École Polytechnique Fédérale de Lausanne

Lab. Information, Learning and Physics, IdePHICS
TP IV

Numerical Evaluation of the Calibration and Optimal Regularization of a Logistic Classifier in an Adversarial Setting

Kasimir Tanner - 338213

Supervised by
Bruno Loureiro
Florent Krzakala

Spring 2022

Table of content

1 Introduction

Uncertainty in deep neural networks is an important problem to solve. With increasing popularity of deep neural networks also their usage in sensitive applications - like in medicine - increases. It is important to make sure they are well calibrated.
Producing an estimator that is well-trained against adversarial attacks is another important problem in application safety. We might eventually build self-driving cars and it will be essential that it will not be possible to trick them by using an adversarial attack.
In the TP IV the initial goal was to implement gradient descent to reproduce the separability thresholds and choice of optimal regularization seen in (Clarté, Loureiro, Krzakala, & Zdeborová, 2022) on a logistic classification problem and to compute new separability thresholds and optimal regularization when defending the logistic classification estimator against adversarial attacks.
Namely, an interesting question with regards to adversarial attacks is whether defending against them, is equivalent to increasing regularization.
My contribution is to show numerical results for the separability threshold and the choice of optimal λ for small adversarial training strengths.
In section 2, I describe the data model, the adversarial attacks, my method to solve the ERM problem, the calibration and separability. In section 4, I show and discuss the calibration, optimal regularization and their properties obtained for the ERM classifier. All code is available at `https://github.com/Yezat/TPIV---Logistic-Classification`

2 Theory

2.1 Data model

We consider binary classification with n samples $(\boldsymbol{x}^\mu, y^\mu) \in \mathbb{R}^d \times \{-1, 1\}, \mu = 1, \cdots, n$ are independently drawn from a probit model:

$$f_\star(\boldsymbol{x}) := \mathbb{P}\left(y^\mu = 1 \mid \boldsymbol{x}^\mu\right) = \sigma_\star\left(\frac{\boldsymbol{w}_\star^\top \boldsymbol{x}^\mu}{\tau}\right) \\ \boldsymbol{x}^\mu \sim \mathcal{N}\left(\mathbf{0}, 1/d\mathrm{I}_d\right), \quad \boldsymbol{w}_\star \sim \mathcal{N}\left(\mathbf{0}, \mathrm{I}_d\right) \tag{1}$$

Here, $\sigma_\star(x) = 1/2\,\mathrm{erfc}(-x/\sqrt{2})$ and $\tau \geq 0$. τ is a parameter determining the noise level. The noise is generated independently by sampling $\xi^\mu \sim \mathcal{N}(0, 1)$.
We generate the labels with $y^\mu = \mathrm{sign}\left(\boldsymbol{w}_\star^\top \boldsymbol{x}^\mu + \tau\xi^\mu\right)$.
Let's call $f_\star(\boldsymbol{x})$ and the parameters $\boldsymbol{w}_\star$ the 'teacher', these are the actual parameters of the problem that we want to learn. The student w are the learned parameters. This is the teacher-student setting.
Assume we are given training data according to $\boldsymbol{x} \sim \mathcal{N}\left(\mathbf{0}, 1/d\mathbf{I}_d\right)$ and a set of generated labels, we want to find a probabilistic classifier $x \to \hat{y}(x)$ that minimizes the 0/1 test error:

$$\varepsilon_g = \mathbb{E}_{\boldsymbol{x},y}\mathbb{P}(\hat{y}(x) \neq y) \tag{2}$$

In order to obtain weights, we perform regularized empirical risk minimization. The weights that minimize the risk are chosen as our estimator of w. We hope that they also minimize the 0/1 test error. Hence, we choose a loss function $\mathcal{L} : \mathbb{R} \to \mathbb{R}$ as a convex and smooth approximation to the 0/1 loss. Specifically $\mathcal{L}$ should be convex and decreasing. The logistic loss $\mathcal{L}(x) = \log(1 + x^{-x})$ fulfills these criteria. The logistic loss function comes up when deriving the maximum likelihood estimator for the logit model. (Hastie, Tibshirani, & Friedman, 2009) show how to derive the maximum of the log-likelihood in a general logistic regression setting in chapter 4.4.1.

$$\hat{\mathcal{R}}_n(\boldsymbol{w}) = \frac{1}{n}\sum_{\mu=1}^{n} \log\left(1 + e^{-y^\mu \boldsymbol{w}^\top \boldsymbol{x}^\mu}\right) + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 \tag{3}$$

The weights $\hat{\boldsymbol{w}} \in \mathbb{R}^d$ can then be used to obtain a probabilistic classifier $\hat{f}_{\text{erm}}(\boldsymbol{x}) = \mathbb{P}(y = 1 \mid \boldsymbol{x}) = \sigma\left(\hat{\boldsymbol{w}}_{\text{erm}}^\top \boldsymbol{x}\right)$ where $\sigma(x) = (1 + e^{-x})^{-1}$ which is the logistic function. (Clarté et al., 2022)

2.2 Adversarial Problem

We can train against an adversarial attack by defining an 'attack budget' ε_{tr} and by penalizing the maximum δ_i added to a given training sample x_i. We can formulate this idea as a robust optimization problem:

$$\widehat{\boldsymbol{w}} := \arg\min_{\boldsymbol{w}\in\mathbb{R}^d} \frac{1}{n}\sum_{\mu=1}^{n} \max_{\|\boldsymbol{\delta}^\mu\|_2 \le \varepsilon_{\text{tr}}} \mathcal{L}\left(y^\mu \left\langle \mathbf{x}^\mu + \boldsymbol{\delta}^\mu, \boldsymbol{w}\right\rangle\right) + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 \tag{4}$$

We can solve the inner maximization by choosing $\boldsymbol{\delta}^\mu = -y^\mu \varepsilon_{\text{tr}} \boldsymbol{w}/\|\boldsymbol{w}\|_2$, which leads to this new equivalent empirical risk minimization problem:

$$\min_{\boldsymbol{w}} \hat{\mathcal{R}}_n(\boldsymbol{w}) = \min_{\boldsymbol{w}} \frac{1}{n}\sum_{\mu=1}^{n} \mathcal{L}\left(y^\mu \left\langle \mathbf{x}^\mu, \boldsymbol{w}\right\rangle - \varepsilon_{\text{tr}}\|\boldsymbol{w}\|_2\right) + \frac{\lambda}{2}\|\boldsymbol{w}\|_2^2 \tag{5}$$

(Taheri, Pedarsani, & Thrampoulidis, 2021)
It is possible to show that the ERM problem in equation 5 is convex, for instance by using Danskin's theorem. Intuitively, we can see that the risk is a sum of convex functions.

2.2.1 Proof of Convexity

Definition of convexity Let f be a real function defined on a real interval I, f is convex on I if and only if:

$$f(\alpha x + \beta y) \le \alpha f(x) + \beta f(y)$$
$$\forall x, y \in I : \forall \alpha, \beta \in \mathbb{R}_{\ge 0}, \alpha + \beta = 1$$

(from https://math.stackexchange.com/questions/1370235/prove-that-the-sum-of-convex-functions-is-again-convex)

Instead of using Danskin's theorem, we want to exploit the following lemma. (taken from https://math.stackexchange.com/questions/108393/is-the-composition-of-n-convex-functions-itself-a-convex-function) Lemma: Let $f_1, \dots, f_n$ be convex nondecreasing functions. Then $f_1 \circ \cdots .f_n$ is convex (and nondecreasing). Proof: Here is a proof for the case where each of the f_i are twice-differentiable. We can show it by induction. Let $g_n = f_1 \circ \cdots \circ f_n$. Suppose that g_n is convex and nondecreasing. Then, $g_{n+1} = g_n \cdot f_{n+1}$. But, two applications of the chain rule yield

$$g'_{n+1} = \left(g'_n \circ f_{n+1}\right) f'_{n+1} \geq 0$$

and

$$g''_{n+1} = \left(g''_n \circ f_{n+1}\right) \left(f'_{n+1}\right)^2 + \left(g'_n \circ f_{n+1}\right) f''_n \geq 0$$

and so the stated result follows.

Let's show that the loss $\mathcal{L}$ and the argument $y^\mu \langle \mathbf{x}^\mu, \boldsymbol{w} \rangle - \varepsilon_{\text{tr}} \|\boldsymbol{w}\|_2$ are convex and non-decreasing. Then we can show that our loss is convex as the sum of convex functions is convex, the regularization term is convex and because the per sample loss is convex by application of the lemma above. not convinved that this will work out... How to show non-decreasingnes..? shouldn't be that hard, how in many d though? I guess we need the minus sign of the loss inside the argument?

form https://mathworld.wolfram.com/NondecreasingFunction.html A function $f(x)$ is said to be nondecreasing on an interval I if $f(b) \geq f(a)$ for all $b > a$, where $a, b \in I$. Conversely, a function $f\ (\ x\ )$ is said to be nonincreasing on an interval I if $f(b) \leq f(a)$ for all $b > a$ with $a, b \in I$.

Check this out: https://arxiv.org/pdf/2102.02950.pdf

So to simplify the argument. Say that the argument of the loss is a sum of convex functions. This is because the L2-norm is trivially convex and the other term is linear and thus trivially convex and concave. The loss is well-known to be convex as it is a variant of logsumexp. Any logsumexp function is convex. The theorem above should hold, maybe one needs to flip a sign of sth...

2.3 Gradients and stopping criteria

A straightforward way of finding a solution to the ERM problem is to implement gradient descent. We can exploit the convexity of the ERM problem and implement a line-search where we start with a high learning rate and reduce it as soon as our gradient step would lead us to a point on the loss landscape that is higher than our current point. We only perform the gradient step if we actually end up at a smaller loss.

The gradient can be written as:

$$\frac{\partial R_n}{\partial w_k} = \frac{1}{n} \sum_{i=1}^{n} \frac{-y_i x_{ik} + \varepsilon_{t_r} \cdot w_k / \|w\|_2}{1 + \exp(y_i \langle x_i, \theta_n \rangle - \varepsilon_{t_r} \|w\|_2)} + \lambda w_k \tag{6}$$

We stop gradient descent either when training error is zero, a maximum number of iterations is reached, or when the norm of the gradient or the difference between succeeding losses is smaller than a precision of our choice.

2.4 Calibration

We are interested in finding out how well the trained model is calibrated. The calibration of the ERM classifier $\hat{f}_{erm} : \mathbb{R}^d \to [0,1]$ is defined as:

$$\Delta_p(\hat{f}_{erm}) := p - \mathbb{E}_{\boldsymbol{x}} \left(f_\star(\boldsymbol{x}) \mid \hat{f}_{erm}(\boldsymbol{x}) = p \right) \tag{7}$$

If $\Delta_p > 0$, we call the classifier $\hat{f}_{erm}$ overconfident and for $\Delta_p < 0$ we say it is underconfident at $p > 1/2$, $p \in [0,1]$
Δ_p can be estimated experimentally by

$$\Delta_p \simeq p - \frac{\sum_{i=1}^{n_{\text{test}}} f_\star\left(x_i\right) \mathbf{1}\left(\hat{f}_{\text{erm}}\left(x_i\right) \in [p, p+\mathrm{d}p]\right)}{\sum_{i=1}^{n_{\text{test}}} \mathbf{1}\left(\hat{f}_{\text{erm}}\left(x_i\right) \in [p, p+\mathrm{d}p]\right)}. \tag{8}$$

Alternatively, (Clarté et al., 2022) show how to compute the calibration as a function of the overlaps m and q_{erm}.

$$\Delta_p\left(\hat{f}_{\text{erm}}\right) = p - \sigma_\star\left(\frac{\frac{m}{q_{\text{erm}}} \times \sigma^{-1}(p)}{\sqrt{1 - \frac{m^2}{q_{\text{erm}}} + \tau^2}}\right). \tag{9}$$

The overlaps m, q_{erm} and ρ are defined as follows, they are physics jargon for the norm of the estimators and the correlation between teacher and student.

$$\rho \equiv \frac{1}{d}\left\|\boldsymbol{w}_\star\right\|^2, \quad q_{erm} \equiv \frac{1}{d}\left\|\hat{\boldsymbol{w}}_{\text{erm}}\right\|^2, \quad m \equiv \frac{1}{d}\hat{\boldsymbol{w}}_{\text{erm}} \cdot \boldsymbol{w}_\star. \tag{10}$$

$\boldsymbol{w}_\star$ and $\hat{\boldsymbol{w}}_{erm}$ define an angle β:

$$\cos\beta = \frac{\boldsymbol{w}_\star}{\|\boldsymbol{w}_\star\|} \cdot \frac{\hat{\boldsymbol{w}}_{erm}}{\|\hat{\boldsymbol{w}}_{erm}\|} = \frac{m}{\sqrt{q_{erm} \cdot \rho}}. \tag{11}$$

If $\cos\beta$ is one, the learned weights by ERM match the teacher exactly.

2.5 Separability

We call data $\{(\mathbf{x}^\mu, y^\mu)\}$ linearly separable if and only if

$$\exists \boldsymbol{w} \in \mathbb{R}^d \quad \text{s.t.} \quad y^\mu \left\langle \mathbf{x}^\mu, \boldsymbol{w} \right\rangle \geq 1, \forall i \in [n]. \tag{12}$$

On the other hand, data are called (ℓ_q, ε)-separable if and only if

$$\exists \boldsymbol{w} \in \mathbb{R}^d \quad \text{s.t.} \quad y^\mu \left\langle \mathbf{x}^\mu, \boldsymbol{w} \right\rangle - \varepsilon\|\boldsymbol{w}\|_p \geq 1, \forall i \in [n]. \tag{13}$$

For $\varepsilon = 0$ both definitions of linear separability are equivalent. For $\varepsilon \geq 0$ (ℓ_q, ε)-separability implies $(\ell_q, 0)$-separability, which is equivalent to linear separability.
(Taheri et al., 2021) conjecture that there is a threshold $\alpha_{\psi,\varepsilon,\Pi}$, where $\alpha = n/d$ is the sampling ratio, ψ denotes the logit function, or in general, the link function connecting the mean of the distribution function with the linear predictor in a generalized linear model, and Π is the probability distribution, such that the data

are (ℓ_q, ε)-separable if and only if $\alpha \leq \alpha_{\psi,\varepsilon,\Pi}$. Furthermore, it must satisfy $\alpha_{\psi,\varepsilon,\Pi} \leq \alpha_{\psi,0,\Pi}$.

Note that when setting the regularization $\lambda = 0$ and the adversarial term $\varepsilon = 0$, then the data are $(\ell_q, 0)$-separable if $\alpha \leq \alpha_{\psi,0,\Pi}$. Furthermore, the ERM problem becomes unbounded in this regime. In this case, the argument of the loss is positive for all data.

The ERM problem is also unbounded if $\alpha \leq \alpha_{\psi,\varepsilon,\Pi}$, below this threshold the data are (ℓ_q, ε)-separable.

We can observe these phase transitions by looking at the training error plotted against the sampling ratio α.

3 Replica

3.1 Setting

Let's formalize the setting. We have n samples drawn from the probit model defined above and in practice one would use Empirical Risk Minimization (ERM) to solve for the weights. ERM can be included in a Bayesian framework, where we are interested in inferring the weights from the observations $\{\mathbf{y}, \mathrm{X}\}$.

$$\mathbb{P}(\mathbf{w} \mid \mathbf{y}, \mathrm{X}) = \frac{\mathbb{P}(\mathbf{y} \mid \mathbf{w}, \mathrm{X})\mathbb{P}(\mathbf{w})}{\mathbb{P}(\mathbf{y}, \mathrm{X})}$$

However, this posterior distribution is often not tractable. Sometimes the ground truth likelihood and prior distributions $\mathbb{P}(\mathbf{y} \mid \mathbf{w}, \mathrm{X}) = P_{\text{out}^\star}(\mathbf{y} \mid \mathbf{z})$ with $\mathbf{z} \equiv \frac{1}{\sqrt{d}}\mathrm{X}\mathbf{w}$ and $\mathbb{P}(\mathbf{w}) = P_{\mathrm{w}^\star}(\mathbf{w})$ are known. Estimating the average of the posterior using the ground truth distributions is called Bayes-optimal estimation. This gives us the minimal mean-squared error (MMSE) estimator $\hat{\mathbf{w}}_{\text{mmse}} = \mathbb{E}_{\mathbb{P}(\mathbf{w}|\mathbf{y},\mathrm{X})}[\mathbf{w}]$. To include ERM in the Bayesian Framework, let's recall the estimator $\hat{\mathbf{w}}_{\text{erm}} = \operatorname{argmin}_{\mathbf{w}}[\mathcal{L}(\mathbf{w}; \mathbf{y}, \mathrm{X})]$ and the loss in general

$$\mathcal{L}(\mathbf{w}; \mathbf{y}, \mathrm{X}) = \sum_{\mu=1}^{n} l\left(\mathbf{w}; y_\mu, \mathbf{x}_\mu\right) + r(\mathbf{w})$$

Simply exponentiating the loss shows that minimizing $\mathcal{L}$ corresponds to maximizing the posterior $\mathbb{P}(\mathbf{w} \mid \mathbf{y}, \mathrm{X}) = e^{-\mathcal{L}(\mathbf{w};\mathbf{y},\mathrm{X})}$. Since the likelihood and the prior as functions of the loss l and the regularizer r are known, the Maximum a Posteriori (MAP) estimator is well defined. Hence, the ERM approach and the MAP approach are equivalent. To be explicit:

$$-\log \mathbb{P}(\mathbf{y} \mid \mathbf{w}, \mathrm{X}) = l(\mathbf{w}; \mathbf{y}, \mathrm{X}), \quad -\log \mathbb{P}(\mathbf{w}) = r(\mathbf{w}).$$

Note that training against adversarial attacks assumes a different likelihood distribution than what we assume the original ground truth would have been. I.e. adversarial training assumes the data have been manipulated by an adversary and thus the likelihood i.e. the loss of ERM changes. Remember that the loss is given by

$$\ell\left(w; y, x, \varepsilon\right) = \log\Big(1 + \exp\Big(-yw^\top x + \varepsilon\|w\|_2\Big)\Big) \tag{14}$$

We choose

$$r(w)=\frac{\lambda}{2}\|w\|_2^2 \tag{15}$$

Let's define the priors and the channel distributions. Functions denoted with * are teacher functions, the others for the student. The prior does not change, hence the teacher matches the student.

$$P_w(h)=P_w^*(h)=\frac{1}{2\pi^{d/2}}\cdot\exp\left(-\frac{1}{2}h^2\right) \tag{16}$$

Note that this is simply saying that the ground truth distributions of the weights in this model is gaussian. Let's define the channel distributions: P_{out}, P_{out}^* with $z^*=w^{\top *}x$ and $z=\hat{w}^\top x$

$$P_{\text{out}}^*\ (y\mid z^*)=\operatorname{sign}\left(\sqrt{\frac{\tau}{2\pi}}\exp\left(-\tau\left(x-z^*\right)/2\right)\right) \tag{17}$$

Is this correct? In general, double check τ vs Δ. In general the noise level is not yet explicit in the following computation... This essentially corresponds to the likelihood of the teacher, as the data is generated accoring to a gaussian with some noise/variance τ. Let's define the channel distribution of the student, this is a good time to introduce the finite temperature Δ.

$$P_{out}\left(y\mid z,w,\varepsilon\right)=P_{out}\left(y\mid w,x,\varepsilon\right)=\frac{\exp(-\frac{1}{\Delta}\ell(w;y,x,\varepsilon))}{\sqrt{2\pi\Delta}} \tag{18}$$

Note that the introduced temperature Δ is fictive. For the Bayes-optimal setting the temperature equals 1. However, it is useful as it gives us another way of minimizing the loss $\mathcal{L}$, i.e. maximizing the posterior, we achieve this by taking the zero temperature limit $\Delta\to 0$. this may be a good location to delve into denoising stuff and definitions of moreau-yosida ...

Let's continue by defining the corresponding partition functions.

$$\begin{aligned}\mathcal{Z}(\mathbf{y},\mathrm{X})\equiv P(\mathbf{y},\mathrm{X})&=\int_{\mathbb{R}^d}\mathrm{d}\mathbf{w}P_{\text{out}}\left(\mathbf{y}\mid\mathbf{w},\mathrm{X}\right)P_{\mathrm{w}}(\mathbf{w})\\&=\int_{\mathbb{R}^n}\mathrm{d}\mathbf{z}P_{\text{out}}\left(\mathbf{y}\mid\mathbf{z}\right)\int_{\mathbb{R}^d}\mathrm{d}\mathbf{w}P_{\mathrm{w}}(\mathbf{w})\delta\left(\mathbf{z}-\frac{1}{\sqrt{d}}\mathrm{X}\mathbf{w}\right)\end{aligned} \tag{19}$$

where $\mathbf{z}=\frac{1}{\sqrt{d}}X\mathbf{w}$. We are interested in the high-dimensional limit $d\to\infty, n\to\infty, \alpha=\Theta(1)$. For this we want the average free entropy Φ

$$\Phi(\alpha)\equiv\mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\lim_{d\to\infty}\frac{1}{d}\log\mathcal{Z}(\mathbf{y},\mathbf{X})\right]$$

This average is generally not tractable. Hence, we first apply the replica trick

$$\mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\lim_{d\to\infty}\frac{1}{d}\log\mathcal{Z}(\mathbf{y},\mathrm{X})\right]=\lim_{r\to 0}\left[\lim_{d\to\infty}\frac{1}{d}\frac{\partial\log\mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\mathcal{Z}(\mathbf{y},\mathrm{X})^r\right]}{\partial r}\right] \tag{20}$$

We then proceed evaluating $\mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\mathcal{Z}(\mathbf{y},\mathrm{X})^r\right]$.

$$\begin{aligned}
&\mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\mathcal{Z}(\mathbf{y},\mathrm{X})^r\right] = \mathbb{E}_{\mathbf{w}^\star,\mathrm{X}}\left[\prod_{a=1}^{r}\int_{\mathbb{R}^n}\,\mathrm{d}\mathbf{z}^a P_{\mathrm{out}^a}\left(\mathbf{y}\mid\mathbf{z}^a\right)\int_{\mathbb{R}^d}\,\mathrm{d}\mathbf{w}^a P_{\mathrm{w}^a}\left(\mathbf{w}^a\right)\delta\left(\mathbf{z}^a-\frac{1}{\sqrt{d}}\mathrm{X}\mathbf{w}^a\right)\right]\\
&= \mathbb{E}_{\mathrm{X}}\int_{\mathbb{R}^n}\,\mathrm{d}\mathbf{y}\int_{\mathbb{R}^n}\,\mathrm{d}\mathbf{z}^\star P_{\mathrm{out}\ \star}\left(\mathbf{y}\mid\mathbf{z}^\star\right)\int_{\mathbb{R}^d}\,\mathrm{d}\mathbf{w}^\star P_{\mathrm{w}^\star}\left(\mathbf{w}^\star\right)\delta\left(\mathbf{z}^\star-\frac{1}{\sqrt{d}}\mathbf{X}\mathbf{w}^\star\right)\\
&\times\left[\prod_{a=1}^{r}\int_{\mathbb{R}^n}\,\mathrm{d}\mathbf{z}^a P_{\mathrm{out}^a}\left(\mathbf{y}\mid\mathbf{z}^a,\mathbf{w}^a,\varepsilon\right)\int_{\mathbb{R}^d}\,\mathrm{d}\mathbf{w}^a P_{\mathrm{w}^a}\left(\mathbf{w}^a\right)\delta\left(\mathbf{z}^a-\frac{1}{\sqrt{d}}\mathbf{X}\mathbf{w}^a\right)\right]
\end{aligned} \tag{21}$$

$$\begin{aligned}
&\mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\mathcal{Z}(\mathbf{y},\mathrm{X})^r\right] =\\
&\mathbb{E}_{\mathrm{X}}\int_{\mathbb{R}^n}\,\mathrm{d}\mathbf{y}\prod_{a=0}^{r}\int_{\mathbb{R}^n}\,\mathrm{d}\mathbf{z}^a P_{\mathrm{out}^0}\left(\mathbf{y}\mid\mathbf{z}^0\right)\cdot P_{\mathrm{out}^b}\left(\mathbf{y}\mid\mathbf{z}^b,q,\varepsilon\right)\int_{\mathbb{R}^d}\,\mathrm{d}\mathbf{w}^a P_{\mathrm{w}^a}\left(\mathbf{w}^a\right)\delta\left(\mathbf{z}^a-\frac{1}{\sqrt{d}}\mathbf{X}\mathbf{w}^a\right)\\
&= \left[\int_{\mathbb{R}}\mathrm{d}y\int_{\mathbb{R}}\,\mathrm{d}\tilde{\mathbf{z}}_0 P_{\mathrm{out}}\ (y\mid\tilde{\mathbf{z}}_0)P_{\tilde{\mathbf{z}}_0}(\tilde{\mathbf{z}}_0;Q(\tilde{\mathbf{w}}))\int_{\mathbb{R}^r}\,\mathrm{d}\tilde{\mathbf{z}}\cdot P_{\mathrm{out}}\ (y\mid\tilde{\mathbf{z}},q,\varepsilon)P_{\tilde{\mathbf{z}}}(\tilde{\mathbf{z}};Q(\tilde{\mathbf{w}}))\right]^n\left[\int_{\mathbb{R}^{r+1}}\,\mathrm{d}\tilde{\mathbf{w}}P_{\tilde{\mathbf{w}}}(\tilde{\mathbf{w}})\right]^d
\end{aligned} \tag{22}$$

Let b run from 1 to r. Notice we already plugged in the replica symmetric ansatz for the overlap matrix $Q\left(\{\mathbf{w}^a\}\right)\equiv\left(\frac{1}{d}\mathbf{w}^a\cdot\mathbf{w}^b\right)_{a,b=0..r}$. We justify this ansatz by supposing that inputs are iid distributed, for instance Gaussian $\mathcal{N}(0,1)$. For $i,j\in[1:d],\mu,\nu\in[1:n],\mathbb{E}_{\mathrm{X}}\left[x_i^{(\mu)}x_j^{(\nu)}\right]=\delta_{\mu\nu}\delta_{ij}$. Hence $z_\mu^a=\frac{1}{\sqrt{d}}\sum_{i=1}^d x_i^{(\mu)}w_i^a$ is the sum of iid random variables. By the central limit theorem $z_\mu^a\sim\mathcal{N}\left(\mathbb{E}_{\mathrm{X}}\left[z_\mu^a\right],\mathbb{E}_{\mathrm{X}}\left[z_\mu^a z_\mu^b\right]\right)$, where the first two moments are

$$\begin{cases}\mathbb{E}_{\mathrm{X}}\left[z_\mu^a\right]=\frac{1}{\sqrt{d}}\sum_{i=1}^d\mathbb{E}_{\mathrm{X}}\left[x_i^{(\mu)}\right]w_i^a=0\\ \mathbb{E}_{\mathrm{X}}\left[z_\mu^a z_\mu^b\right]=\frac{1}{d}\sum_{ij}\mathbb{E}_{\mathrm{X}}\left[x_i^{(\mu)}x_j^{(\mu)}\right]w_i^a w_j^b=\frac{1}{d}\sum_{ij}\delta_{ij}w_i^a w_j^b=\frac{1}{d}\mathbf{w}^a\cdot\mathbf{w}^b\end{cases}$$

We define $\tilde{\mathbf{z}}_\mu\equiv\left(z_\mu^a\right)_{a=0..r}$ and $\tilde{\mathbf{w}}_i\equiv\left(w_i^a\right)_{a=0..r}$ and their distributions $\tilde{\mathbf{z}}_\mu\sim P_{\tilde{\mathbf{z}}}(\tilde{\mathbf{z}};Q)=\mathcal{N}_{\tilde{\mathbf{z}}}\left(\mathbf{0}_{r+1},Q\right)$ and $P_{\tilde{\mathbf{w}}}(\tilde{\mathbf{w}})=\prod_{a=0}^r P_{\mathrm{w}}\left(\tilde{w}^a\right)$.
Now we perform a change of variable and introduce the δ-Dirac distribution represented as a Fourier transform which involves the new parameter $\hat{Q}$.

$$\begin{aligned}
1 &= \int_{\mathbb{R}^{r+1\times r+1}}\,\mathrm{d}Q\prod_{a\le b}\delta\left(dQ_{ab}-\sum_{i=1}^d w_i^a w_i^b\right)\\
&\propto\int_{\mathbb{R}^{r+1\times r+1}}\,\mathrm{d}Q\int_{\mathbb{R}^{r+1\times r+1}}\,\mathrm{d}\hat{Q}\exp(-d\,\mathrm{Tr}[Q\hat{Q}])\exp\left(\frac{1}{2}\sum_{i=1}^d\tilde{\mathbf{w}}_i^\top\hat{Q}\tilde{\mathbf{w}}_i\right).
\end{aligned} \tag{23}$$

This leaves the replicated partition function as an integral over the symmetric matrices $Q\in\mathbb{R}^{r+1\times r+1}$ and $\hat{Q}\in\mathbb{R}^{r+1\times r+1}$. We evaluate it using the Laplace method in the $d\to\infty$ limit.

$$\begin{aligned}
\mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\mathcal{Z}(\mathbf{y},\mathrm{X})^r\right] &= \int_{\mathbb{R}^{r+1\times r+1}}\,\mathrm{d}Q\int_{\mathbb{R}^{r+1\times r+1}}\,\mathrm{d}\hat{Q}e^{d\Phi^{(r)}(Q,\hat{Q})}\\
&\underset{d\to\infty}{\simeq}\exp\left(d\cdot\mathrm{extr}_{Q,\hat{Q}}\left\{\Phi^{(r)}(Q,\hat{Q})\right\}\right)
\end{aligned} \tag{24}$$

where we defined

$$\begin{cases} \Phi^{(r)}(Q,\hat{Q}) = -\operatorname{Tr}[Q\hat{Q}] + \log\Psi_{\mathrm{w}}^{(r)}(\hat{Q}) + \alpha\log\Psi_{\text{out}}^{(r)}\,(Q) \\ \Psi_{\mathrm{w}}^{(r)}(\hat{Q}) = \int_{\mathbb{R}^{r+1}}\ \mathrm{d}\tilde{\mathbf{w}}P_{\tilde{\mathbf{w}}}(\tilde{\mathbf{w}})e^{\frac{1}{2}\tilde{\mathbf{w}}\tilde{Q}\hat{Q}\tilde{\mathbf{w}}} \\ \Psi_{\text{out}}^{(r)}\,(Q) = \int_{\mathbb{R}}\mathrm{d}y\int_{\mathbb{R}}\mathrm{d}\tilde{\mathbf{z}}_0 P_{\text{out}}\,\left(y\mid\tilde{\mathbf{z}}_0\right)P_{\tilde{\mathbf{z}}_0}\left(\tilde{\mathbf{z}}_0;Q(\tilde{\mathbf{w}})\right)\int_{\mathbb{R}^r}\ \mathrm{d}\tilde{\mathbf{z}}\cdot P_{\text{out}}\,\left(y\mid\tilde{\mathbf{z}},q,\varepsilon\right)P_{\tilde{\mathbf{z}}}(\tilde{\mathbf{z}};Q(\tilde{\mathbf{w}})) \end{cases} \tag{25}$$

and $P_{\tilde{z}}(\tilde{\mathbf{z}};Q) = \frac{e^{-\frac{1}{2}\tilde{\mathbf{z}}^\top Q^{-1}\tilde{\mathbf{z}}}}{\det(2\pi Q)^{1/2}}$ We now switch the limits $r\to 0$ and $d\to\infty$, leading us the quenched free entropy Φ as a saddle point equation

$$\Phi(\alpha) = \operatorname{extr}_{Q,\hat{Q}}\left\{\lim_{r\to 0}\frac{\partial\Phi^{(r)}(Q,\hat{Q})}{\partial r}\right\},$$

over symmetric matrices $Q\in\mathbb{R}^{r+1\times r+1}$ and $\hat{Q}\in\mathbb{R}^{r+1\times r+1}$. Let's propose the following replica symmetric ansatz for these matrices to be able to compute the derivative in r. We assume that all replica remain equivalent with a common overlap $q = \frac{1}{d}\mathbf{w}^a\cdot\mathbf{w}^b$ for $a\neq b$, a norm $Q = \frac{1}{d}\left\|\mathbf{w}^a\right\|_2^2$, and an overlap with the ground truth $m = \frac{1}{d}\mathbf{w}^a\cdot\mathbf{w}^\star$

$$Q_{\text{rs}} = \begin{pmatrix} Q^0 & m & \dots & m \\ m & Q & \dots & \dots \\ \dots & \dots & \dots & q \\ m & \dots & q & Q \end{pmatrix} \quad\text{and}\quad \hat{Q}_{\text{rs}} = \begin{pmatrix} \hat{Q}^0 & \hat{m} & \dots & \hat{m} \\ \hat{m} & -\frac{1}{2}\hat{Q} & \dots & \dots \\ \dots & \dots & \dots & \hat{q} \\ \hat{m} & \dots & \hat{q} & -\frac{1}{2}\hat{Q} \end{pmatrix}$$

with $Q^0 = \rho_{\mathrm{w}^\star} = \frac{1}{d}\left\|\mathbf{w}^\star\right\|_2^2$. Now we are ready to compute the components of the free entropy functional $\Phi^{(r)}(Q,\hat{Q})$. We have a trace term, a term $\Psi_{\mathrm{w}}^{(r)}$ depending on the prior distributions $P_{\mathrm{w}}, P_{\mathrm{w}^*}$ and a term $\Psi_{\text{out}}^{(r)}$ that depends on the channel distributions. Let's start with the trace term:

$$\operatorname{Tr}(Q\hat{Q})\Big|_{rs} = Q^0\hat{Q}^0 + rm\hat{m} - \frac{1}{2}rQ\hat{Q} + \frac{r(r-1)}{2}q\hat{q}. \tag{26}$$

The prior term remains as in (Aubin, Krzakala, Lu, & Zdeborová, 2020). Note the usage of the Hubbard-Stratonovich transformation $\mathbb{E}_\xi\exp(\sqrt{a}\xi) = e^{\frac{a}{2}}$.

$$\begin{aligned} \Psi_{\mathrm{w}}^{(r)}(\hat{Q})\Big|_{\text{rs}} &= \int_{\mathbb{R}^{r+1}} d\tilde{\mathbf{w}}P_{\tilde{\mathbf{w}}}(\tilde{\mathbf{w}})e^{\frac{1}{2}\tilde{\mathbf{w}}^\top\hat{Q}_{\text{rs}}\tilde{\mathbf{w}}} \\ &= \mathbb{E}_{w^\star}e^{\frac{1}{2}\hat{Q}^0(w^\star)^2}\int_{\mathbb{R}^r} d\tilde{\mathbf{w}}P_{\tilde{\mathbf{w}}}(\tilde{\mathbf{w}})e^{w^\star\hat{m}\sum_{a=1}^r\tilde{w}^a - \frac{1}{2}(\hat{Q}+\hat{q})\sum_{a=1}^r(\tilde{w}^a)^2 + \frac{1}{2}\hat{q}\left(\sum_{a=1}^r\tilde{w}^a\right)^2} \\ &= \mathbb{E}_{\xi,w^\star}e^{\frac{1}{2}\hat{Q}^0(w^\star)^2}\left[\mathbb{E}_w\exp\left(\left[\hat{m}w^\star w - \frac{1}{2}(\hat{Q}+\hat{q})w^2 + \hat{q}^{1/2}\xi w\right]\right)\right]^r \end{aligned} \tag{27}$$

The interesting part is the channel integral. To evaluate this, we introduce the following inverse matrix of Q_{rs}:

$$Q_{\text{rs}}^{-1} = \begin{bmatrix} Q_{00}^{-1} & Q_{01}^{-1} & Q_{01}^{-1} & Q_{01}^{-1} \\ Q_{01}^{-1} & Q_{11}^{-1} & Q_{12}^{-1} & Q_{12}^{-1} \\ Q_{01}^{-1} & Q_{12}^{-1} & Q_{11}^{-1} & Q_{12}^{-1} \\ Q_{01}^{-1} & Q_{12}^{-1} & Q_{12}^{-1} & Q_{11}^{-1} \end{bmatrix} \tag{28}$$

The components are given by

$$\begin{cases} Q_{00}^{-1} = \left(Q^0 - rm(Q+(r-1)q)^{-1}m\right)^{-1} \\ Q_{01}^{-1} = -\left(Q^0 - rm(Q+(r-1)q)^{-1}m\right)^{-1} m(Q+(r-1)q)^{-1} \\ Q_{11}^{-1} = (Q-q)^{-1} - (Q+(r-1)q)^{-1}q(Q-q)^{-1} \\ \qquad + (Q+(r-1)q)^{-1}m\left(Q^0 - rm(Q+(r-1)q)^{-1}m\right)^{-1} m(Q+(r-1)q)^{-1} \\ Q_{12}^{-1} = -(Q+(r-1)q)^{-1}q(Q-q)^{-1} \\ \qquad + (Q+(r-1)q)^{-1}m\left(Q^0 - rm(Q+(r-1)q)^{-1}m\right)^{-1} m(Q+(r-1)q)^{-1} \end{cases} \tag{29}$$

and the determinant by

$$\det Q_{\mathrm{rs}} = (Q-q)^{r-1}(Q+(r-1)q)\left(Q^0 - rm(Q+(r-1)q)^{-1}m\right). \tag{30}$$

Again using Hubbard-Stratanovich, we obtain

$$\begin{aligned} &\Psi_{\text{out}}^{(r)}(Q)\Big|_{\text{rs}} = \int \mathrm{d}y \int_{\mathbb{R}^{r+1}} d\tilde{\mathbf{z}} e^{-\frac{1}{2}\tilde{\mathbf{z}}^\top Q_{\text{rs}}^{-1}\tilde{\mathbf{z}} - \frac{1}{2}\log(\det(2\pi Q_{\text{rs}}))} P_{\text{out}}(y \mid \tilde{\mathbf{z}}, q, \varepsilon) \\ &= \mathbb{E}_{y,\xi} \mathrm{e}^{-\frac{1}{2}\log(\det(2\pi Q_{\text{rs}}))} \\ &\times \int \mathrm{d}z^\star P_{\text{out}}{}^\star(y \mid z^\star)\, e^{-\frac{1}{2}Q_{00}^{-1}(z^\star)^2} \left[\int dz P_{\text{out}}(y \mid z, q, \varepsilon) e^{-Q_{01}^{-1}z^\star z - \frac{1}{2}\left(Q_{11}^{-1} - Q_{12}^{-1}\right)z^2 - Q_{12}^{-1/2}\xi z}\right]^r \end{aligned} \tag{31}$$

In the case where the loss function is convex and the regularization strength λ is positive, we can justify the replica symmetric ansatz. Convexity implies that there is only one solution and thus it must coincide with the replica symmetric one. If the loss function was not convex, we would resort to replica symmetry breaking which would be more involved. However, we first need to verify that our chosen Ansatz is well-defined. To do this, we need to show that $\lim_{r\to 0^+}\Phi = 0$. This ensures, that we did not introduce a diverging term. With some algebra it is possible to show that $\lim_{r\to 0^+}\Psi_w^{(r)} = 0$. Therefore, to continue... Not fully elaborated...

$$\lim_{r\to 0^+}\Phi^{(r)} = -Q^0\hat{Q}^0 + \alpha\log\int_{\mathbb{R}} \mathrm{d}\theta^0 P_\theta\left(\theta^0\right) e^{\hat{\rho}\theta^{0^2}} \tag{32}$$

where we have used the fact that P_θ is a factorised distribution to take the $p\to\infty$ limit. In order for this limit to be 0 , we need that $\hat{\rho}=0$, which also fixes ρ to be a constant given by the second moment of θ^0 :

$$\rho = \mathbb{E}_{\theta^0}\left[\theta^{0^2}\right]$$

The above should be a formality. Let's continue... The takeaway is that $\hat{Q}^0 = 0$. Hence, we can write the free-entropy as

$$\begin{aligned} \Phi_{\text{rs}}(\alpha) &\equiv \mathbb{E}_{\mathbf{y},\mathrm{X}}\left[\lim_{d\to\infty}\frac{1}{d}\log(\mathcal{Z}(\mathbf{y},\mathrm{X}))\right] \\ &= \operatorname{extr}_{Q,\hat{Q},q,\hat{q},m,\hat{m}}\left\{-m\hat{m} + \frac{1}{2}Q\hat{Q} + \frac{1}{2}q\hat{q} + \Psi_{\mathrm{w}}(\hat{Q},\hat{m},\hat{q}) + \alpha\Psi_{\text{out}}(Q,m,q;\rho_{\mathrm{w}^\star},\varepsilon)\right\} \end{aligned} \tag{33}$$

where $\rho_{\mathrm{w}^\star} = \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star} \frac{1}{d} \left\|\mathbf{w}^\star\right\|_2^2 = 1$. The prior and the channel terms can be written as

$$\begin{aligned}\Psi_{\mathrm{w}}(\hat{Q},\hat{m},\hat{q}) &\equiv \mathbb{E}_\xi \left[\mathcal{Z}_{\mathrm{w}^\star}\left(\hat{m}\hat{q}^{-1/2}\xi, \hat{m}\hat{q}^{-1}\hat{m}\right) \log \mathcal{Z}_{\mathrm{w}}\left(\hat{q}^{1/2}\xi, \hat{Q}+\hat{q}\right)\right] \\ \Psi_{\mathrm{out}}\ (Q,m,q;\rho_{\mathrm{w}^\star}) &\equiv \mathbb{E}_{y,\xi}\left[\mathcal{Z}_{\mathrm{out}\ ^\star}\left(y, mq^{-1/2}\xi, \rho_{\mathrm{w}^\star} - mq^{-1}m\right) \log \mathcal{Z}_{\mathrm{out}}\ \left(y, q^{1/2}\xi, Q-q, q, \varepsilon\right)\right]\end{aligned} \tag{34}$$

was this rigorous enough?

Note that we define

$$\mathcal{Z}_{out^{\cdot/*}}(y;\omega,V,q,\varepsilon) = \int \frac{\mathrm{d}x}{\sqrt{2\pi V}} e^{-\frac{1}{2V}(x-\omega)^2} P_y^{\cdot/0}(y \mid x,q,\varepsilon) \tag{35}$$

Define also Z_w

In the following we will be interested in derivatives of the partition function w.r.t to it's parameters. Hence this is a good moment to introduce an alternative equivalent form for the Maximum a Posteriori partition function:

$$\mathcal{Z}_{\mathrm{out}}^{\mathrm{map}}(y,\omega,V) = \lim_{\Delta\to 0} \int_{\mathbb{R}} \mathrm{d}z Q_{\mathrm{out}}^{\mathrm{map}}(z;y,\omega,V) = \lim_{\Delta\to 0} \frac{e^{-\frac{1}{\Delta}\mathcal{M}_{V_\dagger}[l(y,.)](\omega)}}{\sqrt{2\pi\Delta V_\dagger}\sqrt{2\pi\Delta}}. \tag{36}$$

Note that the Moreau-Yosida regularization $\mathcal{M}_\Sigma$ and the proximal map $\mathcal{P}_\Sigma$ are given by

$$\begin{aligned}\mathcal{P}_\Sigma[f(,.)](x) &= \mathrm{argmin}_z\, \mathcal{L}_\Sigma[f(,.)](z;x) = \mathrm{argmin}_z \left[f(,z) + \frac{1}{2\Sigma}(z-x)^2\right] \\ \mathcal{M}_\Sigma[f(,.)](x) &= \min_z \mathcal{L}_\Sigma[f(,.)](z;x) = \min_z \left[f(,z) + \frac{1}{2\Sigma}(z-x)^2\right]\end{aligned}$$

Q_{out}^{map} is the channel denoising distribution and it's given by

$$\begin{aligned}Q_{\mathrm{out}}^{\mathrm{map}}(z;y,\omega,V) \equiv \lim_{\Delta\to 0} \frac{e^{-\frac{1}{\Delta}l(y,z)+\frac{1}{2V}(z-\omega)^2}}{\sqrt{2\pi\Delta V_\dagger}\sqrt{2\pi\Delta}} &= \lim_{\Delta\to 0} \frac{e^{-\frac{1}{\Delta}\mathcal{L}_{V_\dagger}[l(y,.)](z;\omega)}}{\sqrt{2\pi\Delta V_\dagger}\sqrt{2\pi\Delta}}, \\ \propto \delta\left(z - \mathcal{P}_{V_+}[l(y,.)](\omega)\right)&\end{aligned} \tag{37}$$

Explain why these show up... and properly include ε and q

describe the proper scaling

make sure to use Sigma instead of V+ and use ell instead of l To obtain the fixed point equations in the following, we need the derivative of the partition function $\mathcal{Z}_{out}$ w.r.t q. As it depends on the Moreau-yosida regularization, we get the following:

$$\partial_q \mathcal{Z}_{out} = \lim_{\Delta\to 0} \frac{\exp\left(-\frac{1}{\Delta}\mathcal{M}_{V^+}[l(y,z,q,\varepsilon)](w)\right)}{\sqrt{2\pi\Delta V_+}\sqrt{2\pi\Delta}} \cdot \left(-\frac{1}{\Delta}\right) \cdot \partial_q \mathcal{M}_{V^+}[l(y,z,q,\varepsilon)](w) \tag{38}$$

Next, we need to compute the derivative of the Moreau-Yosida regularization.

$$\partial_q \mathcal{M}_{V^+}[\ell(y,z,q,\varepsilon)](w) = \partial_q \min_z \left[\ell(y,z,q,\varepsilon) + \frac{1}{2V^+}(z-w)^2\right] \tag{39}$$

A minimum z^* must satisfy $\partial_z \ell\left(y, z^*, q, \varepsilon\right) + \frac{1}{V^+}\left(z^* - w\right) = 0$ Writing the derivative explicitly leads

$$\partial_q \mathcal{M}_{V^+} = \lim_{h \to 0} \frac{1}{h}\left(\mathcal{M}_{V^+}[\ell(y, z, q+h, \varepsilon)](\omega) - \mathcal{M}_{V^+}[\ell(y, z, q, \varepsilon)](\omega)\right) \tag{40}$$

Thus, this minimum will be at $z = z^* + dz$ in the limit $h \to 0$ and thus z will satisfy $\partial_z \ell\left(y, z^+ + dz, q+h, \varepsilon\right) + \frac{1}{V^+}\left(z^* + dz - w\right) = 0$. Expanding this expression around h before taking the limit allows us to obtain an expression of dz in term of h.

$$\partial_z \ell\left(y, z^*, q, \varepsilon\right) + \partial_q \partial_z \ell\left(y, z^*, q, \varepsilon\right) \cdot h \quad + \frac{1}{V^+}\left(z^* + dz - \omega\right) = 0 \tag{41}$$

Giving

$$\frac{\Delta}{V} dz = -\partial_q \partial_z \ell\left(y, z^*, q, \varepsilon\right) \cdot h \tag{42}$$

Finally, we can expand expressions in the explicit derivative, cancel some terms and take the $h \to 0$ limit

$$\begin{aligned}
\partial_q \mathcal{M}_{V^+} &= \lim_{h \to 0} \frac{1}{h}\left[\ell\left(y, z^* + dz, q+h, \varepsilon\right) + \frac{1}{2V^+}\left(z^* + dz - w\right)^2 - \ell\left(y, z^*, q, \varepsilon\right) + \frac{1}{2V^+}\left(z^* - w\right)^2\right] \\
&= \lim_{h \to 0} \frac{1}{h}\left[\ell(y, z^*, q+h, \varepsilon) + \partial_z \ell\left(y, z^*, q+h, \varepsilon\right) \cdot dz + \frac{1}{2V^+}\left(z^{*2} + 2z^* dz - 2z^* w + dz^2 - 2wdz\right.\right. \\
&\quad \left.\left. + w^2\right) - \ell\left(y, z^*, q, \varepsilon\right) + \frac{1}{2V^+}\left(z^{*2} - 2wz^* + w^2\right)\right] \\
&= \lim_{h \to 0} \frac{1}{h}\left[\partial_z \ell\left(y, z^*, q+h, \varepsilon\right) dz + \frac{1}{2v^+}\left(2z^* - 2w\right) dz\right]
\end{aligned} \tag{43}$$

Plug in the expression for dz.

$$\partial_q \mathcal{M}_{V^+} = -\left(\partial_z \ell\left(y, z^*, q, \varepsilon\right) \cdot V^+ + \left(z^* - \omega\right)\right) \cdot \partial_q \partial_z \ell\left(y, z^*, q, \varepsilon\right) \tag{44}$$

Finally, we get

$$\begin{aligned}
\partial_q \mathcal{Z}_{\text{out}} &= \mathcal{Z}_{\text{out}} \cdot -\frac{1}{\Delta} \cdot \partial_q \mathcal{M}_{V^+} \\
&= -\mathcal{Z}_{out} \cdot \left(\frac{y\varepsilon}{\sqrt{q} \cdot (4 + 4\cosh(yz - \varepsilon\sqrt{q}))}(z^* - w - \frac{yV}{1 + \exp(yz - \varepsilon\sqrt{q})})\right)
\end{aligned} \tag{45}$$

formalize the scaling...

3.2 Fixed Point Equations

$$\begin{aligned}
\hat{Q} &= -2\alpha \partial_Q \Psi_{\text{out}}\,, & Q &= -2\partial_{\hat{Q}} \Psi_{\text{w}} \\
\hat{q} &= -2\alpha \partial_q \Psi_{\text{out}}\,, & q &= -2\partial_{\hat{q}} \Psi_{\text{w}} \\
\hat{m} &= \alpha \partial_m \Psi_{\text{out}}\,, & m &= \partial_{\hat{m}} \Psi_{\text{w}}
\end{aligned} \tag{46}$$

$$\begin{aligned}
m &= \mathbb{E}_{\xi}\left[\mathcal{Z}_{\mathrm{w}^\star}(\sqrt{\hat{\eta}}\xi,\hat{\eta}) f_{\mathrm{w}^\star}(\sqrt{\hat{\eta}}\xi,\hat{\eta}) f_{\mathrm{w}}\left(\hat{q}^{1/2}\xi,\hat{\Sigma}\right)\right], \\
q &= \mathbb{E}_{\xi}\left[\mathcal{Z}_{\mathrm{w}^\star}(\sqrt{\hat{\eta}}\xi,\hat{\eta}) f_{\mathrm{w}}\left(\hat{q}^{1/2}\xi,\hat{\Sigma}\right)^2\right], \\
\Sigma &= \mathbb{E}_{\xi}\left[\mathcal{Z}_{\mathrm{w}^\star}(\sqrt{\hat{\eta}}\xi,\hat{\eta}) \partial_\gamma f_{\mathrm{w}}\left(\hat{q}^{1/2}\xi,\hat{\Sigma}\right)\right], \\
\hat{m} &= \alpha\mathbb{E}_{y,\xi}\left[\mathcal{Z}_{\mathrm{out}\ ^\star}(.)\cdot f_{\mathrm{out}\ ^\star}\left(y,\sqrt{\rho_{\mathrm{w}^\star}\eta}\xi,\rho_{\mathrm{w}^\star}(1-\eta)\right) f_{\mathrm{out}}\ \left(y,q^{1/2}\xi,\Sigma,q,\varepsilon\right)\right], \\
\hat{q} &= \alpha\mathbb{E}_{y,\xi}\left[\mathcal{Z}^*_{\mathrm{out}}\ \left(y,\sqrt{\rho_{\mathrm{w}^\star}\eta}\xi,\rho_{\mathrm{w}^\star}(1-\eta)\right)\left(f_{\mathrm{out}}\ \left(y,q^{1/2}\xi,\Sigma\right)^2 - \partial_q M_\Sigma\right)\right], \\
\hat{\Sigma} &= -\alpha\mathbb{E}_{y,\xi}\left[\mathcal{Z}_{\mathrm{out}}\ ^\star\left(y,\sqrt{\rho_{\mathrm{w}^\star}\eta}\xi,\rho_{\mathrm{w}^\star}(1-\eta)\right)\partial_\omega f_{\mathrm{out}}\ \left(y,q^{1/2}\xi,\Sigma,q,\varepsilon\right)\right],
\end{aligned} \tag{47}$$

Where we define $\Sigma = Q - q, \hat{\Sigma} = \hat{Q} + \hat{q}, \eta \equiv \frac{m^2}{\rho_{\mathrm{w}^\star} q}$ and $\hat{\eta} \equiv \frac{\hat{m}^2}{\hat{q}}$. Did I properly define the denoising functions? $f_{\mathrm{out}\ ^\star}, f_{\mathrm{w}^\star}, f_{\mathrm{out}}\ , f_{\mathrm{w}}$
Here are the denoising functions in more detail

$$\begin{array}{ll}
f_{\omega *}(\gamma,\Lambda) \equiv \partial_\gamma \log \mathcal{Z}_{\omega^*}(\gamma,\Lambda) & f_{out}(y,\omega,v) = \partial_\omega \log \mathcal{Z}_{out^*}(y,\omega,v) \\
f_w(\gamma,\Lambda) \equiv \partial_\gamma \log \mathcal{Z}_\omega(\gamma,\Lambda) & f_{out}(y,\omega,V,q,\varepsilon) = \partial_\omega \log z_{out}(y,\omega,V,Q,\varepsilon)
\end{array} \tag{48}$$

3.3 Derivatives for fixed point equations

In order to find the fixed point equations, we need the partial derivatives of the partition function:

$$\begin{aligned}
\partial_\omega \mathcal{Z}_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon) &= \mathcal{Z}_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon) \times V^{-1}\mathbb{E}_{Q_{\mathrm{out}}}\ [z-\omega] \\
&= \mathcal{Z}_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon) f_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon) \\
\partial_V \mathcal{Z}_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon) &= \frac{1}{2}\mathcal{Z}_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon) \times \left(\mathbb{E}_{Q_{\mathrm{out}}}\ \left[V^{-2}(z-\omega)^2\right] - V^{-1}\right) \\
&= \frac{1}{2}\mathcal{Z}_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon)\left(\partial_\omega f_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon) + f^2_{\mathrm{out}}\ (y,\omega,V,q,\varepsilon)\right)
\end{aligned} \tag{49}$$

The equations over $\hat{\Sigma}$ and $\hat{m}$ do not change w.r.t. to Aubin apart from two more parameters in the denoising functions, only the equation over $\hat{q}$ changes.
With the above derivation of the partial derivative of the partition function, we can derive the fixed point equation for $\hat{q}$ by computing the derivative of Ψ_{out} w.r.t q.

$$\begin{aligned}
\partial_q \Psi_{\text{out}} =& \partial_q \mathbb{E}_{y,\xi}\left[\mathcal{Z}_{\text{out}\,\star}\left(y, mq^{-1/2}\xi, \rho_{\text{w}^\star} - mq^{-1}m\right)\log \mathcal{Z}_{\text{out}}\ \left(y, q^{1/2}\xi, Q-q, q, \varepsilon\right)\right] \\
=& \mathbb{E}_{y,\xi}\left[\partial_q \omega^\star \partial_\omega \mathcal{Z}_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\ + \partial_q V^\star \partial_V \mathcal{Z}_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\right. \\
&\left. + \frac{\mathcal{Z}_{\text{out}\,\star}}{\mathcal{Z}_{\text{out}}}\left(\partial_q \omega \partial_\omega \mathcal{Z}_{\text{out}}\ + \partial_q V \partial_V \mathcal{Z}_{\text{out}}\ + \partial_q \mathcal{Z}_{out}\right)\right] \\
=& \mathbb{E}_{y,\xi}\left[-\frac{m}{2}q^{-1/2}\xi f_{\text{out}\,\star}\mathcal{Z}_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\ + \frac{m^2q^{-2}}{2}\left(\partial_\omega f_{\text{out}\,\star} + f^2_{\text{out}\,\star}\right)\mathcal{Z}_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\right. \\
&\left. + \frac{\mathcal{Z}_{\text{out}}{}^\star}{\mathcal{Z}_{\text{out}}}\left(\frac{1}{2}q^{-1/2}\xi f_{\text{out}}\ \mathcal{Z}_{\text{out}}\ - \frac{1}{2}\left(\partial_\omega f_{\text{out}}\ + f^2_{\text{out}}\ \right)\mathcal{Z}_{\text{out}}\ + \partial_q \mathcal{Z}_{out}\right)\right] \\
=& \frac{1}{2}\mathbb{E}_{y,\xi}\left[-m^2q^{-2}\partial_\xi\left(f_{\text{out}\,\star}\mathcal{Z}_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\ \right) + m^2q^{-2}\left(\partial_\omega f_{\text{out}\,\star} + f^2_{\text{out}\,\star}\right)\mathcal{Z}_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\right. \\
&\left. + \left(\partial_\xi\left(f_{\text{out}}\ \mathcal{Z}_{\text{out}\,\star}\right) - \left(\partial_\omega f_{\text{out}}\ + f^2_{\text{out}}\ \right)\mathcal{Z}_{\text{out}\,\star}\right) + \mathcal{Z}_{\text{out}\,\star}\partial_q \mathcal{M}_\Sigma\right] \\
=& \frac{1}{2}\mathbb{E}_{y,\xi}\left[-m^2q^{-2}\left(\partial_\omega f_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\ + \mathcal{Z}_{\text{out}\,\star}f^2_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\right.\right. \\
&\left.\left. - \left(\partial_\omega f_{\text{out}\,\star} + f^2_{\text{out}\,\star}\right)\mathcal{Z}_{\text{out}\,\star}\log \mathcal{Z}_{\text{out}}\ \right)\right] + \frac{1}{2}\mathbb{E}_{y,\xi}\left[-mq^{-1}\mathcal{Z}_{\text{out}\,\star}f_{\text{out}\,\star}f_{\text{out}}\ \right] \\
& + \frac{1}{2}\mathbb{E}_{y,\xi}\left[\partial_\omega f_{\text{out}}\ \mathcal{Z}_{\text{out}}\ + mq^{-1}\mathcal{Z}_{\text{out}\,\star}f_{\text{out}\,\star}f_{\text{out}}\ - \left(\partial_\omega f_{\text{out}}\ + f^2_{\text{out}}\ \right)\mathcal{Z}_{\text{out}\,\star} + \mathcal{Z}_{\text{out}\,\star}\partial_q \mathcal{M}_\Sigma\right] \\
=& -\frac{1}{2}\mathbb{E}_{y,\xi}\left[\mathcal{Z}_{\text{out}\,\star}\left(y, mq^{-1/2}\xi, \rho_{\text{w}^\star} - mq^{-1}m\right)\left(f^2_{\text{out}}\ \left(y, q^{1/2}\xi, Q-q, q, \varepsilon\right) + \partial_q \mathcal{M}_\Sigma\right)\right]
\end{aligned} \tag{50}$$

Finally leading to

$$\hat{q} = -2\alpha \partial_q \Psi_{\text{out}}\ = \alpha \mathbb{E}_{y,\xi}\left[\mathcal{Z}_{\text{out}\,\star}\left(y, mq^{-1/2}\xi, \rho_{\text{w}^\star} - mq^{-1}m\right)\left(f_{\text{out}}\ \left(y, q^{1/2}\xi, Q-q, q, \varepsilon\right)^2 + \partial_q \mathcal{M}_\Sigma\right)\right]. \tag{51}$$

3.4 Gaussian Prior

Directly taken from Aubin... needs adaptation and some stuff obviously wrong.. Gaussian prior Let us compute the corresponding free entropy term with partition functions $\mathcal{Z}_{\text{w}^\star}$ for a Gaussian prior $P_{\text{w}^\star}(w^\star) = \mathcal{N}_{w^\star}(0, \rho_{\text{w}^\star})$ and $\mathcal{Z}_{\text{w}}^{\ell_2,\lambda}$ for a ℓ_2 regularization respectively given by eq. (41) and eq. (47):

$$\mathcal{Z}_{\text{w}^\star}(\gamma, \Lambda) = \frac{e^{\frac{\gamma^2 \rho_{\text{w}^\star}}{2(\Lambda \rho_{\text{w}^\star}+1)}}}{\sqrt{\Lambda \rho_{\text{w}^\star}+1}}, \quad \mathcal{Z}_{\text{w}}^{\ell_2,\lambda}(\gamma, \Lambda) = \frac{e^{\frac{\gamma^2}{2(\Lambda+\lambda)}}}{\sqrt{\Lambda+\lambda}}.$$

The prior free entropy term reads

$$\begin{aligned}\Psi_{\mathrm{w}}(\hat{Q}, \hat{m}, \hat{q}) &= \mathbb{E}_{\xi}\left[\mathcal{Z}_{\mathrm{w}^\star}\left(\hat{m}\hat{q}^{-1/2}\xi, \hat{m}\hat{q}^{-1}\hat{m}\right)\log\mathcal{Z}_{\mathrm{w}}^{\ell_2,\lambda}\left(\hat{q}^{1/2}\xi, \hat{q}+\hat{Q}\right)\right] \\ &= \mathbb{E}_{\xi}\left[\mathcal{Z}_{\mathrm{w}^\star}\left(\hat{m}\hat{q}^{-1/2}\xi, \hat{m}\hat{q}^{-1}\hat{m}\right)\left(\frac{\hat{q}\xi^2}{2(\lambda+\hat{Q}+\hat{q})} - \frac{1}{2}\log(\lambda+\hat{Q}+\hat{q})\right)\right] \\ &= \int \mathrm{d}\xi \mathcal{N}_{\xi}\left(0, 1+\rho_{\mathrm{w}^\star}\hat{m}^2\hat{q}^{-1}\right)\left(\frac{\hat{q}\xi^2}{2(\lambda+\hat{Q}+\hat{q})} - \frac{1}{2}\log(\lambda+\hat{Q}+\hat{q})\right) \\ &= \frac{1}{2}\left(\frac{\hat{q}+\rho_{\mathrm{w}^\star}\hat{m}^2}{\lambda+\hat{Q}+\hat{q}} - \log(\lambda+\hat{Q}+\hat{q})\right)\end{aligned}$$

In the Bayes-optimal case for $\rho_{\mathrm{w}^\star} = 1$, the computation is similar and is given by the above expression with $\lambda = 1, \hat{Q} = 0, \hat{m} = \hat{q}$:

$$\Psi_{\mathrm{w}}^{\mathrm{bayes}}(\hat{q}) == \frac{1}{2}(\hat{q} - \log(1+\hat{q}))$$

3.5 Obtaining the proximal

from aubin - Generic differentiable convex loss In general, finding the proximal map in (29) is intractable. In particular, it is the case for the logistic loss considered in Sec. V.5. However assuming the convex loss is a generic two times differentiable function $l \in \mathcal{D}^2$, taking the derivative of the proximal map

$$\mathcal{P}_V[l(y,.)](\omega) = \mathrm{argmin}_z\left[l(y,z) + \frac{1}{2V}(z-\omega)^2\right] \equiv z^\star(y,\omega,V),$$

verifies therefore the implicit equations:

$$z^\star(y,\omega,V) = \omega - V\partial_z l\left(y, z^\star(y,\omega,V)\right), \quad \partial_\omega z^\star(y,\omega,V) = \left(1 + V\partial_z^2 l\left(y, z^\star(y,\omega,V)\right)\right)^{-1}$$

Once those equations solved, the denoising function and its derivative are simply expressed as

$$f_{\mathrm{out}}^{\mathrm{diff}}(y,\omega,V) = V^{-1}\left(z^\star(y,\omega,V) - \omega\right), \quad \partial_\omega f_{\mathrm{out}}^{\mathrm{diff}}(y,\omega,V) = V^{-1}\left(\partial_\omega z^\star(y,\omega,V) - 1\right)$$

with $z^\star(y,\omega,V) = \mathcal{P}_V[l(y,)].(\omega)$ solution of (45).
Regularizations - ℓ_2 regularization Using the definition of the prior update in eq. (29) for the ℓ_2 regularization $r(w) = \frac{\lambda w^2}{2}$, we obtain

$$\begin{gathered} f_{\mathrm{w}}^{\ell_2}(\gamma,\Lambda) = \mathrm{argmin}_w\left[\frac{\lambda w^2}{2} + \frac{1}{2}\Lambda w^2 - \gamma w\right] = \frac{\gamma}{\lambda+\Lambda} \\ \partial_\gamma f_{\mathrm{w}}^{\ell_2}(\gamma,\Lambda) = \frac{1}{\lambda+\Lambda} \text{ and } \mathcal{Z}_{\mathrm{w}}^{\ell_2}(\gamma,\Lambda) = \exp\left(\frac{\gamma^2\Lambda}{2(\lambda+\Lambda)^2}\right). \end{gathered}$$

from aubin

3.6 Generalization Error

from aubin II Binary classification generalization errors In this section, we present the computation of the asymptotic generalization error

$$e_{\mathrm{g}}(\alpha) \equiv \lim_{d\to\infty} \mathbb{E}_{y,\mathbf{x}} \mathbb{1}[y \neq \hat{y}(\hat{\mathbf{w}}(\alpha);\mathbf{x})]$$

leading to expressions in Proposition. 2.1 and Thm. 2.4. The computation at finite dimension is similar if we do not consider the limit $d \to \infty$. II.1 General case The generalization error e_{g} is the prediction error of the estimator $\hat{\mathbf{w}}$ on new samples $\{\mathbf{y}, \mathrm{X}\}$, where X is an iid Gaussian matrix and $\mathbf{y}$ are ± 1 labels generated according to (18):

$$\mathbf{y} = \varphi_{\mathrm{out}\,^\star}(\mathbf{z}) \quad \text{with} \quad \mathbf{z} = \frac{1}{\sqrt{d}}\mathrm{X}\mathbf{w}^\star$$

As the model fitted by ERM may not lead to binary outputs, we may add a non-linearity $\varphi : \mathbb{R} \mapsto \{\pm 1\}$ (for example a sign) on top of it to insure to obtain binary outputs $\hat{\mathbf{y}} = \pm 1$ according to

$$\hat{\mathbf{y}} = \varphi(\hat{\mathbf{z}}) \quad \text{with} \quad \hat{\mathbf{z}} = \frac{1}{\sqrt{d}}\mathrm{X}\hat{\mathbf{w}}$$

The classification generalization error is given by the probability that the predicted labels $\hat{y}$ and the true labels y do not match. To compute it, first note that the vectors $(\mathbf{z}, \hat{\mathbf{z}})$ averaged over all possible ground truth vectors $\mathbf{w}^\star$ (or equivalently labels y) and input matrix X follow in the large size limit a joint Gaussian distribution with zero mean and covariance matrix

$$\sigma = \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star,X} \frac{1}{d} \begin{bmatrix} \mathbf{w}^{\star\top}\mathbf{w}^\star & \mathbf{w}^{\star\top}\hat{\mathbf{w}} \\ \mathbf{w}^{\star\top}\hat{\mathbf{w}} & \hat{\mathbf{w}}^\top\hat{\mathbf{w}} \end{bmatrix} \equiv \begin{bmatrix} \sigma_{\mathrm{w}^\star} & \sigma_{\mathrm{w}^\star\hat{\mathrm{w}}} \\ \sigma_{\mathrm{w}^\star\hat{\mathrm{w}}} & \sigma_{\hat{\mathrm{w}}} \end{bmatrix}.$$

The asymptotic generalization error depends only on the covariance matrix σ and as the samples are iid it reads

$$\begin{aligned} e_{\mathrm{g}}(\alpha) &= \lim_{d\to\infty} \mathbb{E}_{y,\mathbf{x}} \mathbb{1}[y \neq \hat{y}(\hat{\mathbf{w}}(\alpha);\mathbf{x})] = 1 - \mathbb{P}[y = \hat{y}(\hat{\mathbf{w}}(\alpha);\mathbf{x})] = 1 - 2\int_{(\mathbb{R}^+)^2} d\mathbf{x}\mathcal{N}_{\mathbf{x}}(\mathbf{0},\sigma) \\ &= 1 - \left(\frac{1}{2} + \frac{1}{\pi}\operatorname{atan}\left(\sqrt{\frac{\sigma^2_{\mathrm{w}^\star\hat{\mathrm{w}}}}{\sigma_{\mathrm{w}^\star}\sigma_{\hat{\mathrm{w}}} - \sigma^2_{\mathrm{w}^\star\hat{\mathrm{w}}}}}\right)\right) = \frac{1}{\pi}\operatorname{acos}\left(\frac{\sigma_{\mathrm{w}^\star\hat{\mathrm{w}}}}{\sqrt{\sigma_{\mathrm{w}^\star}\sigma_{\hat{\mathrm{w}}}}}\right) \end{aligned}$$

where we used the fact that $\operatorname{atan}(x) = \frac{\pi}{2} - \frac{1}{2}\operatorname{acos}\left(\frac{x^2-1}{1+x^2}\right)$ and $\frac{1}{2}\operatorname{acos}(2x^2-1) = \operatorname{acos}(x)$ Finally

$$e_{\mathrm{g}}(\alpha) \equiv \lim_{d\to\infty} \mathbb{E}_{y,\mathbf{x}} \mathbb{1}[y \neq \hat{y}(\hat{\mathbf{w}}(\alpha);\mathbf{x})] = \frac{1}{\pi}\operatorname{acos}\left(\frac{\sigma_{\mathrm{w}^\star\hat{\mathrm{w}}}}{\sqrt{\rho_{\mathrm{w}^\star}\sigma_{\hat{\mathrm{w}}}}}\right)$$

with

$$\sigma_{\mathbf{w}^\star\hat{\mathbf{w}}} \equiv \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star,\mathrm{X}} \frac{1}{d}\hat{\mathbf{w}}^\top\mathbf{w}^\star, \quad \rho_{\mathbf{w}^\star} \equiv \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star} \frac{1}{d}\left\|\mathbf{w}^\star\right\|_2^2, \quad \sigma_{\hat{\mathbf{w}}} \equiv \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star,\mathrm{X}} \frac{1}{d}\|\hat{\mathbf{w}}\|_2^2$$

II.3 ERM generalization error The generalization error of the ERM estimator is given again by eq. (54) with parameters

$$\sigma_{\hat{\mathbf{w}}} \equiv \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star,\mathrm{X}} \frac{1}{d} \|\hat{\mathbf{w}}\|_2^2 = \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star,\mathrm{X}} \frac{1}{d} \|\hat{\mathbf{w}}^{\mathrm{erm}}\|_2^2 \equiv q,$$
$$\sigma_{\mathbf{w}^\star\hat{\mathbf{w}}} \equiv \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star,\mathrm{X}} \frac{1}{d} \hat{\mathbf{w}}^\top \mathbf{w}^\star = \lim_{d\to\infty} \mathbb{E}_{\mathbf{w}^\star,\mathrm{X}} \frac{1}{d} \left(\hat{\mathbf{w}}^{\mathrm{erm}}\right)^\top \mathbf{w}^\star \equiv m.$$

where the parameters m, q are the asymptotic ERM overlaps solutions of eq. (11) and that finally lead to the ERM generalization error for classification:

$$e_{\mathrm{g}}^{\mathrm{erm}}(\alpha) = \frac{1}{\pi} \operatorname{acos}(\sqrt{\eta}), \qquad \text{with } \eta \equiv \frac{m^2}{\rho_{\mathrm{w}^\star} q}.$$

from aubin

4 Simulations

In this section I describe the simulations and results I obtained during the TP IV. All code is available at `https://github.com/Yezat/TPIV---Logistic-Classification`.

4.1 Calibration vs α

Let us see how the calibration behaves for various values of ε. Figure 1 shows the calibration at $p = 0.75, \tau = 0.5, \lambda = 1e-5$ and $d = 300$ for different sampling ratios. It was computed by using equation 9 and using the overlaps computed using the learned weights $\hat{\boldsymbol{w}}_{erm}$ which were obtained using gradient descent. An alternative would have been to the experimental formula given in equation 8. I decided to use the formula in equation 9 as I was able to produce smoother curves. Especially for large adversarial training strengths and large sampling ratios, the distribution of the estimator $\hat{f}_{erm}$ becomes tight and binning becomes challenging.
The calibration starts out overconfident and eventually decreases. For $\epsilon = 0$ it approaches zero in the large sampling ratio limit.

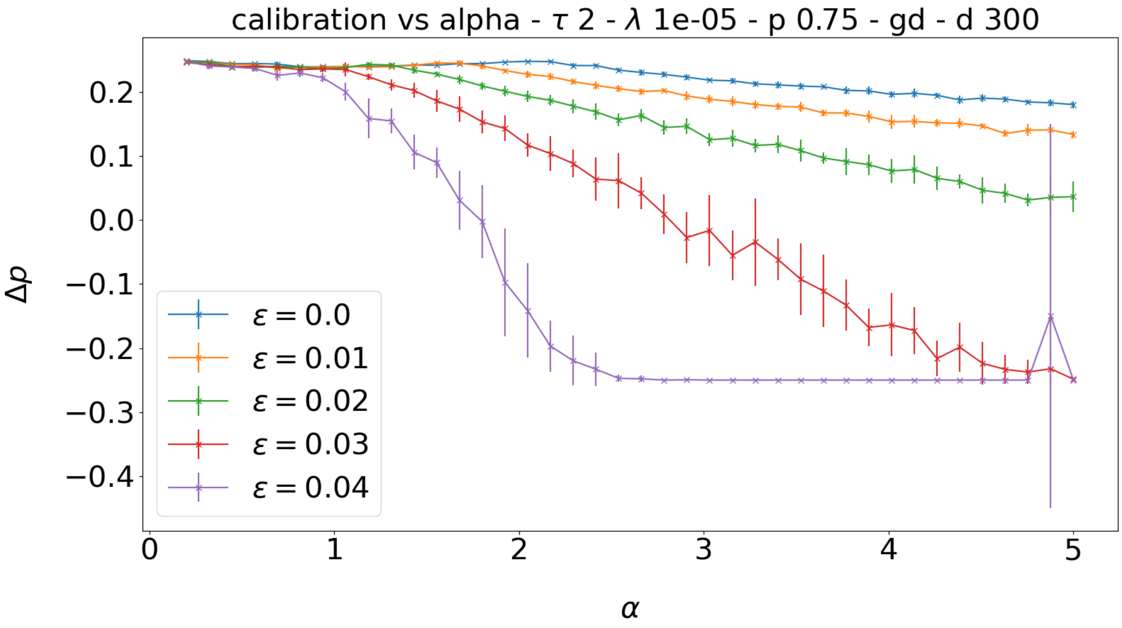

Figure 1: Δp vs the sampling ratio $\alpha = n/d$ for different adversarial training strengths ε at a noise-level $\tau = 2$ with l2-regularization $\lambda = 1e - 5$, dimension $d = 300$ and $p = 0.75$

4.1.1 Limits of $\varepsilon \to \infty$ and large data

We can understand the behaviour of the calibration for in the limit $\varepsilon \to \infty$ by noticing that the argument of the loss becomes solely dependent on $-\varepsilon \cdot ||\hat{\boldsymbol{w}}||_2$. Thus, the minimum of the risk will be at $\hat{\boldsymbol{w}}_{erm} = \mathbf{0}$. In this regime, the calibration is expected to become completely underconfident. In the case of $p = 0.75$, this means that the calibration approaches $\Delta p = 0.75 - 1 = -0.25$.
For the large data limit $n \to \infty$ or the large sampling ratio limit $\alpha \to \infty$ the calibration converges to a particular unknown value. With more theory, we could predict the exact value the calibration converges to. Figure 2 shows the logarithm of the calibration against large sampling ratios for $\varepsilon \leq 0.02$. Larger ε are not interesting as they will lead to complete underconfidence, i.e. the calibration will be $\Delta p = p - 1$. The calibration for $\varepsilon = 0.01$ or $\varepsilon = 0.02$ seem to be converging to a value above $p - 1$ as they decay like the same power-law as the calibration for $\varepsilon = 0$, which converges to 0.
Note that the scale of the adversarial training strength ε may seem small. This is due to the scales of $\boldsymbol{w}$ and $\boldsymbol{x}$. $\boldsymbol{w}$ is sampled with a variance of order 1 whilst $\boldsymbol{x}$ is sampled with a variance of order $\frac{1}{d}$. Hence, small adversarial training strength ε are sufficient to change the loss significantly.

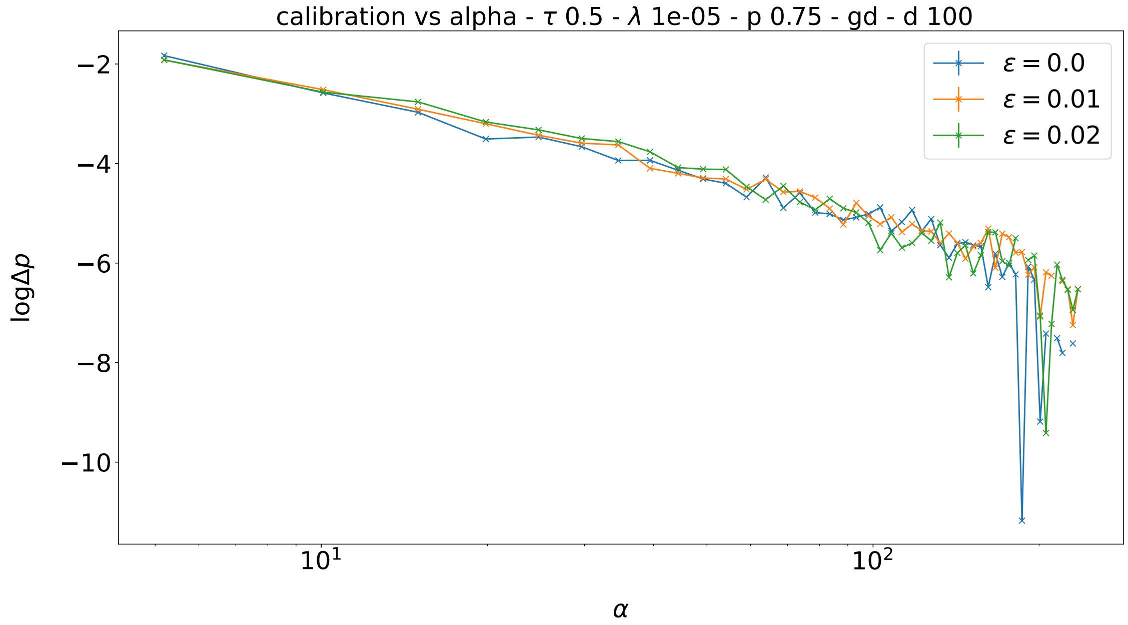

Figure 2: The logarithm of the calibration Δp vs the sampling ratio $\alpha = n/d$ for different adversarial training strengths ε at a noise-level $\tau = 2$ with l2-regularization $\lambda = 1e - 5$, dimension $d = 300$ and $p = 0.75$. We can conclude that the calibration decays like a power-law with increasing sampling ratio. Hence, it converges to a particular unknown value.

4.1.2 Correctness

By looking at the training error 3, we can see the described separability thresholds again. For α below the threshold, the ERM problem is unbounded and gradient descent can stop at zero training error. Above the separability-threshold the data are not linearly separable and hence training error will never be zero. As predicted by the conjecture in (Taheri et al., 2021), for $\varepsilon > 0$, $\alpha_{c,\varepsilon} < \alpha_{c,\varepsilon=0}$. Figure 4 shows the generalization error. It behaves similarly for any ε and it shows that the estimator gets better if there is more data.
Looking at these two metrics, we can argue that the gradient descent works correctly and we can use it for further studies.

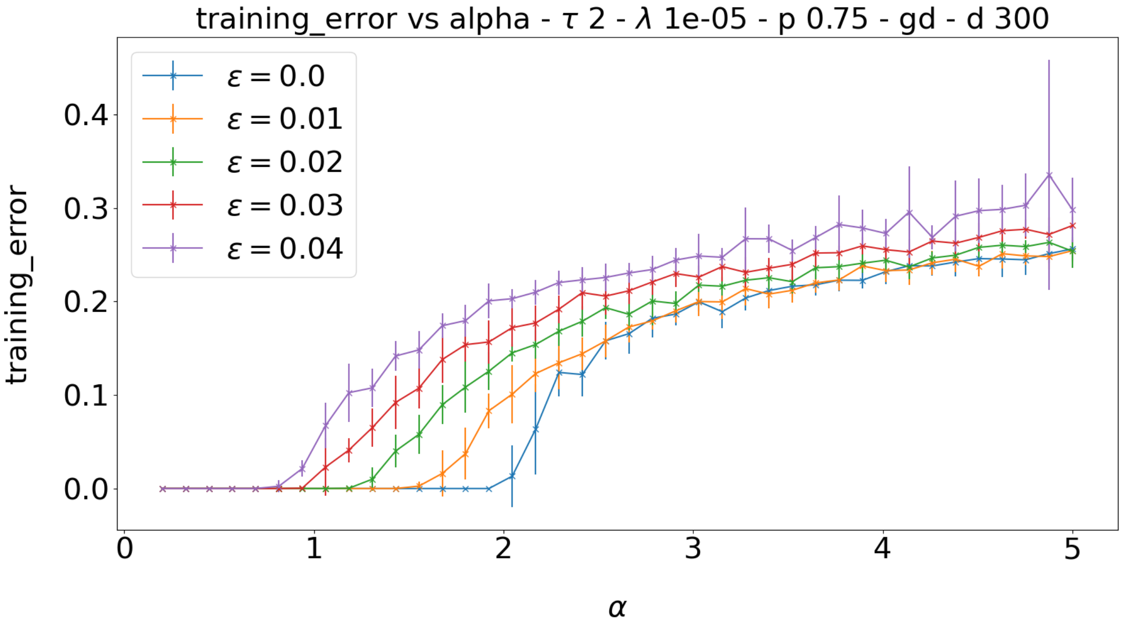

Figure 3: Training error plotted against the sampling ratio α for various values of the adversarial training strength ε at a noise-level $\tau = 2$, l2-regularization $\lambda = 1e{-}5$, dimension $d = 300$ and $p = 0.75$

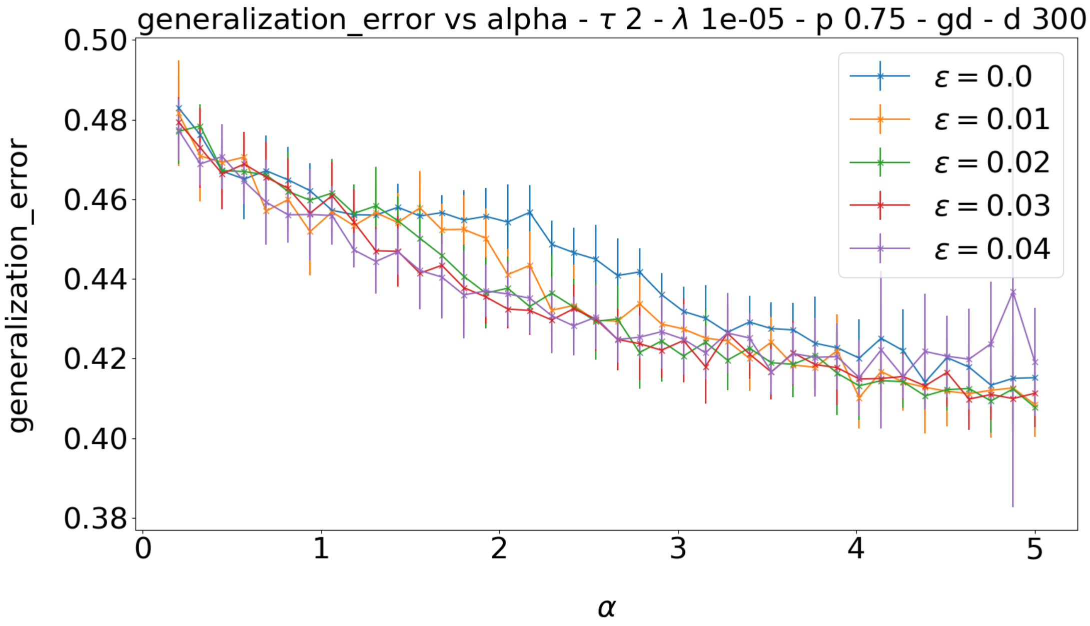

Figure 4: Generalization error plotted against the sampling ratio α for various values of the adversarial training strength ε at a noise-level $\tau = 2$, l2-regularization $\lambda = 1e - 5$, dimension $d = 300$ and $p = 0.75$

4.2 Optimal choice of λ

In particular, we would like to figure out if adding an adversarial term during training acts like adding a higher regularization term. This question is of practical interest as many real world machine learning applications require hyper-parameter tuning including the choice of optimal regularization or adversarial parameters. In practice the strength of regularization is chosen by cross-validating and choosing the λ that minimizes validation error. Ideally we would get a λ that minimizes validation error and gives a well-calibrated estimator. Let's call λ_{loss} the regularization strength that minimizes test loss.

Figure 5 shows the test loss plotted versus λ and figure 8 shows the calibration plotted against different λ. For $\lambda \to \infty$, the estimator is underconfident for $p \geq 0.5$ and as $\lambda \to 0$ the estimator becomes overconfident. (Clarté et al., 2022) also show that optimal λ with respect to calibration is only mildly dependent on p. They also show that choosing λ optimally with respect to test loss rather than test error gives a better calibrated estimator.

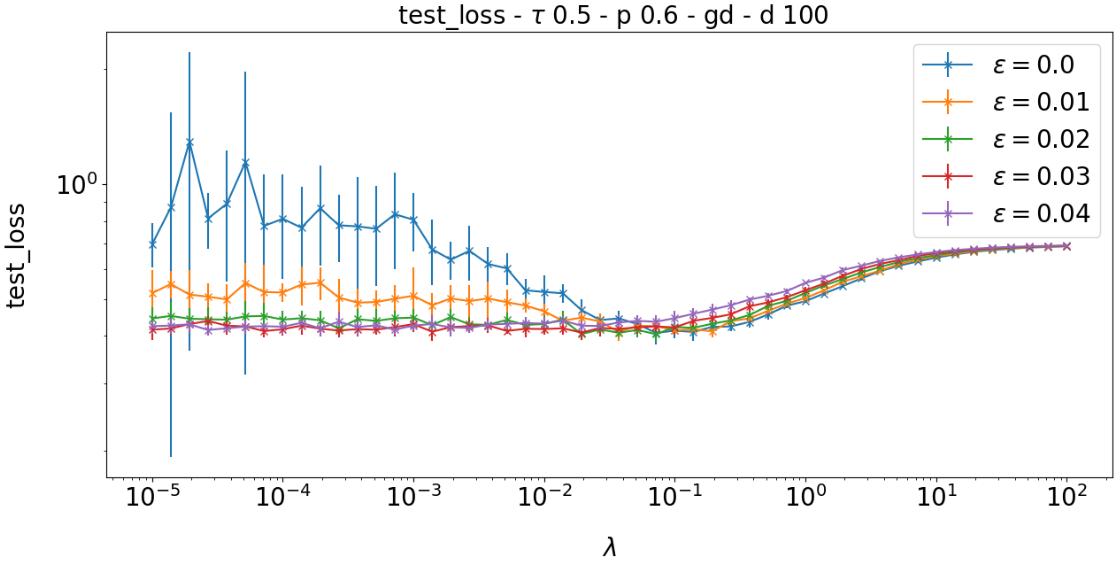

Figure 5: Test loss as a function of l2-regularization strength λ for different adversarial training strength ε at a sampling ratio $\alpha = 5$ with noise-level $\tau = 0.5$, $p = 0.6$ and dimension 100

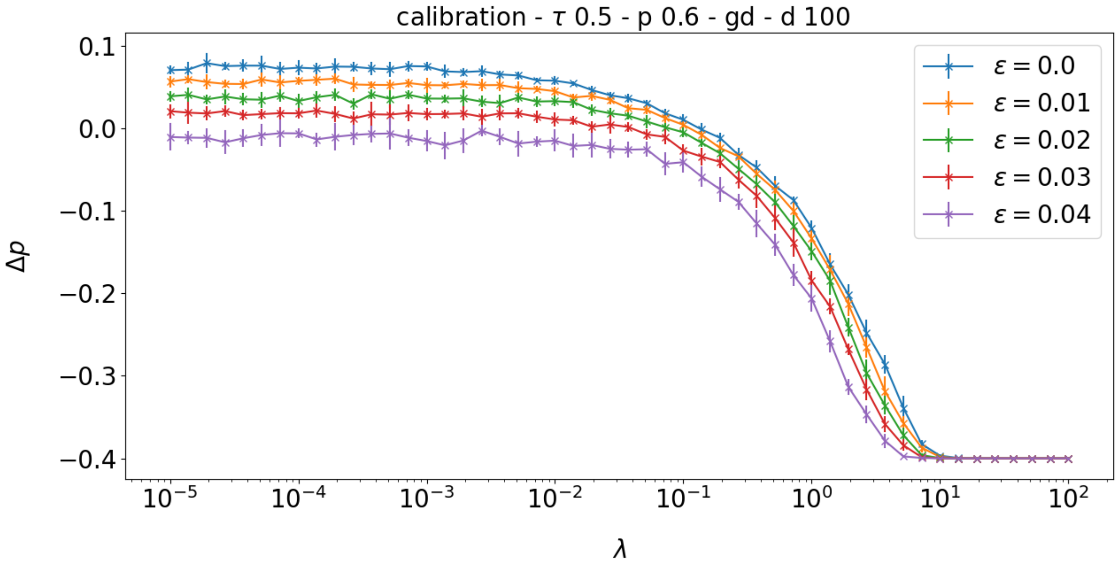

Figure 6: Calibration Δp plotted against l2-regularization strength λ at a sampling ratio of $\alpha = 5$, dimension $d = 100$, $p = 0.6$ and noise-level $\tau = 0.5$

We can now proceed computing the optimal λ at different values for ε. To do this, we use scipy and its minimize function. We minimize the loss of our estimator over λ. This is a compute-intensive operation and errors propagate from the computation of the estimator, to the computation of the loss and finally to the resulting optimal value of λ. Hence, the standard deviation of the optimal choice of λ, shown in figure 7, is large. Note that 7 only shows optimal λ for $\varepsilon \leq 0.02$. This is due to the behaviour of the test loss at higher ε. As can be seen in figure 5 it is difficult to make out a minimum of the test loss for $\varepsilon \geq 0.02$. The minimum may even be at $\lambda = 0$ for high ϵ. This suggests that adding an adversarial term acts similarly to regularizing. If the adversarial term is high, there is no gain in regularizing. Nevertheless, it is not a good idea to choose a high adversarial term as it leads to an underconfident estimator where the weights are approaching zero. If the adversarial term is not that high, the optimal choice of λ decreases with increasing ε.

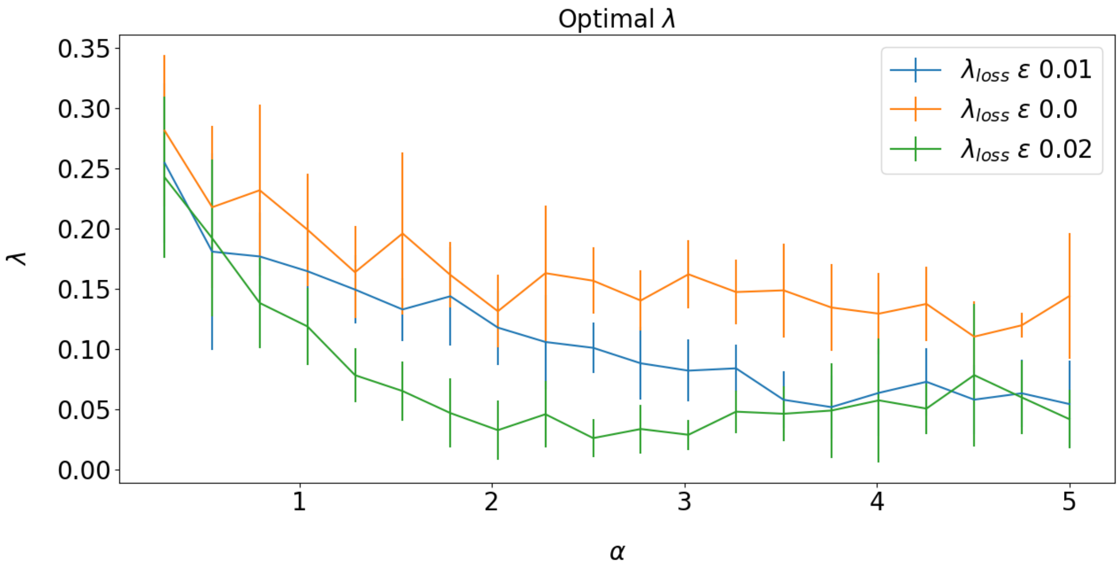

Figure 7: The optimal l2-regularization strength λ plotted against the sampling ratio α at different adversarial training strengths ε. The parameters are $d = 1000$ and $\tau = 0.5$

4.3 Calibration at optimal l2-regularization strength λ

Now that we have computed the optimal values of λ, we can investigate the calibration, generalization error and loss at the sampling ratios where we computed the respective optimal λ. Figure 8 shows the calibration evaluated at values for α for which we computed the optimal λ. For $\varepsilon \leq 0.01$, the calibration is almost optimal for every optimal choice of λ. As ε increases, the estimator becomes underconfident with increasing sampling ratio.

In figure 10 and figure 9, we can see the test loss and the generalization error in the same setting as described above. Again there is almost no difference between $\varepsilon = 0$ and $\varepsilon = 0.01$. For higher ϵ both test loss and generalization error are higher than with lower adversarial terms. Note that the calibration for larger ε is to be taken with a grain of salt. As explained before, for higher ε the optimization procedure of finding the optimal λ as implemented during this project is not reliable. Hence the optimal values of λ might be off.

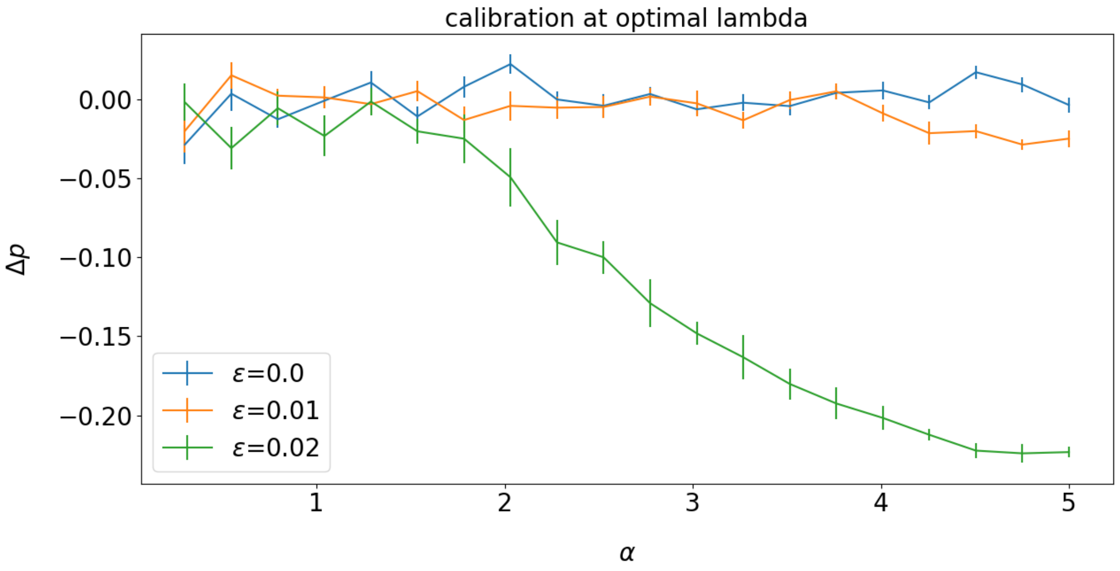

Figure 8: The calibration Δp at optimal l2-regularization strength λ at different adversarial training strengths ε. Every point is computed at it's respective optimal value of λ. Here the dimension is $d = 1000$, $p = 0.75$ and the noise-level $\tau = 0.5$

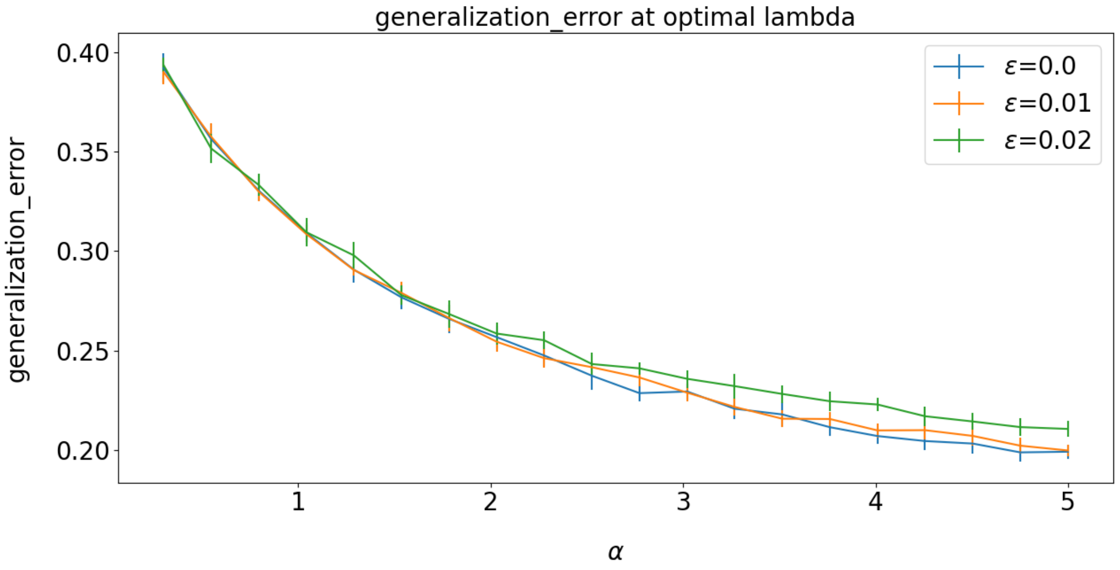

Figure 9: Generalization error at optimal l2-regularization strength λ at different adversarial training strengths ε. Every point is computed at it's respective optimal value of λ. Here the dimension is $d = 1000$ and the noise-level $\tau = 0.5$

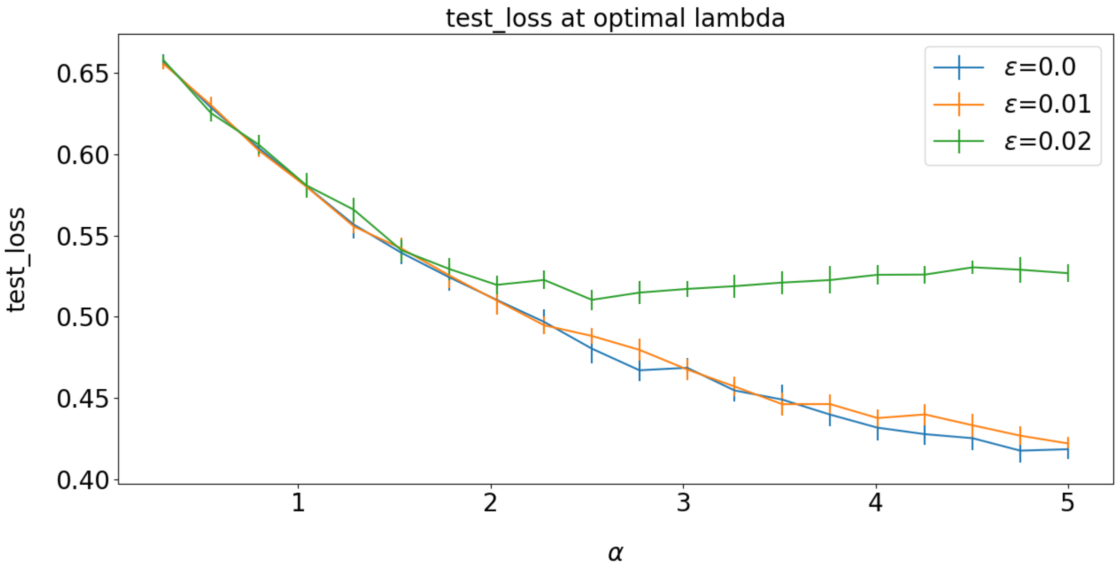

Figure 10: Test loss at optimal l2-regularization strengths λ at different adversarial training strengths ε. Every point is computed at it's respective optimal value of λ. Here the dimension is $d = 1000$ and the noise-level $\tau = 0.5$

5 Discussion

To wrap up, we have seen the separability thresholds to decrease with increasing adversarial terms ε as predicted by (Taheri et al., 2021). We have computed optimal values for λ at different ε and observed that slight adversarial attack does not change the calibration much at optimal λ and it is still well-calibrated. For higher ε we started to see numerical imprecision in the computation of the optimal λ. Nevertheless, from the results we got it looks like adding an adversarial term behaves like increasing regularization. Hence, for this logistic regression on probit data with gaussian inputs, there seems to be no benefit in training against an adversary. This may change in a settings with more complicated data and noise models. The kind of adversarial training might also impact its effectiveness. We could retry the experiments for instance in training against l1-norm attacks.

Ideally we would optimize over the l2-regularization and the adversarial term and then compare the the calibration. We can increase confidence in our results by investing more time in optimizing the minimization over the l2 regularization.

An alternative to computing the optimal λ using scipy minimize, would be to curve-fit the loss as a function of λ and to find the minimum and do the same optimization for the adversarial training strength. In the next TP IV, we could derive exact asymptotic formulas for the calibration and errors, akin to the analysis in (Taheri et al., 2021). This would allow us to consolidate our preliminary results and to further investigate the interplay between adversarial training and calibration when both regularization and adversarial attack are optimally chosen.

References

Aubin, B., Krzakala, F., Lu, Y. M., & Zdeborová, L. (2020, November). *Generalization error in high-dimensional perceptrons: Approaching Bayes error with convex optimization.* arXiv. Retrieved 2023-04-27, from `http://arxiv.org/abs/2006.06560` (arXiv:2006.06560 [cond-mat, stat]) doi: 10.48550/arXiv.2006.06560 9

Clarté, L., Loureiro, B., Krzakala, F., & Zdeborová, L. (2022). Theoretical characterization of uncertainty in high-dimensional linear classification. *CoRR, abs/2202.03295*. Retrieved from `https://arxiv.org/abs/2202.03295` 2, 3, 5, 21

Hastie, T., Tibshirani, R., & Friedman, J. (2009). Linear methods for classification. In *The elements of statistical learning: Data mining, inference, and prediction* (pp. 101–137). New York, NY: Springer New York. Retrieved from `https://doi.org/10.1007/978-0-387-84858-7_4` doi: 10.1007/978-0-387-84858-7_4 3

Taheri, H., Pedarsani, R., & Thrampoulidis, C. (2021). *Asymptotic behavior of adversarial training in binary classification.* 3, 5, 19, 25