Uncertainty quantification with deep ensembles : a toy model

clarte.lucas

July 2024

1 Introduction

1.1 Related work

Bootstrap resampling is a cornerstone of classical statistics [citation needed], and its application to modern machine learning techniques has been more recently explored [**?**].

2 Setting

Consider training data $\mathcal{D} = (\boldsymbol{x}_i, y_i)_{i=1}^n$ where the input follows the distribution $\boldsymbol{x} \sim \mathcal{N}(0, I_d)$ and the labels are generated by a teacher vector $\boldsymbol{w}_\star$, such that $y = f\left(\boldsymbol{w}_\star{}^\top \boldsymbol{x}_i\right)$ where f is a function that can be random. For instance, in the case of linear regression, one can use $f(z) = z + \sqrt{\Delta}\varepsilon, \varepsilon \sim \mathcal{N}(0,1)$. For binary classification, one can use $p\left(y = 1|\boldsymbol{w}_\star{}^\top \boldsymbol{x}\right)$. The function f and the teacher $\boldsymbol{w}_\star$ will typically be unknown to the statistician, and the goal is to recover $\boldsymbol{w}_\star$ using limited data. Putting the context of overparametrized models, we consider now an ensemble of random features $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_K$ generated by

$$\boldsymbol{v}_i^j = \sigma_i\left(F^j \boldsymbol{x}_i\right), \quad F^j \in \mathbb{R}^{p\times d} \tag{1}$$

Where the components of F_i are i.i.d $\mathcal{N}(0, 1/d)$. We will define $\mathcal{D}^j = \left(\boldsymbol{v}_i^j, y_i\right)_{i=1}^n$.

Empirical risk minimizer We will first consider frequentist estimators that minimize the Ridge loss on the random features $\boldsymbol{v}$:

$$\hat{\boldsymbol{w}}^j = \arg\min_{\boldsymbol{w}} \mathcal{L}\left(\boldsymbol{w}|\mathcal{D}^j\right) = \sum_i \frac{|y_i - \boldsymbol{w}^\top \boldsymbol{v}_i^j|^2}{2} + \frac{\lambda}{2}\|\boldsymbol{w}\|^2 \tag{2}$$

Bayes-optimal estimator Given data $\mathcal{D} = (\boldsymbol{v}_i, y_i)_{i=1}^n$, the Bayes-optimal estimator computes the posterior distribution

$$p\left(y|\boldsymbol{v}, \mathcal{D}\right) = \int \mathrm{d}\boldsymbol{v} p(\boldsymbol{w}|\mathcal{D}) p(y|\boldsymbol{w}) \tag{3}$$

By definition, the Bayes-optimal estimator is the one with the best accuracy, given $\mathcal{D}$.

Pseudo-Bayes Computing the Bayes-optimal estimator eq. (3) is practically impossible as it requires the knowledge of the prior and likelihood function of the data-generating process, which is inaccessible in practice. Then, an alternative is the *pseudo-Bayes estimator* that samples the following distribution associated to the loss.

$$p\left(\boldsymbol{w}|\mathcal{D}^j\right) \propto e^{-\beta \mathcal{L}(\boldsymbol{w})} \tag{4}$$

where β is the inverse temperature that controls the variance of the distribution. In the context of Bayesian neural network, $\beta > 1$ (respectivelly $\beta < 1$) correspond to *cold posteriors* (respectivelly *hot posteriors*).

3 Uncertainty quantification with ensembles

Resampling overparametrized models Resampling techniques such as bootstrap or subsampling are a classical tools in classical statistics to estimate the variance of an estimator and quantify uncertainty. However, in the high-dimensional regime, applying these techniques is not straightforward [citation needed]. One way to model resampling is to reframe it as a weighted empirical risk minimization :

$$\hat{\boldsymbol{w}}^j = \arg\min_{\boldsymbol{w}} \mathcal{L}\left(\boldsymbol{w}|\mathcal{D}^j, \boldsymbol{p}^j\right) = \sum_i p_i^j \frac{|y_i - \boldsymbol{w}^\top \boldsymbol{v}_i^j|^2}{2} + \frac{\lambda}{2}\|\boldsymbol{w}\|^2 \tag{5}$$

where the weights $p_i^j \in \mathbb{N}$ follow a multinomial distribution for the Bootstrap, or a Bernoulli distribution for subsampling. Previous work [citation needed] has analyzed the performance of bootstrap and resampling for linear regression, but not in the context of overparametrized models.

3.1 Regression

In the context of regression, when working with an ensemble of estimators (2) $\hat{\boldsymbol{w}}^1, \cdots, \hat{\boldsymbol{w}}^K$ obtained either by resampling, using different random features or sampling a Bayesian estimator, the standard way to quantify uncertainty is through the prediction variance across the ensemble. For a single sample $\boldsymbol{x}$ with associated random features $\boldsymbol{v}^1, \cdots, \boldsymbol{v}^K$ (potentially equal), the prediction variance for $\boldsymbol{x}$ is

$$\hat{\sigma}^2(\boldsymbol{x}) := \mathrm{Var}_{\hat{\boldsymbol{w}}^j}\left[\hat{\boldsymbol{w}}^{j\top} \boldsymbol{v}^j\right] \tag{6}$$

Rescaling in classifcation For classification tasks, a popular way to improve uncertainty quantification is to rescale the confidence function without changing the predicted class. A popular way is *temperature scaling* that uses the following confidence :

$$f\left(\boldsymbol{v}\right) = \sigma\left({\boldsymbol{w}_\star}^\top \boldsymbol{v}/T\right) \tag{7}$$

where T is a scalar factor learned by minimizing the loss on some validation data. Plenty of alternatives to temperature scaling have been developed [citation needed], but all rely on rescaling the confidence function by a few factors.

Rescaling the variances in regression Similar to temperature scaling for classification tasks [citation needed], one can rescale the ensemble variance of the prediction. If one uses a likelihood $p(y|\boldsymbol{x})$ as a Gaussian with mean $\hat{\mu}(\boldsymbol{x}) = \cdots$ and variance $\hat{\sigma}(\boldsymbol{x}) = Var(\boldsymbol{x})$ as defined in (6), we can rescale the prediction variance by a factor T

$$\hat{\sigma}^2(\boldsymbol{x}) \to \hat{\sigma}^2(\boldsymbol{x})/T \tag{8}$$

A natural way to compute the factor T is such that the average prediction variance is equal to the mean square error on the validation set.

3.2 Classification

In the context of classification,

4 Technical results

Our main technical result is the computation of the correlation $\hat{\boldsymbol{w}}^{1\top}\hat{\boldsymbol{w}}^2$ for $\hat{\boldsymbol{w}}^i$ either computed by empirical risk minimization of defined as the average of a Bayesian posterior, trained on the bootstrap resample or subsampled version of $\mathcal{D}$. This result is an extension of [citation needed].

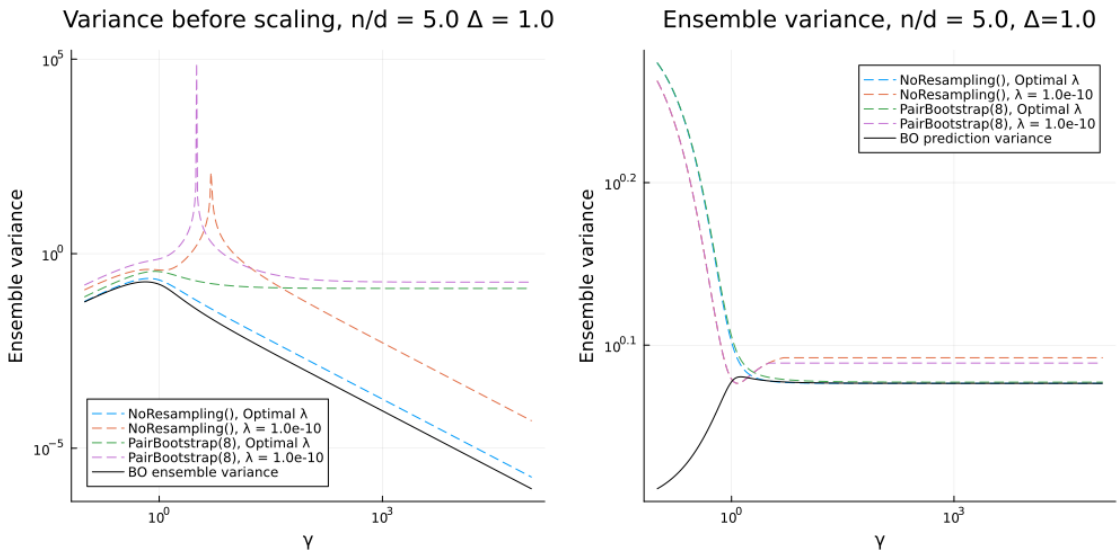

Figure 1: Ensemble variance with and without resampling for random features, at finite overparametrization. Left : before rescaling the variance. Right : after rescaling the variance.

Theorem 4.1. *Consider* $\hat{\boldsymbol{w}}^1, \hat{\boldsymbol{w}}^2$ *that minimize the loss defined in eq.* (5)*. Then, the variance is given by ?? where the overlaps* m, Q *that are solutions of the following equations*

$$\begin{cases} m &= \cdots \\ Q &= \cdots \\ V &= \cdots \end{cases}, \begin{cases} \hat{m} &= \cdots \\ \hat{Q} &= \cdots \\ \hat{V} &= \cdots \end{cases}$$

theorem 4.1 allows us to compute efficiently the correlation between two estimators in high-dimensions benchmark them in the random features model. For the sake of concreteness, this theorem is formulated for Ridge regression and bootstrap resamples, but we refer to ?? for a more general statement.

Theorem 4.2.

4.1 Numerics

Ensemble variance In Figure 1, we plot the variance defined in eq. (6). We compare there the ensemble variance of pair bootstrap and ERM without resampling, at optimal regularization and at almost zero regularization ($\lambda = 1e{-}10$), before (Left) and after rescaling the variance (Right) to match the generalisation error. We see that with rescaling, it makes almost no difference to use bootstrap or not : the determining factor is λ : at optimal λ, the ensemble variance matches the predictive variance of the Bayes-optimal, for the sole reason that at optimal λ, Ridge has almost optimal test error. In other words, good generalisation error induces accurate uncertainty estimation.

Which Bayesian variance are we comparing to ? In the Left panel of fig. 1, we plot the ensemble variance for Bayes posteriors trained on different resamples of F. This ensemble variance is equal to $q_0 - q_1$ using the overlaps from state-evolution. In the Right panel, we plot the variance of the Bayes posterior (which corresponds to the overlap v in the state-evolution equation) for one single random-features model.

Summary It looks like doing resampling does not add much to UQ in this model : using optimal regularization and using post-hoc rescaling of the variance is enough to have good UQ, in the sense that you approximate correctly some notion of variance of the BO estimator. In one case (before the rescaling), the ensemble variance of the ERM estimator is approximately the one of the BO. After rescaling, you approximate the posterior variance.

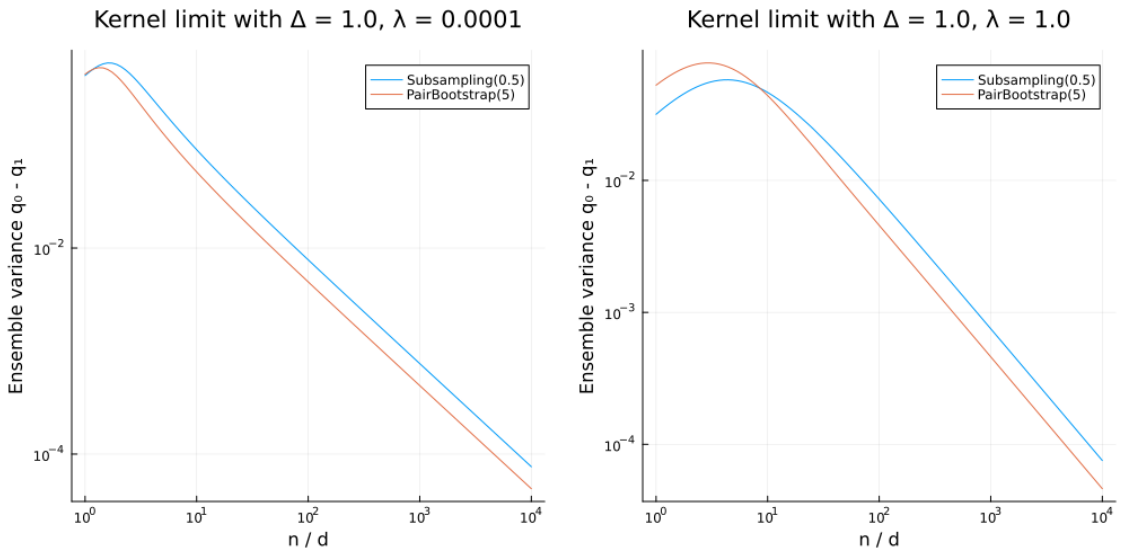

Figure 2: Ensemble variance for subsampling and bootstrap in the kernel limit

Kernel limit At infinite overparametrization, the random features all converge to the same kernel limit independently of the actual realization of the matrix F. For this reason, with resampling the ensemble method converges to 0. On the other hand, in fig. 2, we plot the ensemble variance using bootstrap and subsampling, as a function of the sampling ratio n/d. We observe that in both cases the variance scales as d/n[1]

[1]This is coherent with what we observed in the bootstrap paper where the variance scales as d/n)

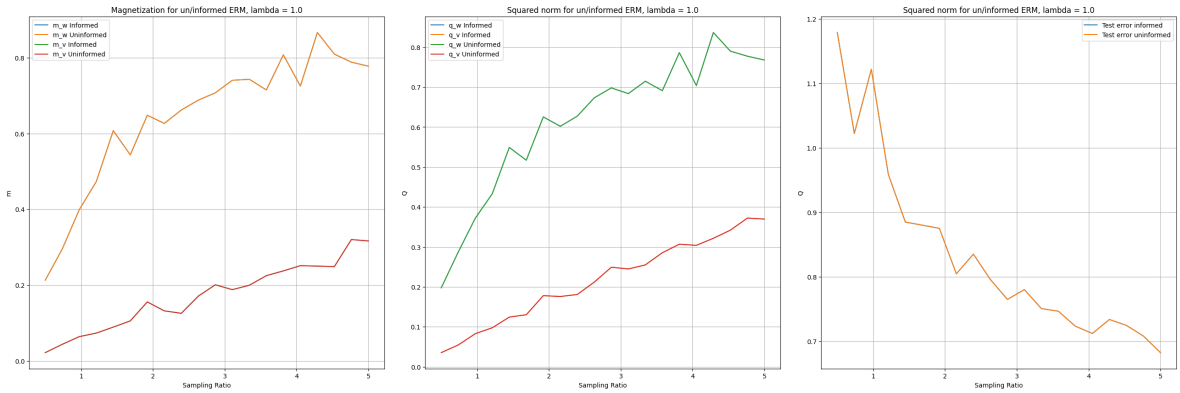

Figure 3: From left to right : plot of m, q and the test error of estimator $\hat{\boldsymbol{w}}, \hat{\boldsymbol{v}}$ obtained by minizing (10) for $\lambda = 1$, which corresponds to the MAP estimator.

5 Model for mean-variance estimation

5.1 Setting

The data will be generated by the model

$$y = \boldsymbol{w}_\star^\top \boldsymbol{x} + \varepsilon, \varepsilon \sim \mathcal{N}(0, \sigma\left(\boldsymbol{v}_\star^\top \boldsymbol{x}\right)) \tag{9}$$

Where once again the input $\boldsymbol{x}$ comes from the standard normal distribution. The teacher vectors $\boldsymbol{w}_\star, \boldsymbol{v}_\star$ are also sampled from the distribution $\mathcal{N}(\boldsymbol{0}, I_d)$. We will use the same non-linearity σ to parametrize the noise variance as the original deep ensemble paper, namely the softplus function[2]

$$\sigma\left(z\right) = \log\left(1 + e^z\right)$$

We will infer the vectors $\boldsymbol{w}, \boldsymbol{v}$ with the empirical risk

$$\mathcal{L}(\boldsymbol{w}, \boldsymbol{v}) = \sum_{i=1}^{n} \frac{\left(\boldsymbol{w}^\top \boldsymbol{x}_i - y_i\right)^2}{2\sigma\left(\boldsymbol{v}^\top \boldsymbol{x}_i\right)} + \frac{\lambda}{2}\left(\|\boldsymbol{w}\|^2 + \|\boldsymbol{v}\|^2\right) \tag{10}$$

The loss $\mathcal{L}$ corresponds to the log-posterior when $\lambda = 1$. The goal is to characterize the performance of the empirical risk minimizer of $\mathcal{L}$ in high-dimensions. First thing to note is that the loss (10) is non-convex, meaning that gradient descent may be stuck in local minima.

5.2 Numerics

Uninformed and informed initialization To test whether the non-convexity of the training loss has an impact on the estimation, we run gradient descent on $\mathcal{L}$ with an *informed* initialization, starting from the teacher $\boldsymbol{w}_\star, \boldsymbol{v}_\star$, and an *uninformed* initialization starting from random vectors. In Figure 3, we plot the overlaps and the test error for $\lambda = 1$ for both initializations. For this choice of λ, both estimators coincide. Same for Figure 4 where we take $\lambda = 0.1$.

5.3 State-evolution

We have the loss

$$\ell(y, \boldsymbol{z}) = \frac{\left(y - z_1\right)^2}{2\sigma\left(z_2\right)} \tag{11}$$

[2]Note : one could use other non-linearities, such as the square function.

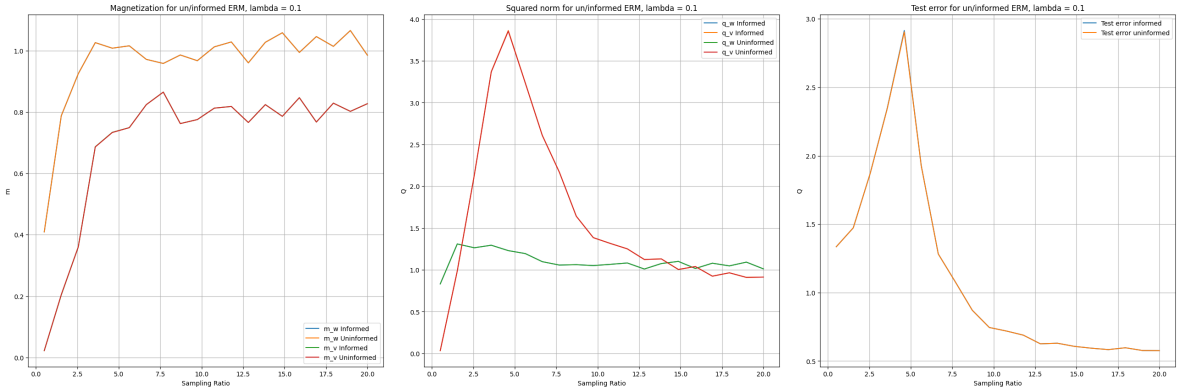

Figure 4: From left to right : plot of m, q and the test error of estimator $\hat{\boldsymbol{w}}, \hat{\boldsymbol{v}}$ obtained by minizing (10) for $\lambda = 0.1$.

for which we can write the proximal operator (and the corresponding channel function) :

$$prox(y, \boldsymbol{\omega}, \boldsymbol{V}) = \arg\min_{\boldsymbol{z}} \ell(y, \boldsymbol{z}) + V^{-1}(\boldsymbol{z} - \boldsymbol{\omega})/2 \tag{12}$$

$$g(y, \boldsymbol{\omega}, \boldsymbol{V}) = V^{-1}\left(prox(y, \boldsymbol{\omega}, \boldsymbol{V}) - \boldsymbol{\omega}\right) \tag{13}$$

Note that because the contributions of z_1 and z_2 in the loss are not separable, the covariance matrix V is *a priori* not diagonal.

The denoising step is similar as for Bootstrap [reference needed], since the prior is factorized.

6 To put in the appendix : Gaussian Covariate Model

Recall that by Gaussian universality, this model is equivalent in the high-dimensional limit to

$$\boldsymbol{v}_i = \kappa_i^1 F_i \boldsymbol{x} + \kappa_i^\star z_i \tag{14}$$

where z_i follows the normal distribution. In what follows, we consider the case $K = 2$. Then, the vector $(\boldsymbol{x}, \boldsymbol{v}_1, \boldsymbol{v}_2)$ follows a zero mean Gaussian with covariance :

$$\begin{pmatrix} I_d & \Phi_1 & \Phi_2 \\ \Phi_1 & \Omega_1 & \Phi_{1,2} \\ \Phi_2 & \Phi_{1,2} & \Omega_2 \end{pmatrix}, \quad \begin{cases} \Phi_i & = \kappa_i^1 F_i \\ \Omega_i & = {\kappa^1}_i^2 F_i F_i^T + {\kappa^\star}_i^2 \\ \Phi_{1,2} & = \kappa_1^1 \times \kappa_2^1 F_1 F_2^T \end{cases} \tag{15}$$

The random features define a *projected teacher* for each random feature :

$$\boldsymbol{w}_{\star i} = \Omega_i^{-1} \Phi_i^T \boldsymbol{\theta}_\star \sim \mathcal{N}\left(0, \Sigma_i^\star\right), \quad \Sigma_i^\star = \Omega_i^{-1} \Phi_i^T \Phi \Omega_i^{-1} \tag{16}$$

The label as a function of ${\boldsymbol{w}_{\star i}}^\top \boldsymbol{v}_i$ is

$$y = {\boldsymbol{w}_{\star i}}^\top \boldsymbol{v}_i / \sqrt{d} + \varepsilon, \quad \varepsilon \sim \mathcal{N}\left(0, \Delta + \tau_{add}\right) \tag{17}$$

with the additional the noise variance $\tau_{add} = 1 - \mathbb{E}_{x \sim MP}\left[\frac{(\kappa^1)^2 x}{(\kappa^1)^2 x + (\kappa^\star)^2}\right]$. Here, the MP distribution is the limiting distribution of the spectrum of FF^T because the entries are i.i.d Gaussian.

Bayes-optimal estimator in the GCM To compute the Bayes-optimal estimator, we sample from the distribution

$$p(\boldsymbol{w}_i | \mathcal{D}) \propto \prod_{\mu=1}^{n} p\left(y_\mu | \boldsymbol{w}^\top \boldsymbol{v}_{i\mu}\right) \prod \mathcal{N}\left(\boldsymbol{w} | \boldsymbol{0}, \Sigma_\star\right) \tag{18}$$

In the case of a Gaussian likelihood and Gaussian prior, the Bayes-optimal estimator $\boldsymbol{w}_{BO,i}$ is just the mean of this distribution. The performance of a single Bayes-optimal estimator has been characterized in our AISTATS paper. The goal here is to compute the correlation between two estimators trained on different random features, coming from two matrices F_1 and F_2. We thus define the overlap

$$q_1 := \frac{1}{d} \mathbb{E}\left[\boldsymbol{w}_1^\top \boldsymbol{v}_1 \times \boldsymbol{w}_2^\top \boldsymbol{v}_2\right] = \frac{1}{d} \boldsymbol{w}_1^\top Cov(\boldsymbol{v}_1, \boldsymbol{v}_2) \boldsymbol{w}_2 = \frac{1}{d} \boldsymbol{w}_1^\top \Phi_{1,2} \boldsymbol{w}_2 \tag{19}$$

The computation of this correlation has been done in Loureiro et al. for Ridge estimators. They derive the expression (for Ridge estimator $l = {(y-z)^2}/{2\hat{\Delta}}$) :

$$\begin{cases} \hat{q}_1 & = \frac{\rho + q_1 - 2m + \Delta + \tau_{add}}{\hat{\Delta} + v} \\ q_1 & = \left(1 + \frac{\hat{q}_1}{\hat{m}^2}\right) \times m^2 \end{cases} \tag{20}$$

where the overlaps $m, q, v, \hat{m}, \hat{q}, \hat{v}$ are the usual solutions of the fixed point equations. For the Bayes-optimal estimators, q_1 also solves (20). In 7, we also derive the expression of q_1 from the state-evolution equations of AMP.

7 State evolution for q_1 from AMP

An option is to write an AMP algorithm that estimates the two estimators simultaneously, with the same channel function

$$\dots \tag{21}$$

and the denoisers related to their own prior $\mathcal{N}(0, \Sigma_{\star i})$ where (cf. the AISTATS paper)

$$\Sigma_{\star,i} = \Omega_i^{-1/2} \Phi_i^T \Phi_i \Omega_i^{1/2} \tag{22}$$

which is equivalent of defining a single prior on $(\boldsymbol{w}_1, \boldsymbol{w}_2) \in \mathbb{R}^{2\times p}$ with a factorized form

$$\mathcal{N}\left(\mathbf{0}, \begin{pmatrix} \Sigma_{\star,1} & 0 \\ 0 & \Sigma_{\star,2} \end{pmatrix}\right)$$

the denoiser for a prior $\mathcal{N}(0, \Sigma_\star)$ is

$$f_a(\boldsymbol{b}, \mathrm{A}) = \left(\mathrm{A} + \Sigma_\star^{-1}\right)^{-1} \boldsymbol{b} \in \mathbb{R}^p, \quad f_v(\boldsymbol{b}, \mathrm{A}) = \left(\mathrm{A} + \Sigma_\star^{-1}\right)^{-1} \in \mathbb{R}^p \tag{23}$$

with a diagonal matrix $\mathrm{A} = Diag(\boldsymbol{A})$.

Algorithm 1 GAMP for Bayes-optimal on random features $\boldsymbol{v}_\mu^j$ for $1 \leqslant j \leqslant B$ different random feature matrices F_j.

Input: $(\boldsymbol{v}_\mu^j)_{mu=1}^n \in \mathbb{R}^{n\times p}$ $(1 \leqslant j \leqslant B)$, $\boldsymbol{y} \in \mathbb{R}^n$

Initialize: ${\boldsymbol{g}_{\mathrm{out}}}_\mu^{(0)} = \mathbf{0}$ for $1 \leq \mu \leq n$, $\quad \mathrm{A}_i^{(0)} = \mathrm{I}_B$ for $1 \leq i \leq d$

Initialize: $\hat{\boldsymbol{\theta}}_i^{(1)} \in \mathbb{R}^B$ and $\hat{\mathrm{C}}_i^{(1)} \in \mathbb{R}^{B\times B}$ for $1 \leq i \leq d$

Repeat for $t = 1, 2, \dots$**:**

// Update of the means $\omega_\mu^j \in \mathbb{R}$ and covariances $\mathrm{V}_\mu \in \mathbb{R}^{B\times B}$ for $1 \leqslant \mu \leqslant n$:

${\omega_\mu^j}^{(t)} = \sum_{i=1}^d v_{\mu,i}^j \boldsymbol{w}_i^{j,(t)} - \boldsymbol{v}_{\mu,i}^{j2} \hat{\boldsymbol{c}}_i^{j(t)} {\boldsymbol{g}_{\mathrm{out}}}_\mu^{j(t-1)} \mid {V_\mu^{j,k}}^{(t)} = \sum_{i=1}^d \boldsymbol{v}_{\mu,i}^j \boldsymbol{v}_{\mu,i}^k \hat{\boldsymbol{c}}_i^{j,k(t)}$ for $1 \leqslant j, k \leqslant B$

// Update of $\boldsymbol{g}_{\mathrm{out}\mu} \in \mathbb{R}^B$ and $\partial_\omega \boldsymbol{g}_{\mathrm{out}\mu} \in \mathbb{R}^{B\times B}$ for $1 \leq \mu \leq n$:

${\boldsymbol{g}_{\mathrm{out}}}_\mu^{(t)} = \boldsymbol{g}_{\mathrm{out}}\left(\boldsymbol{\omega}_\mu^{(t)}, y_\mu, \mathrm{V}_\mu^{(t)}\right) \mid \partial_{\boldsymbol{\omega}} {\boldsymbol{g}_{\mathrm{out}}}_\mu^{(t)} = \partial_{\boldsymbol{\omega}} \boldsymbol{g}_{\mathrm{out}}\left(\boldsymbol{\omega}_\mu^{(t)}, y_\mu, \mathrm{V}_\mu^{(t)}\right)$

// Update of means $\boldsymbol{b}_i \in \mathbb{R}^B$ and covariances $\mathrm{A}_i \in \mathbb{R}^{B\times B}$ for $1 \leq i \leq d$:

$\mathrm{A}_i^{jk(t)} = -\sum_{\mu=1}^n \boldsymbol{v}^j{}_{\mu,i} \boldsymbol{v}^k{}_{\mu,i} \partial_\omega {\boldsymbol{g}_{\mathrm{out}}}_\mu^{jk(t)} \mid \boldsymbol{b}_i^{j(t)} = \mathrm{A}_i^{jj(t)} \hat{\boldsymbol{\theta}}_i^{j(t)} + \sum_{\mu=1}^n \boldsymbol{v}_{\mu,i}^j {\boldsymbol{g}_{\mathrm{out}}}_\mu^{jj(t)}$

// Update of the estimated marginals $\hat{\boldsymbol{\theta}}_i \in \mathbb{R}^B$ and $\hat{\mathrm{C}}_i \in \mathbb{R}^{B\times B}$ for $1 \leq i \leq d$:

$\hat{\boldsymbol{\theta}}^{j(t+1)} = \boldsymbol{f}_a(\boldsymbol{b}^{j(t)}, \mathrm{A}^{jj(t)}) \mid \hat{\mathrm{C}}^{j(t+1)} = \partial_{\boldsymbol{b}} \boldsymbol{f}_a(\boldsymbol{b}^{j(t)}, \mathrm{A}^{jj(t)})$

Until convergence

Output: $\hat{\boldsymbol{\theta}}_1, \dots, \hat{\boldsymbol{\theta}}_d$ and $\hat{\mathrm{C}}_1, \dots, \hat{\mathrm{C}}_d$

Note on 1 : we know that the matrix $\mathrm{V} \in \mathbb{R}^{2\times 2}$ is diagonal because the distribution that we are sampling from is factorized between $\boldsymbol{w}^1$ and $\boldsymbol{w}^2$ (so conditionned on the data, the two estimators are independent).

7.1 State-evolution equations

To compute the state-evolution, we consider the relaxed-BP which is equivalent to AMP in high-dimensions. We define :

$$\begin{cases} \boldsymbol{b}_{\mu\to i}^{j(t)} & = \sum_{\nu\neq\mu} \boldsymbol{v}_{\nu,i}^j \boldsymbol{g}_{\mathrm{out}}(\varphi_0(z_\nu), \boldsymbol{\omega}_{\nu\to i}^{j(t)}, \boldsymbol{V}_{\nu\to i}^{j(t)}) \\ \boldsymbol{A}_{\mu\to i}^{j(t)} & = -\sum_{\nu\neq\mu} \boldsymbol{v}_{\nu,i}^{j2} \partial \boldsymbol{g}_{\mathrm{out}}(\varphi_0(z_\nu), \boldsymbol{\omega}_{\nu\to i}^{j(t)}, \boldsymbol{V}_{\nu\to i}^{j(t)}) \end{cases}$$

Concentration of $\boldsymbol{V}_\mu^j$ $\quad \mathbb{E}\left[\boldsymbol{V}_\mu^j\right] = \sum_{k\neq i} \mathbb{E}\left[\boldsymbol{v}_k^{j2}\right] \hat{\boldsymbol{c}}_k^j \to \frac{1}{p} \sum_{p=1} \hat{\boldsymbol{c}}^j$ as the expectation of $\boldsymbol{v}_k^{j2}$ is simply $\frac{1}{d}$.

Limitation of distribution of $\boldsymbol{b}$ For $j = 1, 2$, we have asymptotically

$$\mathbb{E}\left[b_{\mu\to i}^{j}\right] \to \boldsymbol{w}_{\star i}^{\ j}\hat{m}_j \tag{24}$$

Where

$$\hat{m}_j = {}^{n}\!/_{p}\mathbb{E}_{z,\omega}\left[\partial_z g_{out}^{j}\right] \tag{25}$$

Note that if the random matrices are sampled from the same distribution, then the channel functions are the same i.e $\hat{m}^1 = \hat{m}^2$.

The covariance is

$$\mathbb{E}\left[b_{\mu\to i}^{j} b_{\mu\to i}^{k}\right] \to \Phi_{j,k_{i,i}} \times {}^{n}\!/_{p}\mathbb{E}\left[g_{out}^{j} \times g_{out}^{k}\right] \tag{26}$$

Note that the correlation is indexed by i because of the non-i.i.d covariance matrix between the two student samples, but the matrix $\hat{\mathrm{Q}} = \frac{n}{p}\mathbb{E}\left[g_{out}^{1} \times g_{out}^{2}\right]$ is independent from the index.

In the end, for $1 \leqslant i \leqslant d$, the vector $\boldsymbol{b}_i \in \mathbb{R}^2$ has the distribution

$$\boldsymbol{b}_i = \boldsymbol{w}_{\star i} \otimes \hat{\boldsymbol{m}} + \boldsymbol{\xi}_i, \boldsymbol{\xi}_i \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} \hat{Q}_{1,1} & \Phi_{i,i}^{1,2} \times \hat{Q}_{1,2} \\ \Phi_{i,i}^{1,2} \times \hat{Q}_{1,2} & \hat{Q}_{2,2} \end{pmatrix}\right) \tag{27}$$

Computation of the overlaps Consider the magnetization $\boldsymbol{m} \in \mathbb{R}^2$. Asymptotically, we have that[3]

$$\boldsymbol{m} \in \mathbb{R}^2 = \frac{1}{p}\mathbb{E}_{\boldsymbol{w}_\star, \boldsymbol{\xi}}\left[\begin{pmatrix} \boldsymbol{f}_a\left(\hat{m}_1 \boldsymbol{w}_\star^{\ 1} + (\boldsymbol{\xi})_1, \boldsymbol{A}^1\right)^\top \boldsymbol{w}_\star^{\ 1} \\ \boldsymbol{f}_a\left(\hat{m}_2 \boldsymbol{w}_\star^{\ 2} + (\boldsymbol{\xi})_2, \boldsymbol{A}^2\right)^\top \boldsymbol{w}_\star^{\ 2} \end{pmatrix}\right] \tag{28}$$

The expression of e.g. the first coordinate of the expectation is

$$\mathbb{E}\left[\boldsymbol{f}_a\left(\hat{m}_1 \boldsymbol{w}_\star^{\ 1} + (\boldsymbol{\xi})_1, \hat{V}_1\right)^\top \boldsymbol{w}_\star^{\ 1}\right] = \mathbb{E}\left[\boldsymbol{w}_\star^{\ 1\top}(\boldsymbol{A} + \Sigma_{\star,j}^{-1})^{-1}\left(\boldsymbol{w}_{\star i}^{\ 1}\hat{m}_1 + \boldsymbol{\xi}\right)\right] \tag{29}$$

which equates

$$m^1 = \frac{1}{p}\hat{m}^1 \times \mathbb{E}\left[Tr(\boldsymbol{w}_\star^{\ 1}\boldsymbol{w}_\star^{\ 1\top}(\boldsymbol{A} + \Sigma_{\star,1}^{-1})^{-1}\right] = \hat{m}^1 \times Tr\left(\Sigma_{\star,1}(\boldsymbol{A} + \Sigma_{\star,1}^{-1})^{-1}\right) \tag{30}$$

and we recover the result of previous work.

Consider now the overlap $\mathrm{Q} \in \mathbb{R}^{2\times 2}$. We have for $i \neq j$ (the case $i = j$ has been covered before in our AISTATS paper)

$$Q_{1,2} = \mathbb{E}_{\boldsymbol{w}_\star, \xi}\left[\boldsymbol{f}_a\left(\boldsymbol{w}_\star^{\ 1}\hat{m}^1 + \boldsymbol{\xi}^1, \hat{V}^1\right)^\top \boldsymbol{f}_a\left(\boldsymbol{w}_\star^{\ 2}\hat{m}^2 + \boldsymbol{\xi}^2, \hat{V}^2\right)\right] \tag{31}$$

$$= \hat{m}^1\hat{m}^2 \times \mathbb{E}\left[\boldsymbol{w}_\star^{\ 1\top}\left(\hat{\boldsymbol{V}}^1 + \Sigma_{\star,1}^{-1}\right)^{-1}\left(\hat{\boldsymbol{V}}^2 + \Sigma_{\star,2}^{-1}\right)^{-1}\boldsymbol{w}_\star^{\ 2}\right] + \mathbb{E}\left[\boldsymbol{\xi}^{1\top}\left(\hat{\boldsymbol{V}}^1 + \Sigma_{\star,1}^{-1}\right)^{-1}\left(\hat{\boldsymbol{V}}^2 + \Sigma_{\star,2}^{-1}\right)^{-1}\boldsymbol{\xi}^2\right] \tag{32}$$

$$= \hat{m}^1\hat{m}^2 \times Tr\left(Cov(\boldsymbol{w}_\star^{\ 1}, \boldsymbol{w}_\star^{\ 2})\left(\hat{\boldsymbol{V}}^1 + \Sigma_{\star,1}^{-1}\right)^{-1}\left(\hat{\boldsymbol{V}}^2 + \Sigma_{\star,2}^{-1}\right)^{-1}\right) + \hat{Q}_{1,2} \times Tr\left(Cov(\boldsymbol{\xi}^1, \boldsymbol{\xi}^2)\left(\hat{\boldsymbol{V}}^1 + \Sigma_{\star,1}^{-1}\right)^{-1}\left(\hat{\boldsymbol{V}}^2 + \Sigma_{\star,2}^{-1}\right)\right. \tag{33}$$

The last line seems coherent with the expression found by Loureiro et al. when the covariance is $\frac{1}{\lambda}I_d$

Overlap V The overlap V is given by

$$V^j = \frac{1}{p}\mathbb{E}\left[(\boldsymbol{A} + \Sigma_{\star,j}^{-1})^{-1}\right] \tag{34}$$

[3]The $1/p$ factor comes from the scalar product with the input vector.