Our numerical experiments on synthetic and real data show that the method has the potential to provide narrow confidence sets (with coverage guarantees) while reducing the computation time by almost three orders of magnitude compared to the baseline. Our method has a particular theoretical interest, as Taylor-AMP can be used to investigate more easily the properties of full conformal prediction in high dimensions by drastically speeding up the simulations. The proposed algorithm can leverage the fact that it is asymptotically exact on the synthetic Gaussian data and these data can thus be used as a benchmark for other speed-up methods in high-dimensions. The state-evolution equations of AMP can be used to compute exactly the size of the prediction intervals of FCP in this setting.

Possible extensions – While we only investigated conformal prediction for frequentist estimators, AMP can be used to sample from Bayesian posteriors more efficiently than Monte-Carlo methods. Our results could thus be extended to Bayesian conformal prediction, where the conformity scores are given by the predictive posterior [Fong and Holmes, 2021, Papadopoulos, 2024]. Moreover, one could improve Algorithm 1 to compute the prediction intervals of several samples simultaneously.

Limitations – One limitation of our work is the assumption weak dependence on every sample in Taylor-AMP . Further, while we show that our method is applicable to real data. The extension of our method to more complex algorithms of a similar kind such as VAMP Rangan et al. [2019], which would make our method applicable to a broader set of data, is left to future work. The code used to produce the figures can be found the following github repository: `github.com/SPOC-group/ConformalAmp.jl`.

Acknowledgements – We thank Bruno Loureiro and Florent Krzakala for valuable discussions. This research was supported by the NCCR MARVEL, a National Centre of Competence in Research, funded by the Swiss National Science Foundation (grant number 205602).

Dataset	Regularization	Method	Size	Time	Coverage
Wine	Lasso ($\lambda = 1$)	Taylor-AMP	2.49 ± 0.08	0.15	0.89 ± 0.03
		Approximate homotopy	2.6 ± 0.02	0.09	0.91 ± 0.03
		Exact homotopy	2.58 ± 0.02	0.001	0.9 ± 0.03
		Jackknife+	3.19 ± 0.04	0.002	0.94 ± 0.02
Boston	Lasso ($\lambda = 1$)	Taylor-AMP	1.52 ± 0.07	0.027	0.89 ± 0.03
		Approximate homotopy	1.5 ± 0.04	0.04	0.88 ± 0.04
		Exact homotopy	1.57 ± 0.05	5e-4	0.90 ± 0.03
		Jackknife+	2.17 ± 0.10	2e-3	0.95 ± 0.02
Riboflavin	Lasso ($\lambda = 0.25$)	Taylor-AMP	2.2 ± 0.3	0.4	0.88 ± 0.07
		Approximate homotopy	3.6 ± 0.25	2.5	0.96 ± 0.04
		Exact homotopy	2.3 ± 0.16	0.61	0.9 ± 0.09
		Jackknife+	4.1 ± 0.36	0.03	0.95 ± 0.06

Table 4: Comparison of Taylor-AMP with exact homotopy, approximate homotopy and Jackknife+ on the Boston and Riboflavin datasets. We show the mean and standard deviation over 20 train / test splits.

References

Anastasios N. Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification, 2022.

Anastasios N Angelopoulos, Stephen Bates, Emmanuel J Candès, Michael I Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *arXiv preprint arXiv:2110.01052*, 2021.

Gabriel Arpino, Xiaoqi Liu, and Ramji Venkataramanan. Inferring change points in high-dimensional linear regression via approximate message passing. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 1841–1864. PMLR, 21–27 Jul 2024.

Yu Bai, Song Mei, Haiquan Wang, and Caiming Xiong. Don't just blame over-parametrization for overconfidence: Theoretical analysis of calibration in binary classification. In *International Conference on Machine Learning*, 2021a.

Yu Bai, Song Mei, Huan Wang, and Caiming Xiong. Understanding the under-coverage bias in uncertainty estimation. In *Neural Information Processing Systems*, 2021b.

Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. Predictive inference with the jackknife+. *The Annals of Statistics*, 2019.

Mohsen Bayati and Andrea Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *2010 IEEE International Symposium on Information Theory*, pages 1528–1532, 2010.

Peter Bühlmann, Markus Kalisch, and Lukas Meier. High-dimensional statistics with a view toward applications in biology. 2014.

Giovanni Cherubin, Konstantinos Chatzikokolakis, and Martin Jaggi. Exact optimization of conformal predictors via incremental and decremental learning, 2021.

Lucas Clarté, Bruno Loureiro, Florent Krzakala, and Lenka Zdeborova. On double-descent in uncertainty quantification in overparametrized models. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pages 7089–7125. PMLR, 2023a.

Lucas Clarté, Bruno Loureiro, Florent Krzakala, and Lenka Zdeborová. Theoretical characterization of uncertainty in high-dimensional linear classification. *Machine Learning: Science and Technology*, 4(2):025029, jun 2023b.

Lucas Clarté, Adrien Vandenbroucque, Guillaume Dalle, Bruno Loureiro, Florent Krzakala, and Lenka Zdeborová. Analysis of bootstrap and subsampling in high-dimensional regularized regression, 2024.

P. Cortez, Antonio Luíz Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decis. Support Syst.*, 47:547–553, 2009.

A. C. Davison and D. V. Hinkley. *Bootstrap Methods and their Application*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1997.

Al Depope, Marco Mondelli, and Matthew R. Robinson. Inference of genetic effects via approximate message passing. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13151–13155, 2024.

David L. Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45): 18914–18919, November 2009. ISSN 1091-6490.

Ahmed El Alaoui, Andrea Montanari, and Mark Sellke. Sampling from the sherrington-kirkpatrick gibbs measure via algorithmic stochastic localization. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 323–334. IEEE, 2022.

Edwin Fong and Chris C Holmes. Conformal bayesian computation. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 18268–18279. Curran Associates, Inc., 2021.

Lei Jing, G'Sell Max, Rinaldo Alessandro, J. Tibshirani Ryan, and Wasserman Larry. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

Noureddine El Karoui and Elizabeth Purdom. Can we trust the bootstrap in high-dimensions? the case of linear models. *J. Mach. Learn. Res.*, 19:5:1–5:66, 2018.

Jing Lei. Fast exact conformalization of the lasso using piecewise linear homotopy. *Biometrika*, 106(4):pp. 749–764, 2019.

Tengyuan Liang and Pragya Sur. A precise high-dimensional asymptotic theory for boosting and minimum-l1-norm interpolated classifiers. *The Annals of Statistics*, 50(3), 2022.

Bruno Loureiro, Cédric Gerbelot, Maria Refinetti, Gabriele Sicuro, and Florent Krzakala. Fluctuations, bias, variance and ensemble of learners: exact asymptotics for convex losses in high-dimension. *Journal of Statistical Mechanics: Theory and Experiment*, 2023, 2023.

Javier Abad Martinez, Umang Bhatt, Adrian Weller, and Giovanni Cherubin. Approximating full conformal prediction at scale via influence functions. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. ISBN 978-1-57735-880-0.

Charles Millard, Aaron T. Hess, Boris Mailhé, and Jared Tanner. An approximate message passing algorithm for rapid parameter-free compressed sensing MRI. In *IEEE International Conference on Image Processing, ICIP 2020, Abu Dhabi, United Arab Emirates, October 25-28, 2020*, pages 91–95. IEEE, 2020.

Marc Mézard and Andrea Montanari. *Information, Physics, and Computation*. Oxford University Press, 01 2009. ISBN 9780198570837.

Eugene Ndiaye and Ichiro Takeuchi. Computing full conformal prediction set with approximate homotopy. volume 32. Curran Associates, Inc., 2019.

Jeremy Nixon and Dustin Tran. Why aren't bootstrapped neural networks better? Neurips 2020 ICBINB Workshop, 2020.

Tomoyuki Obuchi and Yoshiyuki Kabashima. Cross validation in lasso and its acceleration. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(5):053304, may 2016.

Harris Papadopoulos. Guaranteed coverage prediction intervals with gaussian process regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12): 9072–9083, 2024.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. pages 345–356, 2002.

Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988. ISBN 1558604790.

M. H. Quenouille. Notes on bias in estimation. *Biometrika*, 43(3/4):353–360, 1956.

Sundeep Rangan, Philip Schniter, and Alyson K. Fletcher. Vector approximate message passing. *IEEE Transactions on Information Theory*, 65(10):6664–6684, 2019.

Yaniv Romano, Evan Patterson, and Emmanuel Candes. Conformalized quantile regression. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction, 2007.

Pragya Sur and Emmanuel J. Candès. A modern maximum-likelihood theory for high-dimensional logistic regression. *Proceedings of the National Academy of Sciences*, 116 (29):14516–14525, July 2019. ISSN 1091-6490.

Takashi Takahashi. A replica analysis of under-bagging, 2024.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. 01 2005.

Lenka Zdeborová and Florent Krzakala. Statistical physics of inference: thresholds and algorithms. *Advances in Physics*, 65(5):453–552, August 2016. ISSN 1460-6976.

Ji Zhu, Saharon Rosset, Hui Zou, and Trevor Hastie. Multi-class adaboost. *Statistics and its interface*, 2, 02 2006.

A APPROXIMATE MESSAGE PASSING TO APPROXIMATE LEAVE-ONE-OUT RESIDUALS

A.1 INTRODUCTION OF RELAXED-BELIEF PROPAGATION AND APPROXIMATE MESSAGE PASSING

In this section, we explain how AMP can be used to compute the leave-one-out residuals used in Eq. (4). The naive way to compute these residuals is to fit the leave-one-out estimators $\hat{\boldsymbol{\theta}}_{-i}(y)$ for each sample $1 \leqslant i \leqslant n$ and each possible label y, which requires $n \times |\mathcal{Y}|$ fits, with $\mathcal{Y}$ the set of candidate labels, typically a discretization of $\mathbb{R}$. We will first see that AMP can be used to compute all the $\hat{\boldsymbol{\theta}}_{-i}$

To introduce AMP, we first consider the following problem. Consider a dataset $\mathcal{D} = (\boldsymbol{x}_i, y_i)_{i=1}^n$ of size n. Assume that the data is generated from the model (17), where the input $\boldsymbol{x}_i \in \mathbb{R}^d$ are sampled according to $\mathcal{N}(\mathbf{0}, {}^{I_d}/_d)$, and the labels are generated from a *teacher* as $y \sim p(y|{\boldsymbol{\theta}_\star}^\top \boldsymbol{x})$. Our goal is to sample the following distribution

$$p(\boldsymbol{\theta}) = \frac{1}{Z} \prod_{i=1}^{n} P_{out}\left(y_i|\boldsymbol{\theta}^\top \boldsymbol{x}_i\right) \prod_{\mu=1}^{d} P_\theta(\boldsymbol{\theta}_\mu) \tag{22}$$

The empirical risk minimization problem (2) introduced in Section 2 is a particular instance of Eq. (22) where

$$P_{out}(y|z) \propto e^{-\beta \ell(y,z)}, \qquad P_\theta(z) \propto e^{-\beta r(z)} \tag{23}$$

in the limit $\beta \to \infty$. The starting point of approximate message passing is the writing of the belief-propagation algorithm for the graph associated with Eq. (22), where the variable-nodes of the graph are the coordinates $\boldsymbol{\theta}_\mu$ and the factor nodes, representing the interaction between the variable-nodes, are the observations y_i. The message passing consists in iterating messages $m_{\mu\to i}$ from variable to factor-nodes and $m_{i\to\mu}$ from factor to variable-nodes. These messages read

$$m_{\mu\to i}(\theta_\mu) = \frac{1}{z_{i\to\mu}} P_\theta(\theta_\mu) \prod_{j\neq i} m_{j\to\mu}(\theta_\mu) \tag{24}$$

$$m_{i\to\mu}(\theta_\mu) = \frac{1}{z_{\mu\to i}} \int \prod_{\nu\neq\mu} \mathrm{d}\theta_\nu m_{\nu\to i} P_{out}\left(y_i|\sum_\nu \boldsymbol{x}_{i\nu}\theta_\nu\right) \tag{25}$$

This messages give access to the distribution $p\left(\boldsymbol{\theta}\right)$ and in particular this marginals : indeed, the marginal distribution $p(\theta_\mu)$ is given by

$$p(\theta_\mu) = \frac{1}{z_\mu} P_\theta(\theta_\mu) \prod_{i=1}^{n} m_{i\to\mu}(\theta_\mu) \tag{26}$$

where z_μ is a normalization constant. Iterating Eq. (25) is not tractable, especially in high-dimensions as it involves $(d-1)$ integrals to update each $m_{i\to\mu}$. To make these equations tractable, one can use relaxed-Belief Propagation (rBP), which relies on the central limit theorem and the projection of the messages on their first two moments. We thus define the *cavity mean* $\hat{\theta}_{\mu\to i}$ and *cavity variance* $\hat{v}_{\mu\to i}$ as

$$\hat{\theta}_{\mu\to i} = \int \mathrm{d}\theta_\mu \theta_\mu m_{\mu\to i}(\theta_\mu) \tag{27}$$

$$\hat{v}_{\mu\to i} = \int \mathrm{d}\theta_\mu \theta_\mu^2 m_{\mu\to i}(\theta_\mu) - \hat{\theta}_{\mu\to i}^2 \tag{28}$$

In particular, the vector $\left(\hat{\theta}_{\mu\to i}\right)_{\mu=1}^d$ represents the mean of the marginals of distribution (22) in the absence of the i-th sample. In the context of empirical risk minimization, this is exactly the leave-one-out estimator $\hat{\boldsymbol{\theta}}_{-i}$ defined as

$$\hat{\boldsymbol{\theta}}_{-i} = \arg\min_{\boldsymbol{\theta}} \sum_{j\neq i} \ell\left(y_j, \boldsymbol{\theta}^\top \boldsymbol{x}_j\right) + \sum_{\mu=1}^{d} r(\theta_\mu) \tag{29}$$

Our goal is thus to compute efficiently the cavity means and use them to compute the leave-one-out residuals.

rBP The main idea behind rBP is to iteratively compute the cavity means and variances, to obtain the desired marginal mean and variance of $\boldsymbol{\theta}$. We define $\omega_{i\to\mu}, V_{i\to\mu}$ the mean and variance of the messages $m_{i\to\mu}$ and $\hat{\theta}_{\mu\to i}, \hat{v}_{\mu\to i}$ the mean and variance of $m_{\mu\to i}$.

We detail rBP in Algorithm 3, and refer to [Zdeborová and Krzakala, 2016, Chapter VI, Section C] for a detailed explanation of the algorithm. In particular, the algorithm makes use of the *channel* and *denoising* functions g_{out} and f_w functions, defined respectively as

$$g_{\text{out}}(y,\omega,V) = \frac{\partial \log \mathcal{Z}_y(y,\omega,V)}{\partial \omega}, \qquad \mathcal{Z}_y(y,\omega,V) = \int \mathrm{d}z P_{out}(y|z) e^{-\frac{1}{2V}(z-\omega)^2} \tag{30}$$

and

$$f_w(b,A) = \frac{\partial \log \mathcal{Z}_w(b,A)}{\partial b}, \qquad \mathcal{Z}_w(b,A) = \int \mathrm{d}x P_\theta(x) e^{bx - \frac{A}{2}x^2} \tag{31}$$

In the case of empirical risk minization (2), using the prior and likelihood from Eq. (23) into the definitions(30) and (31) and taking the limit $\beta \to \infty$ yields Equation (11).

From rBP to AMP Note that in rBP, we iterate over $n \times d$ means and variances $\omega_{i\to\mu}, V_{i\to\mu}, \hat{\theta}_{\mu\to i}, \hat{v}_{\mu\to i}$, which scales quadratically with the dimension in the high-dimensional limit where $n, d \to \infty$ with a constant sampling ratio $n/d = \alpha$. However, a key observation is that the quantities $\hat{\theta}_{\mu\to i}, \hat{v}_{\mu\to i}$ only weakly depend on μ, and similarly $\omega_{i\to\mu}, V_{i\to\mu}$ weakly depend on μ. Hence, let us define

$$\begin{cases} \omega_i &= \sum_\mu \boldsymbol{x}_{i\mu} \hat{\theta}_{\mu\to i} \\ V_i &= \sum_\mu \boldsymbol{x}_{i\mu}^2 \hat{v}_{\mu\to i} \end{cases}, \qquad \begin{cases} A_\mu &= -\sum_{i=1}^n \partial_\omega g_{out}(y_i, \omega_i, V_i) \boldsymbol{x}_{i\mu}^2 \\ b_\mu &= \sum_{i=1}^n g_{out}(y_i, \omega_{i\to\mu}, V_{i\to\mu}) \boldsymbol{x}_{i\mu} \end{cases} \tag{32}$$

note that for all μ and all i, in the high-dimensonal limit considered here we have

$$\omega_i = \omega_{i\to\mu} + \boldsymbol{x}_{i\mu}\hat{\theta}_{\mu\to i} = \omega_{i\to\mu} + O(1/\sqrt{n}) \tag{33}$$

$$V_i = V_{i\to\mu} + \boldsymbol{x}_{i\mu}^2 \hat{v}_{\mu\to i} = V_{i\to\mu} + O(1/n) \tag{34}$$

As a consequence, we have for all μ and all i

$$A_\mu = -\sum_{j=1}^n \boldsymbol{x}_{j\mu}^2 \partial_\omega g_{out}(y_j, \omega_j, V_j) = \sum_{j=1}^n \boldsymbol{x}_{j\mu}^2 \left[\partial_\omega g_{out}(y_j, \omega_{j\to\mu}, V_{j\to\mu}) + O(1/\sqrt{n})\right] \tag{35}$$

$$= -\sum_{j=1}^n \boldsymbol{x}_{j\mu}^2 \partial_\omega g_{out}(y_j, \omega_{j\to\mu}, V_{j\to\mu}) + O(1/\sqrt{n}) \tag{36}$$

$$= -\sum_{j\neq i}^n \boldsymbol{x}_{j\mu}^2 \partial_\omega g_{out}(y_j, \omega_{j\to\mu}, V_{j\to\mu}) + O(1/\sqrt{n}) \tag{37}$$

$$= -A_{\mu\to i} + O(1/\sqrt{n}) \tag{38}$$

Similarly, we get

$$b_\mu = b_{\mu\to i} + O(1/\sqrt{n}) \tag{39}$$

So that one can simply compute the estimator $\boldsymbol{\theta} = f_w(\boldsymbol{b}, \boldsymbol{A})$. The challenge is to compute the vectors $\boldsymbol{\omega}, \boldsymbol{V}, \boldsymbol{b}$. To do so, we note that

$$g_{\text{out}}(y_i, \omega_{i\to\mu}, V_{i\to\mu}) = g_{\text{out}}(y_i, \omega_i, V_i) - \boldsymbol{x}_{i\mu}\hat{\theta}_{\mu\to i}\partial_\omega g_{\text{out}}(y_i, \omega_{i\to\mu}, V_{i\to\mu}) + O(1/n) \tag{40}$$

$$\tag{41}$$

such that

$$b_\mu = \sum_{i=1}^n \boldsymbol{x}_{i\mu} g_{\text{out}}(y_i, \omega_i, V_i) - \sum_i \boldsymbol{x}_{i\mu}^2 \hat{\theta}_\mu \partial_\omega g_{\text{out}}(y_i, \omega_i, V_i) + O(1/\sqrt{n}) \tag{42}$$

$$\tag{43}$$

Moreover,

$$\omega_i = \sum_{\mu=1}^{d} \boldsymbol{x}_{i\mu}\hat{\theta}_{\mu\to i} = \sum_{\mu} \boldsymbol{x}_{i\mu}\left(\hat{\theta}_\mu - \boldsymbol{x}_{i\mu}v_\mu g_{\text{out}}\left(y_i, \omega_i, V_i\right)\right) + O(1/n) \tag{44}$$

$$\tag{45}$$

These iterative equations are, in the leading order, the same as those shown in Algorithm 1. In the high-dimensional regime, these iteratives coincide with rBP. Going from rBP to AMP, we have reduced the number of variables to iterate on from $O(n \times d)$ to $O(n + d)$, and can still recover the marginal distribution by

$$\hat{\theta}_\mu = f_w\left(b_\mu, A_\mu\right) \tag{46}$$

A.2 RECOVERING THE LEAVE-ONE-OUT ESTIMATORS FROM AMP

For each sample i, computing the leave-one-out estimator $\hat{\boldsymbol{\theta}}_{-i}$ means computing the marginals of the distribution

$$p(\boldsymbol{\theta}) = \frac{1}{Z}\prod_{j\neq i} P_{out}\left(y_j|\boldsymbol{\theta}^\top \boldsymbol{x}_j\right)\prod_{\mu=1}^{d} P_\theta\left(\boldsymbol{\theta}_\mu\right) \tag{47}$$

with P_{out} and P_θ defined in Eq. (23) and where the sample $(\boldsymbol{x}_i, y_i)$ is removed from the data. Our method leverages the fact that these marginals are computed iteratively by relaxed-BP and stored in the variables $\hat{\theta}_{\mu\to i}$. Indeed, each $\hat{\theta}_{\mu\to i}$ stores the posterior mean of θ_{mu} when the interaction node i is removed from the graph associated to Eq. (22), which corresponds exactly to the distribution of Eq. (47). While rBP explicitly computes these quantities, its computational complexity makes it unusable. Instead, we will recover these estimators from AMP. Indeed, at the leading order we have :

$$\hat{\theta}_{\mu\to i} = f_w\left(b_{\mu\to i}, A_{\mu\to i}\right) = f_w\left(b_{\mu\to i}, A_\mu\right) + O\left(1/n\right) \tag{48}$$

$$= f_w\left(b_\mu, A_\mu\right) - b_{i\to\mu}\partial_b f_w\left(b_\mu, A_\mu\right) + O\left(1/n\right) = \hat{\boldsymbol{\theta}}_\mu - g_{\text{out}}(y_i, \omega_i, V_i)\boldsymbol{x}_{i\mu}\hat{\boldsymbol{v}}_\mu + O\left(1/n\right) \tag{49}$$

The expression on the right-hand side corresponds to the approximation of the leave-one-out estimators $\hat{\boldsymbol{\theta}}_{-i,\text{gamp}}$ used in Algorithm 1.

Convergence of the leave-one-out residuals in high-dimensions Under the assumptions (17), we see from Eq. (49) thatin the high-dimensional limit the leave-one-out estimators computed by AMP will converge to the exact ones at a $O(1/n)$ rate. As such, for a given test sample $\boldsymbol{x}, y$ the approximated residuals $y - \boldsymbol{x}^\top\boldsymbol{\theta}_{-i,\text{gamp}}$ will converge to $y - \boldsymbol{x}^\top\boldsymbol{\theta}_{-i}$ at a $O\left(1/\sqrt{n}\right)$ rate. This implies that asymptotically the prediction intervals built using the AMP leave-one-out converge to the prediction intervals with the exact residuals.

Applying AMP without Gaussian assumptions We thus see that from AMP, we get an approximation of the leave-one-out estimator that can be used to compute the residuals in Eq. (4). The derivations performed in this section were done under the assumption that the input data are Gaussian with i.i.d. covariance and $1/d$ variance. However, AMP can be applied on any data, with no guarantee a priori on its performance.

B DERIVATION OT TAYLOR-AMP

In this section, we derive the Taylor-AMP algorithm. Our starting point is AMP, derived in Appendix A. In what follow, we consider a dataset $\mathcal{D}$ of size $n + 1$ to stay consistent with the notation of the main text. Our goal is to compute the variation of the $\hat{\boldsymbol{\theta}}_{-i}$ to the first order with respect to the last label y_{n+1}. To this end, we will write the vectors defined in AMP $\hat{\boldsymbol{\theta}}(y), \hat{\boldsymbol{v}}(y), \boldsymbol{g}(y), \partial\boldsymbol{g}(y), \boldsymbol{b}(y), \boldsymbol{A}(y), \boldsymbol{\omega}(y), \boldsymbol{V}(y)$ as functions of $y_{n+1} = y$

For the sake of conciseness, let us define the vector

$$\Omega\left(y\right) = \left(\hat{\boldsymbol{\theta}}(y), \hat{\boldsymbol{v}}(y), \boldsymbol{\omega}(y), \boldsymbol{V}(y), \boldsymbol{g}(y), \partial\boldsymbol{g}(y), \boldsymbol{b}(y), \boldsymbol{A}(y)\right) \in \mathbb{R}^{4\times(d+n)} \tag{56}$$

Algorithm 3 relaxed-Belief Propagation

repeat

Input: Dataset $\mathcal{D} = (\boldsymbol{x}_i, y_i)_{i=1}^n$

$$\begin{cases} V_{i\to\mu}^t & = \sum_{\nu\neq\mu} \boldsymbol{x}_{i\mu}^2 v_{\nu\to i}^{t-1} \\ \omega_{i\to\mu}^t & = \sum_{\nu\neq\mu} \boldsymbol{x}_{i\mu} \hat{\theta}_{\nu\to i}^{t-1} \end{cases} \tag{50}$$

$$\begin{cases} A_{\mu\to i}^t & = -\sum_{j\neq i} \partial_\omega g_{\text{out}}\left(y_j, \omega_{j\to\mu}^t, V_{j\to\mu}\right) \boldsymbol{x}_{j\mu}^2 \\ b_{\mu\to i}^t & = \sum_{j\neq i} g_{\text{out}}\left(y_j, \omega_{j\to\mu}^t, V_{j\to\mu}\right) \boldsymbol{x}_{j\mu} \end{cases} \tag{51}$$

$$\hat{\theta}_{\mu\to i}^t = f_w\left(b_{\mu\to i}^t, A_{\mu\to i}^t\right) \tag{52}$$

$$\hat{v}_{\mu\to i}^t = \partial_b f_w\left(b_{\mu\to i}^t, A_{\mu\to i}^t\right) \tag{53}$$

until Convergence of $\hat{\theta}_{\mu\to i}, \hat{v}_{\mu\to i}$

Return $\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{v}}$ such that :

$$\hat{\theta}_\mu = f_w\left(\sum_i b_{\mu\to i}, \sum_i A_{\mu\to i}\right) \tag{54}$$

$$\hat{v}_\mu = \partial_b f_w\left(\sum_i b_{\mu\to i}, \sum_i A_{\mu\to i}\right) \tag{55}$$

Then, $\Omega(y)$ is the fixed point of the equation

$$\Omega(y) = \boldsymbol{f}_{\text{gamp}}(\Omega(y), y)$$

where the function $\boldsymbol{f}_{\text{gamp}}(\Omega) = \left(f_{\text{gamp}}^{\hat{\boldsymbol{\theta}}}, f_{\text{gamp}}^{\hat{\boldsymbol{v}}}, f_{\text{gamp}}^{\boldsymbol{\omega}}, f_{\text{gamp}}^{\boldsymbol{V}}, f_{\text{gamp}}^{\boldsymbol{g}}, f_{\text{gamp}}^{\partial \boldsymbol{g}}, f_{\text{gamp}}^{\boldsymbol{b}}, f_{\text{gamp}}^{\partial \boldsymbol{A}}\right)$ is defined as

$$\begin{cases} \boldsymbol{f}_{\text{gamp}}^{\hat{\boldsymbol{\theta}}} & = f_w(\boldsymbol{b}, \boldsymbol{A}) \\ \boldsymbol{f}_{\text{gamp}}^{\hat{\boldsymbol{v}}} & = \partial_b f_w(\boldsymbol{b}, \boldsymbol{A}) \\ \boldsymbol{f}_{\text{gamp}}^{\boldsymbol{\omega}} & = X\hat{\boldsymbol{\theta}} - \boldsymbol{V} \odot \boldsymbol{g} \\ \boldsymbol{f}_{\text{gamp}}^{\boldsymbol{V}} & = X^2 \hat{\boldsymbol{v}} \\ \boldsymbol{f}_{\text{gamp}}^{\boldsymbol{g}} & = g_{\text{out}}\left(\boldsymbol{y}, \boldsymbol{\omega}, \boldsymbol{V}\right) \\ \boldsymbol{f}_{\text{gamp}}^{\partial \boldsymbol{g}} & = \partial_\omega g_{\text{out}}\left(\boldsymbol{y}, \boldsymbol{\omega}, \boldsymbol{V}\right) \\ \boldsymbol{f}_{\text{gamp}}^{\boldsymbol{b}} & = X^\top \boldsymbol{g} + \boldsymbol{A} \odot \hat{\boldsymbol{\theta}} \\ \boldsymbol{f}_{\text{gamp}}^{\partial \boldsymbol{A}} & = -X^{2\top} \partial \boldsymbol{g} \end{cases} \tag{57}$$

Equivalently, we have $\Omega(y) - \boldsymbol{f}_{\text{gamp}}(\Omega(y), y) = \mathbf{0}$. Under the assumption that the function $\Omega(y)$ is differentiable, one can use the implicit function theorem around a value $\hat{y}$ to write

$$\frac{\partial \Omega}{\partial y}(\hat{y}) = \left(\mathbf{I} - Jac\left(\boldsymbol{f}_{\text{gamp}}\right)\right)^{-1} \frac{\partial \boldsymbol{f}_{\text{gamp}}}{\partial y}(\hat{y}) \tag{58}$$

$$\Leftrightarrow \frac{\partial \Omega}{\partial y}(\hat{y}) = Jac\left(\boldsymbol{f}_{\text{gamp}}\right)\left(\frac{\partial \Omega}{\partial y}(\hat{y})\right) + \frac{\partial \boldsymbol{f}_{\text{gamp}}}{\partial y}(\hat{y}) \tag{59}$$

From the last equality we find that we can compute the derivative $\frac{\partial \Omega}{\partial y}(\hat{y})$ by iterating the following system of linear equations over a vector $\Delta\Omega^t$:

$$\Delta\Omega^{t+1} = Jac\left(\boldsymbol{f}_{\text{gamp}}\right)\left(\Delta\Omega^t\right) + \frac{\partial \boldsymbol{f}_{\text{gamp}}}{\partial y}(\hat{y}) \tag{60}$$

The jacobian of the function $\boldsymbol{f}_{\text{gamp}}$ is written

$$\begin{cases} Jacf^{\hat{\boldsymbol{\theta}}}_{\text{amp}} &= (\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\partial_b f_w(\boldsymbol{b},\boldsymbol{A}),\partial_A f_w(\boldsymbol{b},\boldsymbol{A})) \\ Jacf^{\hat{\boldsymbol{v}}}_{\text{amp}} &= (\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},\partial_b\partial_b f_w(\boldsymbol{b},\boldsymbol{A}),\partial_A\partial_b f_w(\boldsymbol{b},\boldsymbol{A})) \\ Jacf^{\boldsymbol{\omega}}_{\text{amp}} &= (X,\mathbf{0},\mathbf{0},-Diag(\boldsymbol{g}),-Diag(\boldsymbol{V}),\mathbf{0},\mathbf{0},\mathbf{0}) \\ Jacf^{\boldsymbol{V}}_{\text{amp}} &= \big(\mathbf{0},X^2,\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0}\big) \\ Jacf^{\boldsymbol{g}}_{\text{amp}} &= (\mathbf{0},\mathbf{0},\partial_\omega \boldsymbol{g},\partial_V \boldsymbol{g},\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0}) \\ Jacf^{\partial \boldsymbol{g}}_{\text{amp}} &= (\mathbf{0},\mathbf{0},\partial_\omega\partial_\omega \boldsymbol{g},\partial_V\partial_\omega \boldsymbol{g},\mathbf{0},\mathbf{0}) \\ Jacf^{\boldsymbol{b}}_{\text{amp}} &= \big(Diag(\boldsymbol{A}),\mathbf{0},\mathbf{0},\mathbf{0},X^\top,\mathbf{0},\mathbf{0},Diag(\boldsymbol{w})\big) \\ Jacf^{\boldsymbol{A}}_{\text{amp}} &= \big(\mathbf{0},\mathbf{0},\mathbf{0},\mathbf{0},-X^{2\top},\mathbf{0},\mathbf{0},\mathbf{0}\big) \end{cases} \tag{61}$$

and the derivative $\frac{\partial \boldsymbol{f}_{\text{gamp}}}{\partial y}$ with respect to the last label is

$$\begin{cases} \partial_y f^{\boldsymbol{\theta}}_{\text{amp}} &= \mathbf{0} \\ \partial_y f^{\boldsymbol{v}}_{\text{amp}} &= \mathbf{0} \\ \partial_y f^{\boldsymbol{\omega}}_{\text{amp}} &= \mathbf{0} \\ \partial_y f^{\boldsymbol{V}}_{\text{amp}} &= \mathbf{0} \\ \partial_y f^{\boldsymbol{g}}_{\text{amp}} &= (0,\cdots,0,\partial_y g(y_n,\omega_n,V_n)) \\ \partial_y f^{\partial \boldsymbol{g}}_{\text{amp}} &= (0,\cdots,0,\partial_y\partial_\omega g(y_n,\omega_n,V_n)) \\ \partial_y f^{\boldsymbol{b}}_{\text{amp}} &= \mathbf{0} \\ \partial_y f^{\boldsymbol{A}}_{\text{amp}} &= \mathbf{0} \end{cases}$$

When writing Equation (60) with the expression of the Jacobian of Equation (61), one obtains the iterations of Taylor-AMP in Algorithm 2.

B.1 JUSTIFICATION OF TAYLOR-AMP

As stated in the previous subsection, Taylor-AMP is based on the assumption that the function $y \to \Omega\,)y)$ is differentiable. Our underlying assumption behind Taylor-AMP is that the leave-one-out residuals only weakly depend on the last label in high-dimensions. We numerically justify this assumption in Fig. 2. In this Figure, we compare the leave-one-out residuals obtained by computing the estimators $\hat{\boldsymbol{\theta}}_{-i}$ exactly and with Taylor-AMP for different settings. To do so, we sample a dataset $\mathcal{D}$ at random. We use Algorithm 1 and Algorithm 2 to compute the $\hat{\boldsymbol{\theta}}_{-i,\text{gamp}}(y_n)$ and $\Delta\hat{\boldsymbol{\theta}}_{-i,\text{gamp}}(y)$ as prescribed above. Then, we change the last label $y_n \to y_n + \delta y$ with $\delta y = 5$. After this change we compute the leave-one-estimators exactly $\hat{\boldsymbol{\theta}}_{-i}(y_n + \delta y)$ and use our linear approximation $\hat{\boldsymbol{\theta}}_{-i,\text{gamp}}(y + \delta y) = \hat{\boldsymbol{\theta}}_{-i}(y) + \delta y \Delta\hat{\boldsymbol{\theta}}_{-i.\text{gamp}}(y)$. We then compare $\hat{\boldsymbol{\theta}}_{-i}(y_n + \delta y)^\top \boldsymbol{x}_i$ and our approximation $\hat{\boldsymbol{\theta}}_{-i,\text{gamp}}(y_n + \delta y)^\top \boldsymbol{x}_i$ that is used to compute our conformity scores. As we observe in the figure, at high dimensions $d = 1000$, our approximations are very close to the true values, meaning that Taylor-AMP will accurately estimate the scores (hence the prediction intervals) of FCP.

We note however from the lower-left plot that at moderate dimension, Taylor-AMP does not precisely approximates the leave-one-out residuals for the LASSO, which partly explains the mediocre results obtained by Taylor-AMP on real data in Table 4 in the main.

C COVERAGE GUARANTEE FOR AMP

First, we show that AMP is symmetric : indeed, consider a permutation $s : [1,n] \to [1,n]$ and S the corresponding permutation matrix defined as $S_{ij} = \delta\,(j = s(i))$. Then, consider running AMP on the permutated data $\tilde{X} = SX$ and labels $\tilde{y} = SY$. At each iteration t, the channel vectors $\tilde{\boldsymbol{g}}^t, \tilde{\partial \boldsymbol{g}}^t$ $\tilde{\boldsymbol{g}}^t = S\boldsymbol{g}^t$ and $\tilde{\partial \boldsymbol{g}}^t = S\partial \boldsymbol{g}^t$. Then, the vectors $\boldsymbol{b}^t, \boldsymbol{A}^t$ now become

$$\begin{cases} \tilde{\boldsymbol{A}}^t &= -\tilde{X^{2\top}}\tilde{\partial \boldsymbol{g}}^t = -X^{2\top}S^T S\boldsymbol{g}^t = \boldsymbol{A}^t \\ \tilde{\boldsymbol{b}}^t &= \tilde{X^\top}\tilde{\boldsymbol{g}}^t + \tilde{\boldsymbol{A}}^t \otimes \hat{\boldsymbol{\theta}}^t = X^\top S^T S\boldsymbol{g}^t + \tilde{\boldsymbol{A}}^t \otimes \hat{\boldsymbol{\theta}}^t = \boldsymbol{b}^t \end{cases} \tag{62}$$

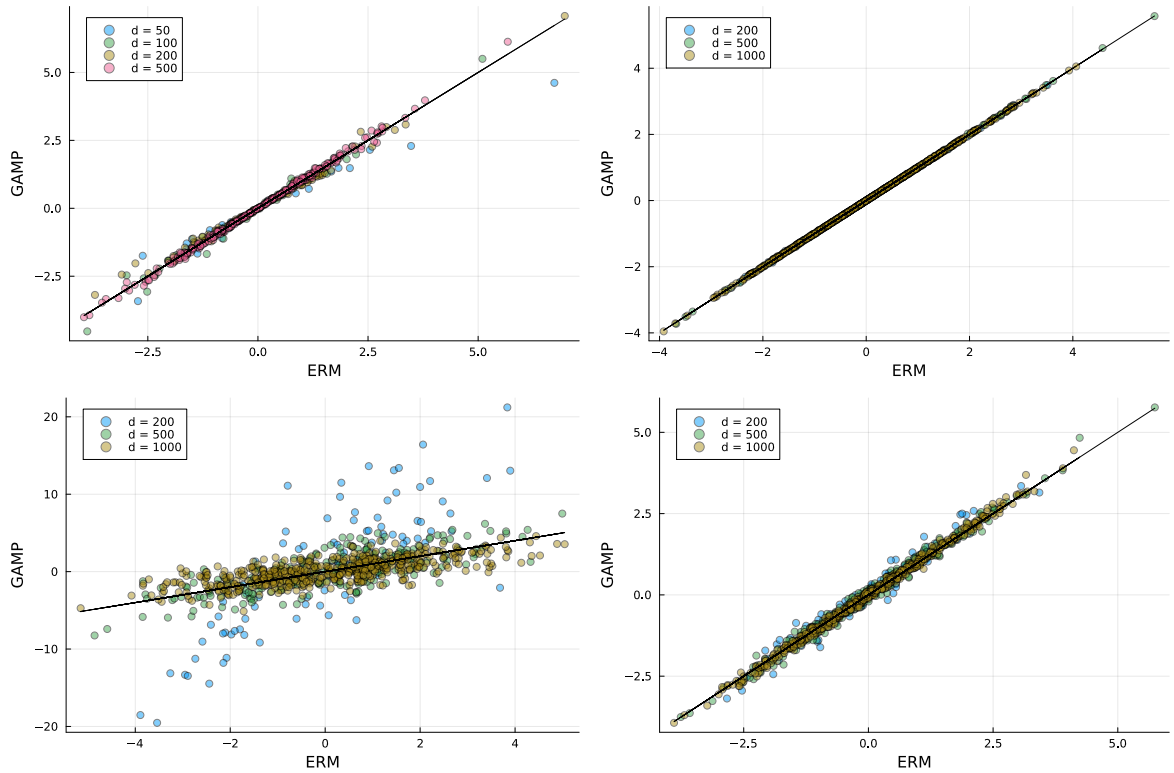

Figure 2: Comparison of the leave-one-out estimators computed exactly by solving Eq. (3) and by Taylor-AMP , for Ridge (top row) and Lasso (bottom row), as $\lambda = 0.01$ (left column) and $\lambda = 1$ (right column). All plots are at $n/d = 0.5$

and by recursion we deduce that the estimator of AMP $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{v}})$ given after convergence is invariant under permutation. Then, the scores computed from Eq. (13) are symmetric. Then, under the assumption that the data $(\boldsymbol{x}_i, y_i)$ is exchangeable, we obtain Property 1 : in expectation over the training and test data

$$\mathbb{P}_{\mathcal{D},\boldsymbol{x}}\left(y \in \mathcal{S}(\boldsymbol{x})\right) \geqslant 1 - \kappa \tag{63}$$

D DETAILS ON REAL DATASETS

In this section, we provide details on the datasets used in Table 4. We use :

1. The wine quality dataset [Cortez et al., 2009], containing 1143 samples at dimension 11, containing a rating of the wine quality on a 1-5 scale as a function of different physical quantities. In our experiments, we split the data into a training and test sets with a 90% / 10% proportion.
2. The Boston housing dataset containing 506 samples at dimension 14, with a training / test split of 80 % / 20 %.
3. The Riboflavin dataset Bühlmann et al. [2014] of 71 samples at dimension 4088

All datasets, with dimension noted as d, where normalized such that that standard deviation of the output y is 1 and the standard deviation of each input dimension is $1/\sqrt{d}$.

For Table 4, approximate homotopy and exact homotopy were used with the default parameters provided by the authors.

D.1 ADDITIONAL TARGET COVERAGES

For the sake of completeness, we reproduce the experiments of Table 4 at other target coverages for the Boston and the Riboflavin datasets. We plot the empirical coverage for GAMP and Taylor-AMP over as a function of the target coverage. The solid line and shaded area are respectively the mean and standard deviation over different train / test splits, and observe that both methods achieve the correct coverage on both datasets.

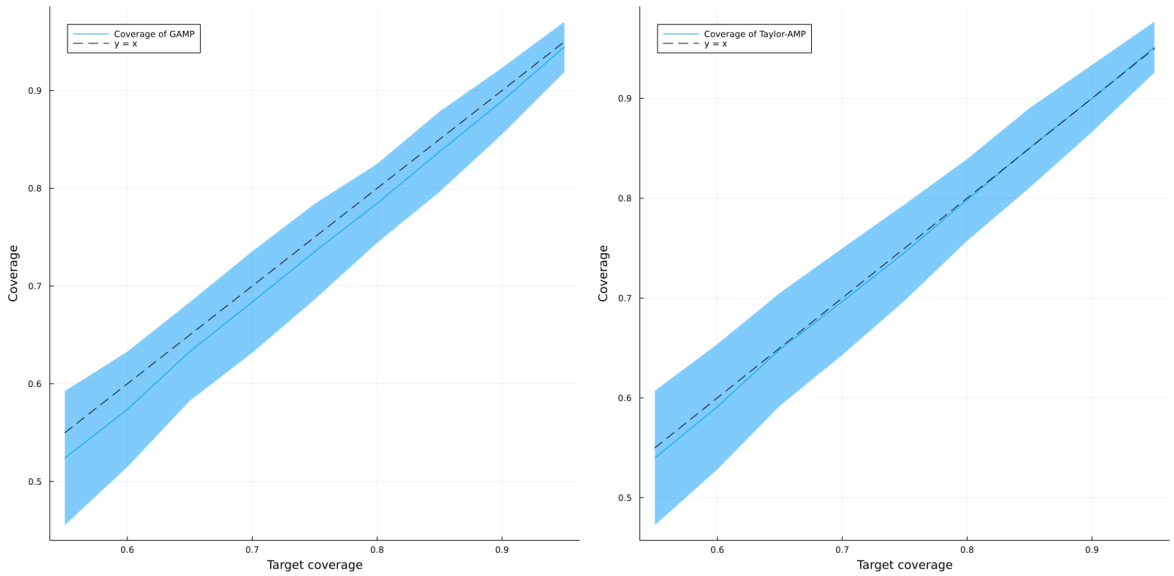

Figure 3: Coverage of AMP (Left) and Taylor-AMP (Right) on the Boston dataset as a function of the target coverage. Line and shaded area are respectively the mean and standard deviation of the coverage over 100 random training / test splits. Black dashed line corresponds to a valid coverage that matches the target.

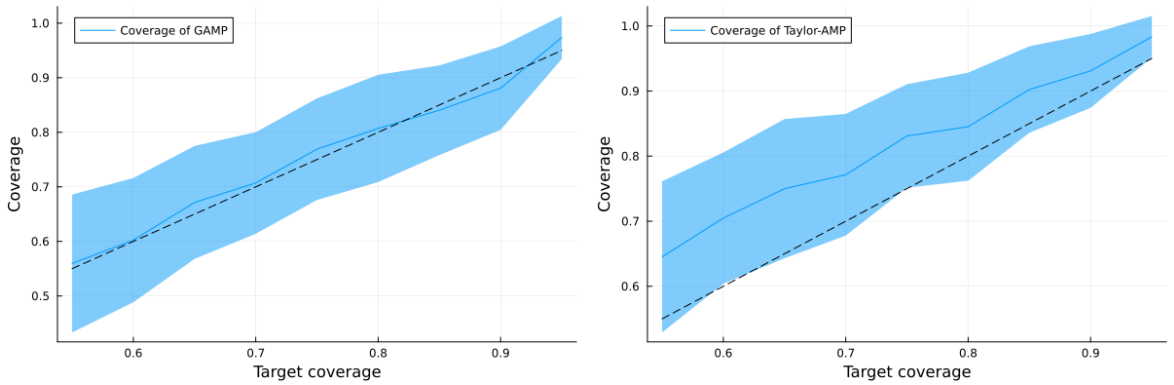

Figure 4: Coverage of AMP (Left) and Taylor-AMP (Right) on the Riboflavin dataset as a function of the target coverage. Line and shaded area are respectively the mean and standard deviation of the coverage over 20 random training / test splits. Black dashed line corresponds to a valid coverage that matches the target.

Note that all experiments in the paper were carried out on a Apple M1 Pro laptop with 16 Go of memory. The predictions intervals are obtained by selecting potential labels over a grid.

The code and data used for the experiments are available at `github.com/lclarte/ConformalAmp.jl`

E EXTENSION TO GENERALIZED LINEAR MODELS

Robust regression and quantile regression Numerical experiments in Section 4 were focused on the square loss. However, our method can be extended to other regression problems. In this section, we consider the pinball loss, also known as quantile loss, defined as

$$\ell(y, \hat{y}) = q \times \max(y - \hat{y}, 0) + (1 - q) \times \max(\hat{y} - y, 0) \tag{64}$$

and used to estimate the quantile function q of the data. The AMP can be applied to this loss with the channel

$$\text{prox}_\ell(y, \omega, V) = \arg\min_z \ell(y, z) + \frac{1}{2V}(\omega - z)^2 = \begin{cases} \omega + (q-1)V \text{ if } \omega > y - (q-1)V \\ \omega + qV \text{ if } \omega < y - qV \\ y \text{ otherwise} \end{cases} \tag{65}$$

and $g_{\text{out}}(y, \omega, V) = \frac{\text{prox}(y,\omega,V)-\omega}{V}$. For $q = 1/2$, this loss is equal (up to a factor 2 scaling) to the absolute value loss, as it equates

$$\ell(y, \hat{y}) = \frac{1}{2}|y - \hat{y}| \tag{66}$$

which is notably used for robust regression in the presence of outliers.

Binary classification Conformal prediction has been successfully applied for classification tasks Angelopoulos et al. [2021], Angelopoulos and Bates [2022]. Consider a classification task with k classes, where a predictor estimate the probabilities $p_1(\boldsymbol{x}), \cdots, p_n(\boldsymbol{x})$. Then, the conformity scores are defined as

$$\sigma_i = \sum_{k=1}^{\pi^{-1}(y)} p_{\pi(1)} \tag{67}$$

where π is a permutation that ranks the classes by decreasing order of probability, i.e $p_{\pi(1)} > \cdots > p_{\pi(K)}$. In words, the score is the sum of the probability of all the classes whose p_i is higher of equal to the true observed class.

In the context of generalized linear model, one might train an estimator using the cross entropy loss with an L_2 regularizer. For $K = 2$ classes, this is logistic regression

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} - \sum_{i=1}^{n} \log\left(1 + e^{-y_i \times \boldsymbol{x}_i^\top \boldsymbol{\theta}}\right) + \lambda/2 \|\boldsymbol{\theta}\|^2 \tag{68}$$

As for regression, one can use AMP and Taylor-AMP with the adequate channel and denoising function to estimate $\hat{\boldsymbol{\theta}}$, and compute the leave-one-out estimators using Eq. (13). For the logistic loss, the channel function is defined as

$$g_{\text{out}}(y, \omega, V) = \frac{\text{prox}\ell_{\omega,V}(y, \cdot) - \omega}{V}, \qquad \text{prox}\ell_{\omega,V}(y, \cdot) = \arg\min_z \ell(y, z) + \frac{1}{2V}(z - \omega)^2 \tag{69}$$

F ASYMPTOTIC OF THE PREDICTION INTERVAL SIZES UNDER GAUSSIAN ASSUMPTION

Our method leverages the state-evolution equations of AMP. In fact, using the state evolution equations of AMP, we can sharply compute the size of the prediction intervals in the high-dimensional limit, under the assumption of Eq. (17). First, consider the leave-one-out residualx r_i

$$r_i := \hat{\boldsymbol{\theta}}_{-i}(y)^\top \boldsymbol{x}_i - y_i \tag{70}$$

such that $\sigma_i = |r_i|$. These residuals can be computed using r-BP as explained in Appendix A. Indeed, in the high-dimensional limit

$$r_i = \hat{\boldsymbol{\theta}}_{-i,rBP}(y)^\top \boldsymbol{x}_i - y_i \tag{71}$$

where the value of the vector $\hat{\boldsymbol{\theta}}_{-i,rBP}$ at the index μ is the cavity mean $\hat{\theta}_{\mu\to i}$ defined in Eq. (28). Now, note that the distribution on the r_i can be easily computed under the Gaussian data assumption : by definition, the vector $\hat{\boldsymbol{\theta}}_{-i}$ is uncorrelated with $\boldsymbol{x}_i$. Hence, $\hat{\boldsymbol{\theta}}_{-i}^\top \boldsymbol{x}_i - y_i = \left(\hat{\boldsymbol{\theta}}_{-i} - \boldsymbol{\theta}_\star\right)^\top \boldsymbol{x}_i + \varepsilon$ follows a Gaussian distribution with mean 0 and variance $\|\frac{1}{d}\left(\boldsymbol{\theta}_\star\|^2 - 2 \times \boldsymbol{\theta}_\star{}^\top \hat{\boldsymbol{\theta}}_{-i} + \|\hat{\boldsymbol{\theta}}_{-i}\|^2 + \Delta\right)$. $\rho = \frac{1}{d}\|\boldsymbol{\theta}_\star\|^2$ is given by the prior on $\boldsymbol{\theta}_\star$, and is for instance equal to 1 when $\boldsymbol{\theta}_{\star i} \sim \mathcal{N}(0,1)$. In the high-dimensional limit, the scalar products $\frac{1}{d}\boldsymbol{\theta}_\star{}^\top \hat{\boldsymbol{\theta}}_{-i}(y)$ converge to a common value $m = \lim_{d\to\infty} \frac{1}{d}\boldsymbol{\theta}_\star{}^\top \hat{\boldsymbol{\theta}}$ for all i and all y. Similarly, the square norms of the leave-one-out estimators converge to the same value $q = \lim_{d\to\infty} \frac{1}{d}\|\hat{\boldsymbol{\theta}}\|^2$.

To summarize, as $n, d \to \infty$ the residuals r_i follow the distribution

$$r_i \sim \mathcal{N}\left(0, \rho - 2m + q + \Delta\right) \tag{72}$$

with

$$m = \lim_{d\to\infty} \frac{1}{d}\boldsymbol{\theta}_\star{}^\top \hat{\boldsymbol{\theta}}, \qquad q = \lim_{d\to\infty} \frac{1}{d}\|\hat{\boldsymbol{\theta}}\|^2 \tag{73}$$

From the distribution of the residuals to the prediction interval Since the asymptotic distribution of the $(r_i)_i$ is Gaussian, one obtains the $1-\kappa$ quantile of the scores σ_i by computing the 1 - $\kappa/2$ and the $\kappa/2$ quantiles of this Gaussian distribution. By definition of full conformal prediction, a label y will be included in the prediction set if and only if $|y - \hat{\boldsymbol{\theta}}_{-(n+1)}^\top \boldsymbol{x}| < q_{1-\kappa}((\sigma_i)_i)$, but since these scores asymptotically follow the distribution of the absolute value of a Gaussian variable, its $1-\kappa$ quantile is equal to the $1-\kappa/2$ quantile of the corresponding Gaussian distribution. In conclusion, asymptotically, the prediction interval will be

$$\mathcal{S}(\boldsymbol{x}) = [\hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \pm \sqrt{\rho - 2m + q + \Delta} \times q_{1-\kappa/2}\left(Z\right)], \qquad Z \sim \mathcal{N}(0,1) \tag{74}$$

where m, q are given by the state-evolution equations of AMP that we detail in Appendix F.1. Note that $\rho - 2m + q + \Delta$ is exactly equal the generalization error (for the mean square error) of the ERM estimator. Thus, Eq. (74) directly links the generalization error of the estimator with the size of the prediction intervals and shows that the best estimator also has the tightest intervals.

F.1 STATE-EVOLUTION EQUATIONS OF AMP

As explained in the previous section, one only needs the value of the overlaps m and q (73) to compute the size of the prediction intervals in high-dimension. To do so, it is useful to go back to relaxed-BP, which is asymptotically equivalent to AMP and thus has the same overlaps.

The rBP equations are written,

$$\begin{cases} \omega_{\mu\to i}^{(t)} &= \sum_{j\neq i} X_{\mu,j}\hat{\theta}_{j\to\mu}^{(t)} \\ V_{\mu\to i}^{(t)} &= \sum_{j\neq i} X_{\mu,j}^2 \hat{C}_{j\to\mu}^{(t)} \end{cases}, \quad \begin{cases} {g_{\text{out}}}_{\mu\to i}^{(t)} &= g_{\text{out}}(y_\mu, \omega_{\mu\to i}^{(t)}, V_{\mu\to i}^{(t)}) \\ \partial {g_{\text{out}}}_{\mu\to i}^{(t)} &= \partial_\omega g_{\text{out}}(y_\mu, \omega_{\mu\to i}^{(t)}, V_{\mu\to i}^{(t)}) \end{cases} \tag{75}$$

$$\begin{cases} b_{\mu\to i}^{(t)} &= \sum_{\nu\neq\mu} X_{\nu,i} {g_{\text{out}}}_{\nu\to i}^{(t)} \\ A_{\mu\to i}^{(t)} &= -\sum_{\nu\neq\mu} X_{\nu,i}^2 \partial {g_{\text{out}}}_{\nu\to i}^{(t)} \end{cases}, \quad \begin{cases} \hat{\theta}_{i\to\mu}^{(t)} &= f_w(b_{i\to\mu}^{(t)}, A_{i\to\mu}^{(t)}) \\ \hat{C}_{i\to\mu}^{(t)} &= \partial_b f_w(b_{i\to\mu}^{(t)}, A_{i\to\mu}^{(t)}). \end{cases} \tag{76}$$

It turns out that the average asymptotic behavior of these equations can be tracked with some overlap parameters defined as follows:

$$m^{(t)} \equiv \lim_{d\to\infty} \frac{1}{d}\sum_{i=1}^d \hat{\boldsymbol{\theta}}_i^{(t)} \boldsymbol{\theta}_\star{}^\top, \qquad Q^{(t)} \equiv \lim_{d\to\infty} \frac{1}{d}\sum_{i=1}^d \hat{\boldsymbol{\theta}}_i^{(t)} \hat{\boldsymbol{\theta}}_i^{(t)\top} \tag{77}$$

$$V^{(t)} \equiv \lim_{d\to\infty} \frac{1}{d}\sum_{i=1}^d \hat{C}_i^{(t)}, \qquad \rho = \lim_{d\to\infty} \frac{\|\boldsymbol{\theta}_\star\|^2}{d}. \tag{78}$$

To derive the asymptotic behavior of these overlap parameters, we compute the overlap distributions starting from the rBP equations above.

F.1.1 Messages Distribution

For convenience, let us define $z_\mu \equiv \boldsymbol{x}_\mu^\top \boldsymbol{\theta}_\star$ and $z_{\mu\to i} \equiv \frac{1}{d}\sum_{j\neq i} \boldsymbol{x}_{\mu,j}\theta_{\star j}$.

Distribution of $(z_\mu, \omega_{\mu\to i}^{(t)})$ By the Central Limit Theorem, since $(z_\mu, \omega_{\mu\to i}^{(t)})$ are the sum of independent variables, they follow Gaussian distributions in the $d\to\infty$ limit. Therefore, we only need to compute their means, variances, and cross-correlation. Recall that from our assumptions, the random variables $X_{\mu,j}$ are i.i.d. zero-mean Gaussian with variance $1/d$. Hence, the first and second-order statistics read

$$\mathbb{E}\left[z_\mu\right] = \boldsymbol{\theta}_\star{}^\top \mathbb{E}[\boldsymbol{X}_\mu] = 0 \tag{79}$$

$$\mathbb{E}\left[z_\mu^2\right] = \sum_{i,j=1}^{d} \mathbb{E}[X_{\mu,i}X_{\mu,j}]\theta_{\star i}\theta_{\star j} = \sum_{i,j=1}^{d} \frac{1}{d}\delta_{ij}\theta_{\star i}\theta_{\star j} = \frac{\|\boldsymbol{\theta}_\star\|^2}{d} \overset{d\to\infty}{\longrightarrow} \rho \tag{80}$$

$$\mathbb{E}\left[\omega_{\mu\to i}^{(t)}\right] = \sum_{j\neq i} \mathbb{E}[X_{\mu,j}]\hat{\boldsymbol{\theta}}_{j\to\mu}^{(t)} = 0 \tag{81}$$

$$\mathbb{E}\left[\omega_{\mu\to i}^{(t)}(\omega_{\mu\to i}^{(t)})^\top\right] = \sum_{j\neq i}^{d}\sum_{k\neq i}^{d} \mathbb{E}[X_{\mu,j}X_{\mu,k}]\hat{\boldsymbol{\theta}}_{j\to\mu}^{(t)}\hat{\boldsymbol{\theta}}_{k\to\mu}^{(t)\top} = \frac{1}{d}\sum_{j\neq i}^{d} \hat{\boldsymbol{\theta}}_{j\to\mu}^{(t)}\hat{\boldsymbol{\theta}}_{k\to\mu}^{(t)\top} \tag{82}$$

$$= \frac{1}{d}\sum_{j=1}^{d} \hat{\boldsymbol{\theta}}_{j\to\mu}^{(t)}\hat{\boldsymbol{\theta}}_{j\to\mu}^{(t)\top} - \frac{1}{d}\hat{\boldsymbol{\theta}}_{i\to\mu}^{(t)}\hat{\boldsymbol{\theta}}_{i\to\mu}^{(t)\top} \overset{d\to\infty}{\longrightarrow} q^{(t)} \tag{83}$$

$$\mathbb{E}\left[z_\mu\omega_{\mu\to i}^{(t)}\right] = \sum_{j=1}^{d}\sum_{k\neq i}^{d} \mathbb{E}[X_{\mu,j}X_{\mu,k}]\hat{\boldsymbol{\theta}}_{k\to\mu}^{(t)}\boldsymbol{\theta}_{\star j} = \frac{1}{d}\sum_{j\neq i} \hat{\boldsymbol{\theta}}_{j\to\mu}^{(t)}\boldsymbol{\theta}_\star \tag{84}$$

$$= \frac{1}{d}\sum_{j=1}^{d} \hat{\boldsymbol{\theta}}_{j\to\mu}^{(t)}\boldsymbol{\theta}_\star - \frac{1}{d}\hat{\boldsymbol{\theta}}_{i\to\mu}^{(t)}\boldsymbol{\theta}_\star \overset{d\to\infty}{\longrightarrow} m^{(t)} \tag{85}$$

In summary, in the $d\to\infty$ limit :

$$\left(z_\mu, \omega_{\mu\to i}^{(t)}\right) \sim \mathcal{N}\left(0, \begin{bmatrix} \rho & m^{(t)\top} \\ m^{(t)} & q^{(t)} \end{bmatrix}\right) \tag{86}$$

Concentration of $V_{\mu\to i}^{(t)}$ In the asymptotic limit, the variances $V_{\mu\to i}^{(t)}$ concentrate around their means, which equates

$$\mathbb{E}\left[V_{\mu\to i}^{(t)}\right] = \sum_{j\neq i}^{d} \mathbb{E}\left[X_{\mu,j}^2\right]\hat{C}^{(t)} = \frac{1}{d}\sum_{j\neq i} \hat{C}_j^{(t)} = \frac{1}{d}\sum_{j=1}^{d} \hat{C}_j^{(t)} - \frac{1}{d}\hat{C}_i^{(t)} \overset{d\to\infty}{\longrightarrow} V^{(t)} \tag{87}$$

Distribution of $b_{\mu\to i}^{(t)}$ Recall from our setting that for a given input $\boldsymbol{x}_\mu$, the corresponding label is distributed as $y_\mu \sim p(\cdot|z_\mu)$. In fact, one can equivalently write $y^\mu = \varphi_0(z_\mu)$ for some (random) function φ_0. For example, the choice $\varphi_0(x) = x + \sqrt{\Delta}\xi$ corresponds to the linear regression, where $\xi \sim \mathcal{N}(0,1)$ is Gaussian noise scaled by a variance $\Delta \geq 0$. With this representation for y_μ, we have

$$b_{\mu\to i}^{(t)} = \sum_{\nu\neq\mu} X_{\nu,i} g_{\text{out}}(\varphi_0\left(z_\nu\right), \omega_{\nu\to i}^{(t)}, V_{\nu\to i}^{(t)}) \tag{88}$$

$$= \sum_{\nu\neq\mu} X_{\nu,i} g_{\text{out}}(\varphi_0\left(z_{\nu\to i} + \theta_{\star i}X_{\nu,i}\right), \omega_{\nu\to i}^{(t)}, V_{\nu\to i}^{(t)}) \tag{89}$$

$$= \sum_{\nu\neq\mu} X_{\nu,i} g_{\text{out}}(\varphi_0\left(z_{\nu\to i}\right), \omega_{\nu\to i}^{(t)}, V_{\nu\to i}^{(t)}) + X_{\nu,i}^2\theta_{\star i}\partial_z g_{\text{out}}(\varphi_0\left(z_{\nu\to i}\right), \omega_{\nu\to i}^{(t)}, V_{\nu\to i}^{(t)}) + O(d^{-3/2}), \tag{90}$$

where in the last equality we have expanded the denoising function at leading order. Taking expectation on both sides yields

$$\mathbb{E}[b_{\mu\to i}^{(t)}] = \frac{\theta_{\star i}}{d}\sum_{\nu\neq\mu}\partial_z g_{\rm out}(\varphi_0\left(z_{\nu\to i}\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) + O(d^{-3/2}) \tag{91}$$

$$= \frac{\theta_{\star i}}{d}\sum_{\nu=1}^{n}\partial_z g_{\rm out}(\varphi_0\left(z_{\nu\to i}\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) - \frac{\theta_{\star i}}{d}\partial_z g_{\rm out}(\varphi_0\left(z_{\mu\to i}\right),\omega_{\mu\to i}^{(t)},V_{\mu\to i}^{(t)}) + O(d^{-3/2}), \tag{92}$$

Note that as $d\to\infty$, it follows from our computations above that for all ν, $(z_{\nu\to i},\omega_{\nu\to i}^{(t)})$ are identically distributed according to Eq. (86). Consequently, by the Law of Large Numbers,

$$\frac{n}{d}\cdot\frac{1}{n}\sum_{\nu=1}^{n}\partial_z g_{\rm out}(\varphi_0\left(z_{\nu\to i}\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) \overset{n,d\to\infty}{\longrightarrow} \alpha\mathbb{E}_{(z,\omega)}\left[\partial_z g_{\rm out}(\varphi_0\left(z\right),\omega,V^{(t)})\right] \equiv \hat{m}^{(t)}, \tag{93}$$

from which we find that

$$\mathbb{E}[b_{\mu\to i}^{(t)}] \overset{n,d\to\infty}{\longrightarrow} \theta_{\star i}\hat{m}^{(t)}. \tag{94}$$

The second moment can be computed in a similar fashion:

$$\mathbb{E}[b_{\mu\to i}^{(t)}b_{\mu\to i}^{(t)\top}] = \sum_{\nu\neq\mu}\sum_{\kappa\neq\mu}\mathbb{E}[X_{\nu,i}X_{\kappa,i}]g_{\rm out}(\varphi_0\left(z_\nu\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)})g_{\rm out}(\varphi_0\left(z_\kappa\right),\omega_{\kappa\to i}^{(t)},V_{\kappa\to i}^{(t)})^\top \tag{95}$$

$$= \frac{1}{d}\sum_{\nu\neq\mu}g_{\rm out}(\varphi_0\left(z_{\nu\to i}\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)})g_{\rm out}(\varphi_0\left(z_{\nu\to i}\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)})^\top + O(d^{-2}) \tag{96}$$

$$= \frac{1}{d}\sum_{\nu=1}^{n}g_{\rm out}(\varphi_0\left(z_{\nu\to i}\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)})g_{\rm out}(\varphi_0\left(z_{\nu\to i}\right),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)})^\top + O(d^{-2}) \tag{97}$$

$$\overset{n,d\to\infty}{\longrightarrow} \alpha\mathbb{E}_{(z,\omega^{(t)})}\left[g_{\rm out}(\varphi_0\left(z\right),\omega^{(t)},V^{(t)})g_{\rm out}(\varphi_0\left(z\right),\omega^{(t)},V^{(t)})^\top\right] \equiv \hat{q}^{(t)}. \tag{98}$$

Hence, $b_{\mu\to i}^{(t)} = \theta_{\star i}\hat{m}^{(t)} + \left(\hat{q}^{(t)}\right)^{1/2}\xi$ with $\xi\sim\mathcal{N}(0,1)$.

Concentration of $A_{\mu\to i}^{(t)}$ It remains to show that the covariances $A_{\mu\to i}^{(t)}$ concentrate. We have

$$A_{\mu\to i}^{(t)} = -\sum_{\nu\neq\mu}X_{\nu,i}^2\partial_\omega g_{\rm out}(y_\nu,\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) \tag{99}$$

$$= -\sum_{\nu\neq\mu}X_{\nu,i}^2\partial_\omega g_{\rm out}(\varphi_0(z_\nu),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) \tag{100}$$

$$= -\sum_{\nu\neq\mu}X_{\nu,i}^2\partial_\omega g_{\rm out}(\varphi_0(z_{\nu\to i}),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) + O(d^{-3/2}). \tag{101}$$

Taking the expectation gives

$$\mathbb{E}[A_{\mu\to i}^{(t)}] = -\frac{1}{d}\sum_{\nu\neq\mu}\partial_\omega g_{\rm out}(\varphi_0(z_{\nu\to i}),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) + O(d^{-3/2}) \tag{102}$$

$$= -\frac{1}{d}\sum_{\nu=1}^{n}\partial_\omega g_{\rm out}(\varphi_0(z_{\nu\to i}),\omega_{\nu\to i}^{(t)},V_{\nu\to i}^{(t)}) - \frac{1}{d}\partial_\omega g_{\rm out}(\varphi_0(z_{\mu\to i}),\omega_{\mu\to i}^{(t)},V_{\mu\to i}^{(t)}) + O(d^{-3/2}) \tag{103}$$

$$\overset{n,d\to\infty}{\longrightarrow} -\alpha\mathbb{E}_{(z,\omega^{(t)})}\left[\partial_\omega g_{\rm out}(\varphi_0\left(z\right),\omega^{(t)},V^{(t)})\right] \equiv \hat{V}^{(t)} \tag{104}$$

State-evolution equations From the previous computations, we deduce that asymptotically the coordinates of the estimator are distributed as

$$\hat{\theta}_i^t \sim f_w\left(\theta_{\star i}\hat{m}^t + \sqrt{\hat{q}}\varepsilon,\hat{V}\right), \quad \varepsilon\sim\mathcal{N}(0,1) \tag{105}$$

And finally, we get that the overlaps m, q are the solutions of the following state-evolution equations

$$\begin{cases} m &= \mathbb{E}_{\theta_\star,\varepsilon}\left[f_w(\hat{m}\theta_\star + \sqrt{\hat{q}}\varepsilon, \hat{v})\theta_\star\right] \\ q &= \mathbb{E}_{\theta_\star,\varepsilon}\left[f_w(\hat{m}\theta_\star + \sqrt{\hat{q}}\varepsilon, \hat{v})^2\right] \\ V &= \mathbb{E}_{\theta_\star,\varepsilon}\left[\partial_b f_w(\hat{m}\theta_\star + \sqrt{\hat{q}}\varepsilon, \hat{v})\right] \end{cases} \tag{106}$$

for $\varepsilon \sim \mathcal{N}(0,1)$ and

$$\begin{cases} \hat{m} &= \alpha\mathbb{E}_{z,\omega}\left[\partial_z g_{\text{out}}(\varphi_0(z), \omega, V)\right] \\ \hat{q} &= \alpha\mathbb{E}_{z,\omega}\left[g_{\text{out}}(\varphi_0(z), \omega, V)^2\right] \\ \hat{V} &= -\alpha\mathbb{E}_{z,\omega}\left[\partial_\omega g_{\text{out}}(\varphi_0(z), \omega, V)\right] \end{cases} \tag{107}$$

Solving these equations, we deduce the value of m, q that we can plug in Eq. (74) to compute the size of the prediction intervals in the high-dimensional limit.