ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute for
Dynamic Systems and Control
IDSC
Institut für Dynamische Systeme
und Regelungstechnik

Victor Jimenez Rodriguez

Improved robustness of deep learning models through posterior agreement-based model selection

Master Thesis

Institute for Machine Learning
Swiss Federal Institute of Technology (ETH) Zurich

Supervision

Dr. Joao Borges de Sa Carvalho, Dr. Alessandro Torcinovich
Prof. Dr. Joachim M. Buhmann

September 2024

IDSC-XX-YY-ZZ

Preface

The work presented in this thesis was performed at the Information and Science Engineering Group at Institute for Machine Learning (ETH Zurich), during the period October 2023 - September 2024, under the supervision of Prof. Dr. Joachim M. Buhmann. The thesis was co-supervised at Universitat Politecnica de Catalunya by Prof. Dr. Alexandre Parera i Lluna.

Contents

Abstract **v**

Notation **vii**

1 Introduction **1**
1.1 Motivation and objectives . . . 1
1.2 Related work . . . 1

2 Theoretical background **3**
2.1 The learning framework . . . 3
2.1.1 Empirical risk minimization . . . 3
2.1.2 Regularized risk minimization . . . 4
2.2 Learning with neural networks . . . 4
2.2.1 Backpropagation and gradient descent . . . 5
2.2.2 Loss landscape and parameter space . . . 5
2.3 Posterior agreement . . . 6
2.3.1 Posterior distribution . . . 6
2.3.2 Generalization error . . . 7
2.3.3 Maximum posterior agreement . . . 9

3 Experimental setup **11**
3.1 Problem formulation . . . 11
3.1.1 The classification problem . . . 11
3.1.2 Covariate shift robustness in classification tasks . . . 12
3.1.3 Adversarial setting . . . 14
3.1.4 Out-of-distribution setting . . . 14
3.2 Posterior agreement as a measure of robustness . . . 14
3.2.1 Posterior in classification tasks . . . 14
3.2.2 The posterior agreement kernel . . . 15
3.2.3 Analytical example . . . 17
3.3 Posterior agreement beyond robustness . . . 17

4 Results and discussion **19**

5 Discussion **21**

A Supplementary material **23**
A.1 Proof of problem formulation . . . 23
A.2 Properties of the PA kernel . . . 25

B Again Something **29**

Abstract

Posterior Agreement (PA) has been proposed as a theoretically-grounded alternative for model robustness assessment in covariate shift settings. In this work, we provide further evidence in favor of this hypothesis and we explore the use of PA as a model selection criterion for deep learning models in supervised classification tasks.

Starting from the theoretical principles leading to PA, we derive a computationally-efficient approximation to its value in discrete hypothesis set problems, and we show that it is a valid alternative to cross-validation in the presence of distribution shifts. Additionally, we follow some threads from the original hypothesis and we extend the use of PA to some other specific settings in which a domain shift can be defined, such as subpopulation (unbalanced) settings, mislabelled datasets and data augmentation strategies.

Notation

Symbols

EHC	Conditional equation	[−]
e	Willans coefficient	[−]
F, G	Parts of the system equation	[K/s]

Indicies

a	Ambient
air	Air

Acronyms and Abbreviations

NEDC	New European Driving Cycle
ETH	Eidgenössische Technische Hochschule

Chapter 1

Introduction

This chapter aims to set the stage for the detailed analysis and discussion that will follow, by providing a general overview of the problem of model robustness in machine learning and the current approaches to address it.

1.1 Motivation and objectives

- Introduce deep learning, the current development and the implications to society.
- Explain what is robustness, give examples and why is it critical for the development and implementation of DL. Give the example of cow/human image classification (funny, switzerland)
- Besides the funny example, provide extensive argumentation to highlight why is solving it is of pararmount importance
- Outline (no maths) why is it a difficult problem.

- Explain current approaches to robustness and ML development in general => trash accuracy.
- Introduction to posterior agreement as a robustness measure. Cite Joao+Alessandro paper, in which the conditions for a robustness metric are outlined.
- Is it necessary to go into the record of PA in other settings ?
- Show initial results from the paper, and explain why it is a promising approach.

- Lead to the derivative work that will be presented in the thesis.=> benchmarking
- Lead to derivative but next level (more useful, current interest...) work => model selection
- Lead to non-derivative (i.e. probably unsuccessful) work => model selection beyond robustness
- Outline the structure of the thesis in terms of hypothesis.

1.2 Related work

- Adversarial learning
- Domain adaptation
- Model selection for robustness

Chapter 2

Theoretical background

2.1 The learning framework

Statistical learning theory encompasses the mathematical framework used to study generalization in machine learning. In this formalism, the goal is to learn a target function $f^* : \mathcal{X} \to \mathcal{Y}$ by means of an approximated function $f \in \mathcal{F}$ using a finite set of observations.

Definition (Supervised dataset)**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be the input and output spaces of the target function, respectively. Let X be a random variable associated with the sampling in $\mathcal{X}$, and $\underline{X} = (X_1, ..., X_N) \overset{iid}{\sim} X$ be a N-sized simple random sample of X. A supervised dataset D is a realization of $\boldsymbol{x} \sim \underline{X}$ paired with its f^*-mapped output values.*

$$D = \{(x_n, f^*(x_n))\}_{n\in[N]} = \{(x_n, y_n)\}_{n\in[N]}$$

To avoid notation clutter, we will define $\mathcal{D}$ as the class of supervised datasets in $\mathcal{X}$, and slightly abuse the notation by considering $\boldsymbol{x} \in \mathcal{D}$, instead of $(\boldsymbol{x}, \boldsymbol{y}) \equiv D \in \mathcal{D}$.

2.1.1 Empirical risk minimization

The quality of the approximation can be measured with the expected risk $\mathcal{R}(f)$

$$\mathcal{R}(f) = \mathbb{E}_X[\mathcal{L}(f(x), f^\star(x))]$$

where $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ denotes a loss function. Glivenko-Cantelli theorem allow us to estimate the expected risk with its empirical (plug-in) analogous when N is large enough.

Definition (Empirical risk)**.** *Let D and $\mathcal{L}$ be the dataset and loss function of our problem, respectively. The empirical risk of $f \in \mathcal{F}$ computed on D is defined as*

$$\hat{\mathcal{R}}_D(f) = \frac{1}{N}\sum_{n=1}^{N} \mathcal{L}(f(x_n), f^\star(x_n))$$

Training, therefore, amounts to minimizing the empirical risk over the function class $\mathcal{F}$.

$$\text{ERM}_D = \hat{f}_D = \min_{f\in\mathcal{F}} \hat{\mathcal{R}}_D(f)$$

The best learning algorithm will be that minimizing the difference between the expected and empirical risks computed on a different realization of the data.

Definition (Empirical generalization error)**.** *Let D_t and D_v be two supervised datasets. Let $\hat{f}^* \in \mathcal{F}$ be the function minimizing the expected risk , and let $\hat{f}_{D_t}$ be the function minimizing the empirical expected risk computed on D_t. The empirical generalization error of f_{D_t} computed on D_v is defined as*

$$\mathcal{E}(D_t, D_v) = \hat{\mathcal{R}}_{D_v}(\hat{f}_{D_t}) = [\hat{\mathcal{R}}_{D_v}(\hat{f}) - \hat{\mathcal{R}}_{D_v}(\hat{f}^*)] + \hat{\mathcal{R}}_{D_v}(\hat{f}^*) = \mathcal{E}_{estimation} + \mathcal{E}_{capacity}$$

where $\mathcal{E}_{capacity}$ is governed by the ability of the function class $\mathcal{F}$ to represent the target function; i.e. its complexity.

The definition of complexity depends on the problem, but intuitively measures the cardinality of the subset of $\mathcal{F}$ that the algorithm is able to represent. A complex or high-capacity algorithm will be able to represent a larger subset of $\mathcal{F}$ and achieve a low empirical error, but will be also prone to overfitting to the specific learning realization thus yielding a higher generalization error.

2.1.2 Regularized risk minimization

As a general principle, the inductive bias of the algorithm (i.e. the set of constraints imposed on $\mathcal{F}$ during learning) should be aligned with that of our target function. Given that more expressive classes are always preferred by optimization algorithms, the ERM objective function is tweaqued to include a regularization term penalizing complexity.

Definition (Regularized empirical risk)**.** *Let $\Omega : \mathcal{F} \to \mathbb{R}$ be a measure of complexity. The regularized empirical risk of a function f computed on D is defined as*

$$\hat{\mathcal{R}}_\Omega(f) = \hat{\mathcal{R}}(f) + \lambda\Omega(f)$$

where $\lambda \in \mathbb{R}$ controls the trade-off between empirical risk and generalization error.

2.2 Learning with neural networks

Neural networks are biologically-inspired machine learning models that consist of a set of nodes (neurons) organized in layers and connected by weighted edges (synapses). Figure 2.1 illustrates the transformation performed within a single node.

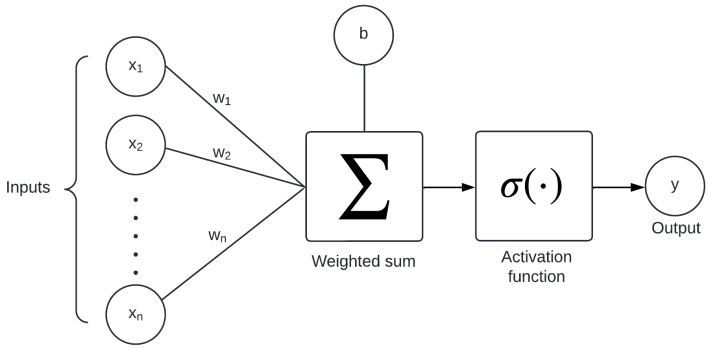

Figure 2.1: The output of a node is computed by applying a non-linear activation function $\sigma(\cdot)$ to the weighted sum of its inputs.

Let $\boldsymbol{x}_k \in \mathbb{R}^{d_k}$ be the input to the layer $k \leq L$, and let $\boldsymbol{W} \in \mathbb{R}^{d_k \times d_{k+1}}$ be the k-th weight matrix. The output of the layer can be expressed as

$$\boldsymbol{x}_{k+1} = \sigma_k(\boldsymbol{z}_{k+1}) = \sigma_k(\boldsymbol{W}_k^T \boldsymbol{x}_k + \boldsymbol{b}_k)$$

where σ_k is the non-linear activation function at layer k. We can therefore express the overall transformation of a neural network as the composition of its layers.

$$f_{\text{NN}}(\boldsymbol{x}) = \bigcirc_{k=0}^{L-1} \sigma_k(\boldsymbol{W}_k^T \boldsymbol{x} + \boldsymbol{b}_k) = f(\boldsymbol{x}; \boldsymbol{\gamma})$$

where $\boldsymbol{\gamma}$ represents the set of parameters of the network. In order to solve the learning problem, the optimization algorithm must navigate the non-convex loss landscape towards the minimum of the empirical risk. This is computationally achieved by means of gradient-descent-based optimizers, which compute the gradient over the parameters by means of the backpropagation algorithm.

2.2.1 Backpropagation and gradient descent

Let $w_{ji}^{(k)}$ be the weight from node j on layer $k-1$ to node i on layer k. Let $a_i^{(k-1)}$ be the output of node i on layer $k-1$ and let $z_j^{(k)} = \sum_{i=0}^{n_k-1} w_{ji}^{(k)} a_i^{(k-1)} + b_k^{(k)}$ be the linear input of node j on layer k, so that $a_j^{(k)} = \sigma_j(z_j^{(k)})$ is the ouput from node j. We can compute the gradient of the loss $\mathcal{L}$ with respect to the weights by means of the chain rule as follows:

$$\frac{\partial \mathcal{L}}{\partial w_{ji}^{(k)}} = \frac{\partial \mathcal{L}}{\partial a_j^{(k)}} \frac{\partial a_j^{(k)}}{\partial z_j^{(k)}} \frac{\partial z_j^{(k)}}{\partial w_{ji}^{(k)}} = \frac{\partial \mathcal{L}}{\partial a_j^{(k)}} \frac{\partial \sigma_j^{(k)}}{\partial z_j^{(k)}} a_i^{(k-1)}$$

Given that the loss is computed as a function of the output of the network, all the edges from node i of layer $k-1$ influence the loss value at that node:

$$\frac{\partial \mathcal{L}}{\partial a_i^{(k-1)}} = \sum_{j=0}^{n_k-1} \frac{\partial \mathcal{L}}{\partial a_j^{(k)}} \frac{\partial a_j^{(k)}}{\partial z_j^{(k)}} \frac{\partial z_j^{(k)}}{\partial a_i^{(k-1)}} = \sum_{j=0}^{n_k-1} \frac{\partial \mathcal{L}}{\partial a_j^{(k)}} \frac{\partial \sigma_j^{(k)}}{\partial z_j^{(k)}} w_{ji}^{(k)}$$

All in all, we see that the same terms are required in different nodes to compute the gradient, making backpropagation algorithm very efficient. Equivalently, for the bias term:

$$\frac{\partial \mathcal{L}}{\partial b_j^{(k)}} = \frac{\partial \mathcal{L}}{\partial a_j^{(k)}} \frac{\partial a_j^{(k)}}{\partial z_j^{(k)}} \frac{\partial z_j^{(k)}}{\partial b_j^{(k)}} = \frac{\partial \mathcal{L}}{\partial a_j^{(k)}} \frac{\partial \sigma_j^{(k)}}{\partial z_j^{(k)}}$$

These derivatives are the components of the gradient vector that will be used to update the weights and biases of the network.

$$w_{ji}^{(k)} = w_{ji}^{(k)} - \eta \frac{\partial \mathcal{L}}{\partial w_{ji}^{(k)}}$$

$$b_j^{(k)} = b_j^{(k)} - \eta \frac{\partial \mathcal{L}}{\partial b_j^{(k)}}$$

where η is the learning rate. More efficient variations of gradient descent such as stochastic gradient descent or Adam are used in practice.

2.2.2 Loss landscape and parameter space

A neural network architecture NN can be expressed as a parametrization of the function space $\mathcal{F}$.

$$\begin{aligned} \mathrm{NN} : \boldsymbol{\Gamma} \subseteq \mathbb{R}^S &\longmapsto \mathcal{F}_{\boldsymbol{\Gamma}} \\ \boldsymbol{\gamma} &\longmapsto f(\boldsymbol{x};\boldsymbol{\gamma}) = f_{\mathrm{NN}}(\boldsymbol{x}) \end{aligned}$$

where Γ is the parameter space associated with this particular architecture. The functional landscape $\mathcal{F}_{\boldsymbol{\Gamma}}$ consists of all mappings $f(\boldsymbol{\gamma}) : \mathcal{X} \longmapsto \mathcal{Y}$ that that can be realized by some parameter configuration $\boldsymbol{\gamma} \in \boldsymbol{\Gamma}$.

Universal approximation theorems state that arbitrarily wide or abitrarily deep architectures are able to represent virtually any function, but it is an open challenge to theoretically describe which complexity measure regulates generalization. A possible approach to this problem is to study the geometry of the loss landscape, especially in the vicinity of local minima. For instance, connected flat minima are often linked to better generalization capabilities, as they intuitively represent a

robust manifold in the parametrization space and should be preferred over sharp minima.

In this work we will explore a different approach to the generalization problem, based on a definition of the generalization error that relies on the implicit randomness of the data sampling process.

2.3 Posterior agreement

As it was mentioned in the first lines of this chapter, the input of learning algorithms are datasets containing samples of the random variable X with support $\mathcal{X}$. The implicit randomness embedded in the sampling process extends to the outcome of algorithms, even when performing a deterministic set of operations. An alternative intuition of generalization arises from this perspective, in the sense that a good algorithm should be expected to learn the same function when trained on different realizations of the same experiment; that is, when datasets are drawn from the same distribution but entail different instantiations of the noise associated with the sampling process.

A regularization principle is derived from this intuition and can be formalized as a generalization-complexity trade-off by defining generalization as the robustness or stability of the learned function to sampling noise. A suitable measure of complexity in this framework is the informativeness of the function, which represents its ability to learn the patterns in the data while filtering out the noise. The more expressive (i.e. complex) a function class is, the higher will be the estimated information content of the data. If the information content is overestimated, the approximated function will overfit to the noise and thus not generalize to different realizations of the experiment.

The robustness-informativeness regularization principle can be enforced from the set of outputs of the learned model. This section will formalize this principle and derive an operative model selection criterion.

Definition (Distribution of $\underline{X}$)**.** *The simple random sample $\underline{X} \overset{iid}{\sim} X$ has a probability distribution described by the density function $f_{\underline{X}}$.*

$$f_{\underline{X}} = \prod_{n=1}^{N} f_X(x)$$

We will use $\mathbf{P}_{\underline{X}}$ to refer to the measure of probability encoded in this distribution.

Definition (Sampling experiment)**.** *Let X be a random variable representing a sampling experiment with support $\mathcal{X}$. Let $\tau \in \mathbb{T}_{\mathcal{X}}$ be a transformation of the sampling experiment X. The set of possible transformations $\mathbb{T}_{\mathcal{X}}$ is composed of the possible experimental conditions for the data sampling process in $\mathcal{X}$. The dependency on the experimental design will be captured by the index τ, and we will implicitely consider*

$$X := \tau \circ X$$

to be a sampling experiment. $\underline{X} \sim X$ will be the simple random sample associated with specific experimental conditions τ, and sampling realizations $\boldsymbol{x}', \boldsymbol{x}'' \sim \underline{X}$ will encode different noise instantiations but equal experimental conditions.

2.3.1 Posterior distribution

Definition (Hypothesis class)**.** *Let $\mathcal{D}$ be the class of datasets generated from N-sized realizations of $\underline{X}$. A data science algorithm learns a function f implementing the following mapping:*

$$\begin{aligned} f : \mathcal{D} &\longmapsto \Theta \\ \boldsymbol{x} &\longmapsto (f(x_1), \dots, f(x_N)) = \theta \end{aligned}$$

The hypothesis class Θ is the output space of hypothesis representing all possible outcomes of a function f learned on a dataset sampled from $\underline{X}$.

Intuitively, this framework interprets complexity from the perspective of the possible set of outcomes of the function, rather than the function class itself. It can be argued that both perspectives are close, in the sense that the function class can be mapped to the hypothesis space Θ. Nevertheless, more suitable generalization regularization constraints can be defined in Θ, especially when dealing with intractable function classes $\mathcal{F}_\Gamma$ represented by deep neural networks.

For instance, complexity in the hypothesis class can be associated to the nature of the randomness displayed by X. Ideally, too restrictive hypothesis classes that lack desirable hypothesis for some realization $\boldsymbol{x} \sim \underline{\mathrm{X}}$ should be avoided, and also those hypothesis classes containing unrealizable elements (i.e. hypothesis that are not outcome of any possible experiment). A richness condition can thus be postulated following this intuition.

Definition (Richness condition)**.** *We require a sufficiently rich set of experiments $\mathbb{T}_\mathcal{X}$ such that every hypothesis $\theta \in \Theta$ is the (most likely) outcome of some realization $\boldsymbol{x} \sim \underline{X}$.*

$$\forall \theta, \exists \tau \in \mathbb{T}_\mathcal{X} \text{ such that } f(\boldsymbol{x}) = \theta$$

Since we assume a mapping f and a data distribution $\mathbf{P}_{\underline{\mathrm{X}}}$, we can describe the randomness of the hypothesis outcome conditioned on the the distribution of the data.

Definition (Posterior)**.** *Let $\mathfrak{P}^f$ be a probability distribution family under consideration. A probability distribution over the hypothesis class can be defined as a conditional distribution given an realization $\boldsymbol{x} \sim \underline{X}$. We will refer to this distribution as the posterior over Θ under f.*

$$\begin{aligned} \mathbf{P}^f : \mathcal{D} \times \Theta &\longmapsto \mathbb{R} \\ (\boldsymbol{x}, \theta) &\longmapsto \mathbf{P}^f(\theta \mid \boldsymbol{x}) \end{aligned}$$

$\mathbf{P}^f \in \mathfrak{P}^f$ establishes the stochastic relation between data realizations and hypotheses.

Using these definitions we can operate over Θ within the framework of probability theory. For instance, we can obtain the (prior) probability of a hypothesis to be selected by f as

$$\Pi^f(\theta) = \mathbb{E}_{\underline{X}} \mathbf{P}^f(\theta \mid \boldsymbol{x})$$

from which we can derive a probabilistic version of the richness condition, where a limit case can be imposed with exactly one experiment per hypothesis, leading to a uniform prior:

$$\Pi^f(\theta) = |\Theta|^{-1}$$

Within this framework, selecting suitable hypothesis classes amounts to selecting posterior distributions that yield a higher probability to the desired subset of hypothesis. This is the leading principle that will guide the derivations that follow.

2.3.2 Generalization error

In order to define a robustness-based generalization error, we will proceed in an analogous way as we did in the previuous section. We will consider the datasets D' and D'' that arise from different sampling realizations $\boldsymbol{x}', \boldsymbol{x}'' \sim \underline{\mathrm{X}}$. Both realizations are conditionally independent given the experiment, as they only differ in the measurement noise.

$$\mathbf{P}(\boldsymbol{x}', \boldsymbol{x}'') = \mathbf{P}(\boldsymbol{x}' \mid \tau)\mathbf{P}(\boldsymbol{x}'' \mid \tau)$$

Two posterior selection principles are derived from the robustness-informativeness trade-off:

P1 Posteriors should be expressive enough to cover the realizable subset of the hypothesis space.
P2 Equally likely inputs drawn from the same experiment should yield similar sets of hypothesis.

Definition (Description length)**.** *Let $\mathcal{F}_{\mathbf{\Gamma}}(\cdot)$ be the function class containing all functions represented by the parametrization $\mathbf{\Gamma}$. Let $\mathbf{P}_{\mathbf{\Gamma}}$ be the universal distribution relative to $\mathcal{F}_{\mathbf{\Gamma}}$ fulfilling the minimum description length principle. The description length of a function $f_\gamma \in \mathcal{F}_{\mathbf{\Gamma}}$ is defined as the number of bits required to encode its parameters. The code length of the argument of such distribution is*

$$DL_{f_\gamma}(\cdot) = -\log f_\gamma(\cdot)$$

The quality of the represented function f will be measured by the description length of its posterior, and thus a loss function can be defined as follows.

$$\ell(\theta, \boldsymbol{x}) = -\log \mathbf{P}^f(\theta \mid \boldsymbol{x})$$

Given that the description length encompasses also the complexity of the hypothesis class and not only the generalization capabilities, we will normalize the loss dividing by the description length of the prior.

$$-\log \Pi^f(\theta) = -\log \mathbb{E}_{\underline{X}} \mathbf{P}^f(\theta \mid \boldsymbol{x})$$

Definition (Generalization error)**.** *Let $\boldsymbol{x'}$ and $\boldsymbol{x''}$ be realizations of sample $\underline{X}$, contained in datasets D' and D'', respectively. Let Θ be the hypothesis class represented by f given $\underline{X}$. The generalization error is defined as the out-of-sample description length:*

$$\mathcal{G}_{\mathcal{X}} = \mathbb{E}_{\boldsymbol{x}', \boldsymbol{x}''} \mathbb{E}_{\theta|\boldsymbol{x}'} \left[-\log \frac{\mathbf{P}^f(\theta|\boldsymbol{x}'')}{\Pi^f(\theta)} \right]$$

It amounts to the expected loss over the normalized posteriors on the validation data $\boldsymbol{x}''$ weighted over the posterior distribution on the training data $\boldsymbol{x}'$. Intuitively, a lower generalization error is achieved when good quality hypothesis on $\boldsymbol{x}''$ are likely to be drawn from $\boldsymbol{x}'$.

Lemma 2.3.1 (Posterior agreement)**.** *The generalization error $\mathcal{G}_{\mathcal{X}}$ is non-negative and has a lower bound $-\mathcal{J}$. We define $\mathcal{J}$ as the posterior agreement.*

Proof.

$$\begin{aligned}
\mathcal{G}_{\mathcal{X}} &\geq \mathbb{E}_{\boldsymbol{x}', \boldsymbol{x}''} \left[-\log \left(\mathbb{E}_{\theta|\boldsymbol{x}'} \frac{\mathbf{P}^f(\theta \mid \boldsymbol{x}'')}{\Pi^f(\theta)} \right) \right] \\
&= \mathbb{E}_{\boldsymbol{x}', \boldsymbol{x}''} \left[-\log \left(\sum_{\theta \in \Theta} \frac{\mathbf{P}^f(\theta \mid \boldsymbol{x}') \mathbf{P}^f(\theta \mid \boldsymbol{x}'')}{\Pi^f(\theta)} \right) \right] = -\mathcal{J} \\
&\geq -\log \left(\mathbb{E}_{\boldsymbol{x}', \boldsymbol{x}''} \mathbb{E}_{\theta|\boldsymbol{x}'} \mathbb{E}_{\theta|\boldsymbol{x}''} \frac{\mathbf{P}^f(\theta \mid \boldsymbol{x}'')}{\Pi^f(\theta)} \right) = 0.
\end{aligned}$$

□

where Jensen's inequality has been applied twice to the convex function $-\log(\cdot)$.

THINGS LEFT TO MENTION: - Symmetric
- Equivalent expression on KL and I. Interpretation of terms.

ALSO LEFT TO MENTION
- From hypothesis class selection to model selection. I think this relies on the interpretation of hypothesis. Regardless of whether my interpretation is right or wrong, we can always define the mapping $\Gamma \longmapsto \Theta$ as (not homeomorphic, but something similar, check carefeully), in a way that model selection is feasible (i.e. not only algorithm selection)

2.3.3 Maximum posterior agreement

This section has outlined the foundations of a generalization-rooted model selection criterion over the hypothesis space. These final lines will formalize the maximum posterior agreement criterion, which follows from Lemma 2.3.1, as an optimization problem over the function class.

Definition (Kullback-Leibler divergence)**.** *Let* $\mathbf{P}$ *and* $\mathbf{Q}$ *be two probability distributions over the same support* Θ*. The Kullback-Leibler divergence of* $Q(\theta)$ *relative to* $P(\theta)$ *is defined as*

$$KL(\mathbf{P}(\theta) \parallel \mathbf{Q}(\theta)) = \mathbb{E}_{\mathbf{P}(\theta)}\left[\log \frac{\mathbf{P}(\theta)}{\mathbf{Q}(\theta)}\right]$$

Definition (Cross-entropy)**.** *Let* $\mathbf{P}$ *and* $\mathbf{Q}$ *be two probability distributions over the same support* Θ*. The cross-entropy of* $Q(\theta)$ *relative to* $P(\theta)$ *is defined as*

$$\mathcal{H}_{\mathbf{P},\mathbf{Q}} = -\mathbb{E}_{\mathbf{P}(\theta)} \log \mathbf{Q}(\theta)$$

Definition (Posterior agreement criterion)**.** *Let* $\boldsymbol{x'}$ *and* $\boldsymbol{x''}$ *be realizations of* $\underline{X}', \underline{X}''$*, respectively. The posterior agreement model-selection criterion is defined as follows.*

$$\begin{aligned} &\sup_{\mathcal{F}} \mathcal{J} \\ &\text{s.t. } KL(\mathbf{\Pi}^f(\theta) \parallel |\Theta|^{-1}) \leq \xi \end{aligned}$$

where $\xi \in \mathbb{R}$ *represents a small allowed deviation from uniformity in the prior.*

Theorem 2.3.2. *The posterior* $\mathbf{P}^f_*$ *maximizing the posterior agreement criterion defines a lower bound in the generalization error* $\mathcal{G}_{\mathcal{X}}$ *under the richness condition.*

$$\inf_{\mathcal{F}} \mathcal{G}_{\mathcal{X}} \geq -\sup_{\mathcal{F}} \mathcal{J}$$

Proof. We consider the lagrangian formulation of the generalization error minimization problem and apply Lemma 2.3.1.

$$\begin{aligned} &\inf_{\mathcal{F}} \left\{\mathcal{G}_{\mathcal{X}} + \alpha \mathrm{KL}(\mathbf{\Pi}^f(\theta)) \parallel |\Theta|^{-1}\right\} \\ =\ &\inf_{\mathcal{F}} \left\{\mathcal{G}_{\mathcal{X}} + \alpha \mathbb{E}_{\mathbf{\Pi}^f(\theta)} \log \mathbf{\Pi}^f(\theta) + \alpha \mathbb{E}_{\mathbf{\Pi}^f(\theta)} \log |\Theta|\right\} \\ \geq\ &\alpha \log |\Theta| + \inf_{\mathcal{F}} \left\{\alpha \mathcal{H}_{\mathbf{\Pi}^f}\right\} - \sup_{\mathcal{F}} \left\{\mathcal{J}\right\} \\ \geq\ &-\sup_{\mathcal{F}} \mathcal{J} \end{aligned}$$

The last inequality follows from the fact that the entropy does not exceed the log-cardinality of the hypothesis class.

$$\mathcal{H}_{\mathbf{\Pi}^f}(\theta) \leq \log |\Theta|, \quad \forall \mathbf{\Pi}^f$$

□

Chapter 3

Experimental setup

This chapter delineates the covariate shift setting within the supervised classification framework and introduces an operative formulation of posterior agreement. This formulation represents the cornerstone of this work as it allows for robustness-based model selection in discrete hypothesis classes.

3.1 Problem formulation

3.1.1 The classification problem

Out of all the possible learning problems in which a distribution shift can be defined, this project will focus on the supervised classification of images. The function space to navigate is composed of parametrized classifiers.

Definition (Classifier)**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be the input and output spaces of the target function, respectively. Let $K \in \mathbb{N}$ be the cardinality of $\mathcal{Y}$. A K-class classifier can be defined as the composition of three functions:*

- *A feature extractor. This function maps the input space to a d-dimensional feature space.*

$$\begin{aligned} \Phi : \mathcal{X} &\longmapsto \mathbb{R}^d \\ x &\longmapsto \Phi(x) = z \end{aligned}$$

- *A discriminant function. This function assigns a score to each of the K classes given a feature vector.*

$$\begin{aligned} \boldsymbol{F} : \mathbb{R}^d &\longmapsto \mathbb{R}^K \\ z &\longmapsto (F_1(z), \dots, F_K(z)) = \boldsymbol{F}(z) \end{aligned}$$

- *A decision rule. This function assigns the class label from a vector of scores. We will set it to be the maximum a posteriori (MAP) rule.*

$$\begin{aligned} \eta : \mathbb{R}^K &\longmapsto \mathcal{Y} = \{1, \dots, K\} \\ \boldsymbol{F}(z) &\longmapsto \hat{y} = \arg\max_j F_j(z) \end{aligned}$$

A classifier is defined as the composition of these three functions.

$$c = \eta \circ \boldsymbol{F} \circ \Phi$$

The results presented in this work are limited to neural network classifiers. These are parametrized NN architectures in $\Gamma \subseteq \mathbb{R}^{|\Gamma|}$, such that:

$$c : \mathcal{X} \times \Gamma \longmapsto \mathcal{Y} = \{1, \dots, K\}$$
$$(x, \gamma) \longmapsto c(x; \gamma) = \hat{y}$$

thus $c(x; \gamma) = \eta \circ (\boldsymbol{F} \circ \Phi)(x; \gamma)$.

The concepts defined in the previous chapter allow us to formalize the learning problem in which our robustness experiments will be conducted. We will refer to this problem as a K-class classification.

Definition (K-class classification)**.** *Let D be a supervised dataset. Let $c(\cdot; \gamma)$ be a neural network classifier, parametrized in $\Gamma \subseteq \mathbb{R}^{|\Gamma|}$. Let RRM_D be the regularized risk minimization problem for c on D. Let $\mathcal{L}$ be the cross-entropy loss function for the classifier c.*

$$\mathcal{L}(x, y) = -\log F_y(\Phi(x); \gamma)$$

The K-class classification problem is the RRM_D with loss function $\mathcal{L}$ parametrized in Γ.

$$\gamma^* = \arg\min_{\gamma \in \Gamma} -\frac{1}{N} \sum_{n=1}^{N} \log F_{y_n}(x_n; \gamma) + \lambda \Omega(\gamma)$$

No further characterization of the regularization factor will be provided in this chapter, as specific learning models and methods will be introduced together with the results.

3.1.2 Covariate shift robustness in classification tasks

The concept of robustness, as defined in the previous chapter, entails a measure of the stability of the learner to the randomness of the data sampling process, but also requires an adequate characterization of such randomness. In the context of the K-class classification problem, sampling randomness can be formalized as a shift in the distribution of the input space, also known as covariate shift.

Definition (Covariate shift)**.** *Let $\boldsymbol{x}'$ and $\boldsymbol{x}''$ be two N-sized samples of the input space $\mathcal{X}$. A covariate shift exists between $\boldsymbol{x}'$ and $\boldsymbol{x}''$ if their (Empirical) distributions are significantly different for N large enough:*

$$\mathbf{P}_{\boldsymbol{x}'} \neq \mathbf{P}_{\boldsymbol{x}''}$$

The presence of covariate shift as defined above already leads to a non-zero generalization error, given that $\boldsymbol{x}'$ and $\boldsymbol{x}''$ represent different noise instantiations and result in different learning outcomes. Nevertheless, this definition can be further expanded to encompass more practical sources of shift in the context of classification tasks.

Definition (Out-of-distribution shift)**.** *Let $X' = \tau' \circ X$ and $X'' = \tau'' \circ X$ be two sampling experiments in $\mathcal{X}$ such that $\mathbf{P}_{X'} \neq \mathbf{P}_{X''}$, as they are associated with different experimental conditions $\tau' \neq \tau''$ $\tau', \tau'' \in \mathbb{T}_{\mathcal{X}}$. In such case*

$$\boldsymbol{x}' \sim \underline{X}' \overset{iid}{\sim} X' \text{ and } \boldsymbol{x}'' \sim \underline{X}'' \overset{iid}{\sim} X''$$

leads to covariate shift known as out-of-distribution given that the major source of distribution shift are experimental conditions.

In the OOD case, $\boldsymbol{x}'$ and $\boldsymbol{x}''$ are drawn from different random variables, each with a distinct probability landscape over the support, namely source and target domains, that result in implicit differences (sometimes unbalanced) in the distribution of some features. Therefore, empirical distributions $\mathbf{P}_{\boldsymbol{x}'}$ and $\mathbf{P}_{\boldsymbol{x}''}$ will be different in general, and thus a covariate shift will be induced leading to a non-zero generalization error.

The reader should note that this definition generalizes the concept of sampling randomness as defined in the previous chapter, as it explicitely allows for X' and X'' to be different random variables. Therefore, each realization of $\underline{X}'$ and $\underline{X}''$ will not only entail a different noise instantiation but might also favour a different region of $\mathcal{X}$.

Definition (Adversarial shift)**.** *Let $\boldsymbol{x}' \sim \underline{X}$ be a sample drawn from experiment $X = \tau \circ X$. Let $\boldsymbol{\Delta}$ be a perturbation over the sample space. In this case, $\boldsymbol{x}''$ is generated by applying the perturbation to $\boldsymbol{x}'$.*

$$\boldsymbol{x}'' = \boldsymbol{x}' + \boldsymbol{\Delta}$$

which induces a covariate shift known as adversarial, given that perturbation $\boldsymbol{\Delta}$ is crafted ad-hoc to hinder the output of the model.

In adversarial examples, sampling randomness is not the source of distribution shift, as both $\boldsymbol{x}'$ and $\boldsymbol{x}''$ arise from the same realization of the experiment.

In this work, we will consider a wider concept of sampling randomness that does not only comprise the implicit noise instantiation of each realization $\boldsymbol{x} \sim \underline{\mathrm{X}}$ but also the explicit shift in the distribution of the input space generated by intentional or unintentional perturbations of the data generation process. This broader interpretation aligns practical covariate shift experiments with the robustness framework defined in the previouos chapter.

Once the possible sources of randomness in the data generation process have been established and formalized, a general concept of robustness measure must be introduced accordingly, so that the suitability of posterior agreement as a robustness metric can be assessed.

Definition (Robustness metric)**.** *Let D' and D'' be datasets generated from realizations $\boldsymbol{x}'$ and $\boldsymbol{x}''$, respectively. A robustness metric is a function Ω that quantifies the generalization capability of a learned $\hat{f}_{D'} \in \mathcal{F}$ to observations in D''.*

$$\Omega : \mathcal{D}'' \times \mathcal{F} \longmapsto \mathbb{R}$$

The baseline robustness metric in supervised classification tasks is accuracy, defined as the proportion of correct predictions achieved by a learned classifier $\hat{c}_{D'}$ over the dataset D''.

$$\mathrm{ACC}_{D'}(D'') = \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{\delta}_{y_n''} \left(\hat{c}_{D'}(x_n'') \right)$$

As it was argued in the previuous chapter, we will interpret the concept of generalization from the perspective of the possible learning outcomes of a specific experiment. The ultimate goal of robustness measurement is thus the characterization of the "resolution" limit that can be achieved in the hypothesis space consistent with the intrinsic randomness entailed by each possible realization of the experiment.

The resolution limit does not depend on the model but on the nature of the randomness of the data generation process. Therefore, a robustness metric should evaluate how stable are hypothesis to different realizations of the same experiment regardless of the complexity of the model. The more complex the model is, the higher will be the resolution of its associated hypothesis space. A regularization or model selection procedure derived from the robustness metric should then find the sweet spot between resolution and stability.

From this perspective, a suitable robustness metric should possess the following set of properties.

Properties (Robustness metric)**.** *See HERE REFERENCE.*

P1 *(Discriminable) The metric should differentiate models displaying different generalization capabilities against covariate shift.*

P2 *(Non-increasing) The metric should be non-increasing with respect to the response of the model under increasing levels of shift.*

P3 *(Task-independent) The metric should be independent of the task performance of the model.*

P1 requires that
Accuracy does not comply in general with any of these properties. For the case of parametrized classifiers, $\mathrm{ACC}(x, y) = \boldsymbol{\delta}_y(c(x; \gamma))$, thus losing the confidence information in the prediction. This may lead to lower in **P1**
DERIVE THEORETICAL EXAMPLES AND PUT THEM IN THE APPENDIX.
- Quan es perd la confianza en la prediccio, P1 es compleix nomes fins a un lower bound de shift, que arriba quan la probabilitat de la prediccio es 1/K.

- Per la mateixa rao, P2 te un altre lower bound implicit, que arriba en el moment en que dos models tenen les mateixes prediccions pero un atorga mes confiansa en la prediccio que l'altre.
- P3 es clarament violat (examples paper).

3.1.3 Adversarial setting

- Problem has been described already in the introduction, so don't do another introduction. - Formalize mathematically (Madry ...) the problem.
- Add plot from the paper x. - Talk about the accuracy-robustness trade-off.
- Talk about some techniques, such as flooding (see flooding paper). + Add justification of flooding with loss landscape and relate it to your previous section.

3.1.4 Out-of-distribution setting

- Describe the OOD setting, etc... see paper notes. For example Wasserstein seems good intro.
- Describe subpopulation shifts.
- Include somehow generalization and shortcut opportunities, relate to causal learning but just for the record

3.2 Posterior agreement as a measure of robustness

3.2.1 Posterior in classification tasks

Definition (Classification confidence)**.** *Let D be a dataset associated with a realization $\boldsymbol{x} \sim \underline{X}$. Let $F_j(\cdot; \gamma)$ be the j-th component of the score vector returned by the discriminant of the classifier. The cost function driving posterior selection will be the negative confidence in the prediction.*

$$R(\theta, \boldsymbol{x}; \gamma) = -\sum_{n}^{N} F_{\theta_i}(x_i; \gamma)$$

where θ_i is the class label associated with the i-th sample in the dataset.

The hypothesis space Θ of a K-class classification problem is the set of all possible vectors of labels associating each of the N samples to one of the K classes.

$$\Theta = \{1, \ldots, K\}^N$$

Its cardinality is thus $|\Theta| = K^N$.

Theorem 3.2.1 (Classification posterior)**.** *Let Θ be the classification hypothesis class associated with the K-class classification problem with approximating function c. The posterior distribution class $\mathfrak{P}^c$ is the Gibbs distribution family with inverse temperature parameter β.*

$$\mathbf{P}^c(\theta|\boldsymbol{x}) = \frac{\exp\left(\beta R(\theta, \boldsymbol{x}; \gamma)\right)}{\sum_{\theta\in\Theta} \exp\left(\beta R(\theta, \boldsymbol{x}; \gamma)\right)}$$

Proof. The proof is based on the maximum entropy principle (MEP), which states that given some prior testable information to be encoded by a probability distribution, the distribution that best encodes that information is the one minimizing additional assumptions besides the testable information; that is, the one maximizing information entropy within the testable space. Testable information amounts to certain constraints on the MEP optimization problem over the non-negative, Lebesgue-integrable function class $\mathcal{P}$.

$$\begin{aligned}
&\max_{\mathbf{P}^c(\theta|\boldsymbol{x})\in\mathcal{P}} \mathcal{H}_{\mathbf{P}^c}(\theta \mid \boldsymbol{x}) \\
&\quad \text{s.t.} \sum_{\theta\in\Theta} \mathbf{P}^c(\theta \mid \boldsymbol{x}) = 1 \\
&\quad \mathbb{E}_{\mathbf{P}^c(\theta|\boldsymbol{x})}[R(\theta, \boldsymbol{x})] = \mu \;\; \forall\theta \in \Theta \\
&\quad [\mathbf{P}^c(\theta_i \mid \boldsymbol{x}) - \mathbf{P}^c(\theta_j \mid \boldsymbol{x})][R(\theta_i, \boldsymbol{x}) - R(\theta_j, \boldsymbol{x})] \geq 0 \;\; \forall\theta_i, \theta_j \in \Theta
\end{aligned}$$

where $\mu \in \mathbb{R}$ is a hyperparameter ensuring that the expected confidence is finite and the last constraint imposes a monotonic relationship between the confidence and the posterior. The lagrangian formulation of the problem with equality constraints is:

$$\mathcal{L}(\mathbf{P}^c, \alpha, \beta) = \mathcal{H}_{\mathbf{P}^c}(\theta \mid \boldsymbol{x}) + \alpha\left(1 - \sum_{\theta\in\Theta} \mathbf{P}^c(\theta \mid \boldsymbol{x})\right) + \beta\left(\mathbb{E}_{\mathbf{P}^c(\theta|\boldsymbol{x})}[R(\theta, \boldsymbol{x})] - \mu\right)$$

Its derivative with respect to $\mathbf{P}^c(\theta \mid \boldsymbol{x})$ is:

$$\frac{\partial\mathcal{L}}{\partial\mathbf{P}^c(\theta \mid \boldsymbol{x})} = -1 - \log\mathbf{P}^c(\theta \mid \boldsymbol{x}) - \alpha + \beta R(\theta, \boldsymbol{x})$$

which has as solution:

$$\mathbf{P}^c(\theta \mid \boldsymbol{x}) = \frac{\exp\left(\beta R(\theta, \boldsymbol{x})\right)}{\exp\left(1 + \alpha\right)}$$

setting $\exp(1+\alpha) = \sum_{\theta\in\Theta} \exp\left(\beta R(\theta, \boldsymbol{x})\right)$ and $\beta \geq 0$ ensures normalization and fulfills the monotonic relationship constraint. □

The regularization and PA formulations introduced in this work are applicable regardless of the nature of the shift. Nevertheless, a proper formulation of each case is required for reference.

3.2.2 The posterior agreement kernel

Lemma 3.2.2 (Exchangeability)**.** *Let $N, K \in \mathbb{N}$ and let $\{\mathcal{E}_{ij} \mid i \leq N, j \leq K\}$ be an indexed set of values. Then,*

$$\sum_{c\in\mathcal{C}} \prod_{i=1}^{N} \mathcal{E}_{i,c(i)} = \prod_{i=1}^{N} \sum_{j=1}^{K} \mathcal{E}_{ij}$$

Proof. See Appendix A.1. □

Theorem 3.2.3 (Posterior factorization)**.** *The posterior distribution for a classification problem can be factorized as follows:*

$$\mathbf{P}^c(\theta \mid \boldsymbol{x}) = \prod_{i}^{N} \mathbf{P}_i^c(\theta_i \mid \boldsymbol{x}) = \prod_{i=1}^{N} \frac{\exp\left(\beta F_{\theta_i}(x_i)\right)}{\sum_{j=1}^{K} \exp\left(\beta F_j(x_i)\right)}$$

Proof. See Appendix A.1. □

Theorem 3.2.4 (PA kernel for classification)**.** *Let $\boldsymbol{x}'$ and $\boldsymbol{x}''$ be N-sized realizations of $\underline{X}', \underline{X}''$, respectively. Let Θ be the hypothesis class represented by c given support $\mathcal{X}$. With no prior information about Θ, the posterior agreement kernel for supervised K-class classification tasks has the following expression.*

$$\mathrm{PA}\left(\boldsymbol{x}', \boldsymbol{x}''; \beta\right) = \frac{1}{N} \sum_{i=1}^{N} \log \left\{ |\Theta| \sum_{j=1}^{K} \mathbf{P}_i^c \left(j \mid \boldsymbol{x}'\right) \mathbf{P}_i^c \left(j \mid \boldsymbol{x}''\right) \right\}$$

where $\mathbf{P}_i^c(j \mid \boldsymbol{x})$ can be shown to be:

$$\mathbf{P}_i^c \left(j \mid \boldsymbol{x}\right) = \frac{\exp\left(\beta F_j(x_i)\right)}{\sum_{k=1}^{K} \exp\left(\beta F_k(x_i)\right)}$$

Proof. The posterior agreement $\mathcal{J}$ has the following expression, derived in Lemma 2.3.1:

$$\mathcal{J} = \mathbb{E}_{X', X''} \left[\log \left(\mathbb{E}_{\mathbf{P}^c(\theta \mid \boldsymbol{x}')} \frac{\mathbf{P}^c(\theta \mid \boldsymbol{x}'')}{\mathbf{\Pi}^c(\theta)} \right) \right]$$

As defined previously, Θ is a discrete, finite set of possible classification vectors of the N observations, and the sampling distribution $\mathbf{P}_X$ is assumed to be uniform. Therefore, the expectation operators amount to:

$$\mathbb{E}_{X', X''} = \frac{1}{N} \sum_{i=1}^{N} \cdot$$

$$\mathbb{E}_{\mathbf{P}^c(\theta \mid \boldsymbol{x}')} = \sum_{\theta \in \Theta} \mathbf{P}^c(\theta \mid \boldsymbol{x}') \cdot$$

A non-informative prior is assumed, thus enforcing the richness condition:

$$\mathbf{\Pi}^c(\theta) = |\Theta|^{-1}$$

$\mathbf{P}^c(\theta | \boldsymbol{x})$ can be factorized on the terms expressed in Theorem 3.2.3.

$$\mathbf{P}^c(\theta \mid \boldsymbol{x}) = \prod_{i}^{N} \mathbf{P}_i^c(\theta_i \mid \boldsymbol{x}) = \prod_{i=1}^{N} \frac{\exp\left(\beta F_{\theta_i}(x_i)\right)}{\sum_{k=1}^{K} \exp\left(\beta F_k(x_i)\right)}$$

Operating analogously for $\boldsymbol{x}'$ and $\boldsymbol{x}''$, the expression for the PA kernel is obtained.

$$\begin{aligned}
\mathrm{PA}\left(\boldsymbol{x}', \boldsymbol{x}''; \beta\right) =& \frac{1}{N} \sum_{i=1}^{N} \left[\log \left(\sum_{\theta \in \Theta} \mathbf{P}^c(\theta \mid \boldsymbol{x}') \frac{\mathbf{P}^c(\theta \mid \boldsymbol{x}'')}{|\Theta|^{-1}} \right) \right] \\
=& \frac{1}{N} \sum_{i=1}^{N} \left[\log \left(|\Theta| \sum_{\theta \in \Theta} \mathbf{P}^c(\theta \mid \boldsymbol{x}') \mathbf{P}^c(\theta \mid \boldsymbol{x}'') \right) \right] \\
=& \frac{1}{N} \sum_{i=1}^{N} \left[\log \left(|\Theta| \sum_{\theta \in \Theta} \prod_{i=1}^{N} \frac{\exp\left(\beta F_{\theta_i}(x_i')\right)}{\sum_{k=1}^{K} \exp\left(\beta F_k(x_i')\right)} \frac{\exp\left(\beta F_{\theta_i}(x_i'')\right)}{\sum_{k=1}^{K} \exp\left(\beta F_k(x_i'')\right)} \right) \right]
\end{aligned}$$

Finally, applying Lemma 3.2.2 to the product inside the logarithm, we reach the final expression. □

Theorem 3.2.5 (Properties of the PA kernel)**.** $\mathrm{PA}\left(\boldsymbol{x}', \boldsymbol{x}''; \beta\right)$ *has the following properties.*

P1 *(Non-negativity)* $\mathrm{PA}\left(\boldsymbol{x}', \boldsymbol{x}''; \beta\right) \geq 0 \;\; \forall \boldsymbol{x} \sim \underline{X}$ *and* $\beta \in \mathbb{R}^+$.

P2 *(Symmetry)* $\mathrm{PA}(\boldsymbol{x}', \boldsymbol{x}''; \beta) = \mathrm{PA}(\boldsymbol{x}'', \boldsymbol{x}'; \beta)$. *This property is important from the robustness perspective, given that noise instantiations are not indexed and no reference noiseless experiment can be performed.*

P3 *(Concavity)* $\mathrm{PA}(\boldsymbol{x}', \boldsymbol{x}''; \beta)$ *is a concave function of* $\beta \in \mathbb{R}^+$ $\forall \boldsymbol{x} \sim \underline{X}$. *This means that the kernel optimization problem will have a unique solution.*

Proof. See Appendix A.2. □

3.2.3 Analytical example

- Include the analytical example derivation, adapt notation. Probably also leave some things in the appendix. For example, in the appendix you can leave the means over the normal distribution.

3.3 Posterior agreement beyond robustness

- Alternative formulation
- OJO: For cross-validation, I can consider the final feature vector associated with the image (i.e. before the classification layer) to be the measurement. Models trained with different subsets of data will have a different noise instantiation of the same measurement. therefore the alternative formulation is not necessary for cross-validation.
- Explain why important, and formalize the data augmentation strategy (presentation)

Chapter 4

Results and discussion

Blah, blah ...

Chapter 5

Discussion

Blah, blah ...

Appendix A

Supplementary material

We will define some notation shortcuts for the following proofs.

A.1 Proof of problem formulation

Lemma A.1.1. *Let $N, K \in \mathbb{N}$ and let $\{\mathcal{E}_{ij} \mid i \leq N, j \leq K\}$ be an indexed set of values. Then,*

$$\sum_{c \in \mathcal{C}} \prod_{i=1}^{N} \mathcal{E}_{i,c(i)} = \prod_{i=1}^{N} \sum_{j=1}^{K} \mathcal{E}_{ij}$$

Proof. By induction on N. For the $N = 1$ base case, observe that $\mathcal{C}$ has only K elements, as there are only K functions mapping $\{1\}$ to $\{1, \ldots, K\}$. Then

$$\sum_{c \in \mathcal{C}} \prod_{i \leq N} \mathcal{E}_{i,c(i)} = \sum_{c \in \mathcal{C}} \mathcal{E}_{1,c(1)} = \sum_{j \leq K} \mathcal{E}_{1,j} = \prod_{i \leq N} \sum_{j \leq K} \mathcal{E}_{i,j}.$$

Assume now that the result holds for some N. We demonstrate then that it also holds for $N + 1$. Observe that there is a bijection between $\mathcal{C}$ and $\{1, \ldots, K\}^N$. Therefore, we identify every function $c \in \mathcal{C}$ with the tuple $(c(1), \ldots, c(N))$. Conversely, we identify every tuple $(c_1, \ldots, c_N) \in \{1, \ldots, K\}^N$, with the function c that maps i to c_i.

$$
\begin{aligned}
&\sum_{c\in\mathcal{C}}\prod_{i\le N+1}\mathcal{E}_{i,c(i)} = \\
&= \sum_{(c_1,\dots,c_{N+1})\in\{1,\dots,K\}^{N+1}}\prod_{i\le N+1}\mathcal{E}_{i,c_i} \\
&= \sum_{\substack{(c_1,\dots,c_N)\in\{1,\dots,K\}^{N}\\ c_{N+1}\le K}}\prod_{i\le N+1}\mathcal{E}_{i,c_i} \\
&= \sum_{(c_1,\dots,c_N)\in\{1,\dots,K\}^{N}}\sum_{c_{N+1}\le K}\prod_{i\le N+1}\mathcal{E}_{i,c_i} \\
&= \sum_{(c_1,\dots,c_N)\in\{1,\dots,K\}^{N}}\sum_{c_{N+1}\le K}\left(\mathcal{E}_{N+1,c(N+1)}\prod_{i\le N}\mathcal{E}_{i,c_i}\right) \\
&= \left(\sum_{c_{N+1}\le K}\mathcal{E}_{N+1,c(N+1)}\right)\sum_{(c_1,\dots,c_N)\in\{1,\dots,K\}^{N}}\prod_{i\le N}\mathcal{E}_{i,c_i} \\
&= \left(\sum_{c_{N+1}\le K}\mathcal{E}_{N+1,c(N+1)}\right)\prod_{i\le N}\sum_{j\le K}\mathcal{E}_{i,j} \\
&= \left(\sum_{j\le K}\mathcal{E}_{N+1,j}\right)\prod_{i\le N}\sum_{j\le K}\mathcal{E}_{i,j} \\
&= \prod_{i\le N+1}\sum_{j\le K}\mathcal{E}_{i,j}.
\end{aligned}
$$

□

Theorem A.1.2 (Posterior factorization)**.** *The posterior distribution for a classification problem can be factorized as follows:*

$$\mathbf{P}^c(\theta \mid \boldsymbol{x}) = \prod_{i}^{N}\mathbf{P}^c(\theta_i \mid \boldsymbol{x}) = \prod_{i=1}^{N}\frac{\exp\left(\beta F_{\theta_i}(x_i)\right)}{\sum_{j=1}^{K}\exp\left(\beta F_j(x_i)\right)}$$

Proof. The posterior distribution solution to the MAP problem is the following:

$$\mathbf{P}^c(\theta \mid \boldsymbol{x})\frac{\exp\left(\beta\sum_{i=1}^{N}F_{\theta_i}(x_i)\right)}{\sum_{\theta\in\Theta}\exp\left(\beta\sum_{i=1}^{N}F_{\theta_i}(x_i)\right)} = \frac{\prod_{i=1}^{N}\exp\left(\beta F_{\theta_i}(x_i)\right)}{\sum_{\theta\in\Theta}\prod_{i=1}^{N}\exp\left(\beta F_{\theta_i}(x_i)\right)}$$

Using Lemma 3.2.2 we can rewrite the denominator as:

$$\sum_{\theta\in\Theta}\prod_{i=1}^{N}\exp\left(\beta F_{\theta_i}(x_i)\right) = \prod_{i=1}^{N}\sum_{\theta\in\Theta}\exp\left(\beta F_{\theta_i}(x_i)\right)$$

Therefore, the posterior distribution can be written as:

$$\mathbf{P}^c(\theta \mid \boldsymbol{x}) = \prod_{i=1}^{N}\mathbf{P}^c(\theta_i \mid \boldsymbol{x}) = \prod_{i=1}^{N}\frac{\exp\left(\beta F_{\theta_i}(x_i)\right)}{\sum_{\theta\in\Theta}\exp\left(\beta R(\theta,\boldsymbol{x})\right)}$$

□

A.2 Properties of the PA kernel

Theorem A.2.1 (Symmetry of the PA kernel)**.** *The posterior agreement kernel is symmetric with respect to the definition of X' and X''.*

$$PA(\boldsymbol{x}', \boldsymbol{x}'') = PA(\boldsymbol{x}'', \boldsymbol{x}')$$

Theorem A.2.2 (Non-negativity of the PA kernel)**.** *The posterior agreement kernel is non-negative.*

$$PA(\boldsymbol{x}', \boldsymbol{x}'') \geq 0$$

Theorem A.2.3 (Concavity of the PA kernel)**.** *The posterior agreement kernel is concave in $\mathbb{R}^+$, and therefore has a unique maximum.*

Proof. The posterior agreement kernel has been shown to have the following form:

$$\mathrm{PA}\left(\boldsymbol{x}', \boldsymbol{x}''\right) \propto \sum_{n=1}^{N} \log \left[\sum_{j=1}^{K} \mathbf{P}_n^c(\theta \mid x_n') \mathbf{P}_n^c(\theta \mid x_n'') \right]$$

where the posteriors $\mathbf{P}_n^c(\theta \mid x_n)$ are Gibbs distributions for each observation.

$$\mathbf{P}_n^c(\theta \mid x_n') = \frac{e^{\beta F_j(x_n)}}{\sum_{k=1}^{K} e^{\beta F_k(x_n)}}$$

We will require three important results from optimization theory:

T1 The minimum of $G(\beta) = -\mathrm{PA}(X', X'')$ over the convex set $\mathbb{R}^+$ is unique $\iff$ $G(\beta)$ is convex.

T2 G is absolutely convex $\iff$ $\frac{d^2}{d\beta^2} G(\beta) > 0$.

T3 The sum of convex functions is also convex.

To streamline the derivation, the following notation will be used:

$$F_j(x_n') = F_j'$$

$$e^{\beta F_j(x_n')} = e^{\beta F_j'} = e_j'$$

The observation index n will be omitted as it does not affect the convexity derivation (see **T3**). With that notation in mind, we can define $G(\beta)$ properly:

$$G(\beta) = -k(\boldsymbol{x}', \boldsymbol{x}'') = \sum_{n=1}^{N} -\log \left[\sum_{j=1}^{K} e_j' e_j'' \right] + \sum_{n=1}^{N} \log \left[\sum_{k=1}^{K} e_k' \sum_{p=1}^{K} e_p'' \right]$$

We will focus on the first term: $G_1^n(\beta) = G_1(\beta) = \log \left[\sum_{j=1}^{K} e_j' e_j'' \right]$.

$$\frac{d}{d\beta} G_1(\beta) = \frac{\sum_{j=1}^{K} (F_j' + F_j'') e_j' e_j''}{\sum_{j=1}^{K} e_j' e_j''}$$

The derivative $\frac{d}{d\beta} e_j' e_k''$ will be used recurrently in this section:

$$\frac{d}{d\beta} e_j' e_k'' = F_j' e_j' e_k'' + e_j' F_k'' e_k'' = (F_j' + F_k'') e_j' e_k''$$

The second derivative is straightforward:

$$\frac{d^2}{d\beta^2}G_1(\beta) = \frac{\sum_{j=1}^{K}(F'_j + F''_j)^2 e'_j e''_j}{\sum_{j=1}^{K} e'_j e''_j} - \frac{\left(\sum_{j=1}^{K}(F'_j + F''_j)e'_j e''_j\right)^2}{\left(\sum_{j=1}^{K} e'_j e''_j\right)^2}$$

We impose the convexity condition and see whether it can be contradicted.

$$\frac{d^2}{d\beta^2}G_1(\beta) > 0 \iff \left(\sum_{j=1}^{K} e'_j e''_j\right)\left(\sum_{j=1}^{K}(F'_j + F''_j)^2 e'_j e''_j\right) - \left(\sum_{j=1}^{K}(F'_j + F''_j)e'_j e''_j\right)^2 > 0$$

Using the distributive property of the product over the sum, we can reindex our expression:

$$\sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j + F''_j)^2 e'_j e''_j e'_k e''_k - \sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j + F''_j)(F'_k + F''_k)e'_j e''_j e'_k e''_k > 0 \iff$$

$$\iff \sum_{k=1}^{K}\sum_{j=1}^{K}[(F'_j + F''_j) - (F'_k + F''_k)](F'_j + F''_j)e'_j e''_j e'_k e''_k > 0$$

As we can see, $\Delta_{(jj),(kk)}$ corresponds to the difference in the cost attributed to reference class j and the cost attributed to class k, accumulated over $\boldsymbol{x}', \boldsymbol{x}''$. We can intuitively devise some symmetry in these terms, and we formalize it as follows:

$$E_{jk} = e'_j e''_j e'_k e''_k = E_{kj}$$

$$\Delta_{(jj),(kk)} = (F'_j + F''_j) - (F'_k + F''_k) = (F'_j - F'_k) + (F''_j - F''_k) = -\Delta_{(kk),(jj)}$$

Even if $\Delta_{(jj),(jj)} = 0$, we will still include this term to facilitate with the indexing. Overall, the sum can be expressed as:

$$\sum_{k=1}^{K}\sum_{j=1}^{K}[(F'_j+F''_j)-(F'_k+F''_k)](F'_j+F''_j)e'_j e''_j e'_k e''_k = \sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j+F''_j)E_{jk}\Delta_{(jj),(kk)} = \sum_{k=1}^{K}\sum_{j=1}^{K} S_{(jj),(kk)}$$

Then, the pairwise sum of symmetric combinations of indexes k and j yields

$$\begin{aligned} S_{(jj),(kk)} + S_{(kk),(jj)} &= (F'_j + F''_j)E_{jk}\Delta_{(jj),(kk)} + (F'_k + F''_k)E_{kj}\Delta_{(kk),(jj)} \\ &= E_{jk}\Delta_{(jj),(kk)}[(F'_j + F''_j) - (F'_k + F''_k)] = E_{jk}\Delta^2_{(jj),(kk)} > 0 \end{aligned}$$

Given that the indexing sets in our nested sum are the same, it's straightforward to see that all the terms will be strictly positive, and the overall sum will be zero only if $e_j = 0\ \forall j = \{1, ...K\}$, which is not possible in a classification setting since $\beta \in \mathbb{R}^+$. We end up with the following expression:

$$\frac{d^2}{d\beta^2}G_1(\beta) = \sum_{k=1}^{K}\sum_{j<k}^{K} E_{jk}\Delta^2_{(jj),(kk)} > 0$$

Now we proceed analogously with the second term:

$$G_2^n(\beta) = G_2(\beta) = \log\left[\sum_{j=1}^{K} e'_j \sum_{k=1}^{K} e''_k\right] = \log\left[\sum_{k=1}^{K}\sum_{j=1}^{K} e'_j e''_k\right]$$

$$\frac{d}{d\beta}G_2(\beta) = \frac{\sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j + F''_k)e'_j e''_k}{\sum_{k=1}^{K}\sum_{j=1}^{K} e'_j e''_k}$$

$$\frac{d^2}{d^2\beta}G_2(\beta) = \frac{\sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j + F''_k)^2 e'_j e''_k}{\sum_{k=1}^{K}\sum_{j=1}^{K} e'_j e''_k} - \frac{\left(\sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j + F''_k) e'_j e''_k\right)^2}{\left(\sum_{k=1}^{K}\sum_{j=1}^{K} e'_j e''_k\right)^2} > 0 \iff$$

$$\iff \left(\sum_{k=1}^{K}\sum_{j=1}^{K} e'_j e''_k\right)\left(\sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j + F''_k)^2 e'_j e''_k\right) - \left(\sum_{k=1}^{K}\sum_{j=1}^{K}(F'_j + F''_k) e'_j e''_k\right)^2 > 0 \iff$$

$$\iff \sum_{k=1}^{K}\sum_{q=1}^{K}\sum_{j=1}^{K}\sum_{i=1}^{K}(F'_j + F''_k)^2 e'_j e''_k e'_i e''_q - (F'_j + F''_k) e'_j e''_k (F'_i + F''_q) e'_i e''_q > 0 \iff$$

$$\iff \sum_{k=1}^{K}\sum_{q=1}^{K}\sum_{j=1}^{K}\sum_{i=1}^{K}(F'_j + F''_k) e'_j e''_k e'_i e''_q [(F'_j + F''_k) - (F'_i + F''_q)] > 0$$

We can define as well:

$$\frac{d^2}{d^2\beta}G_2(\beta) = \sum_{k=1}^{K}\sum_{q=1}^{K}\sum_{j=1}^{K}\sum_{i=1}^{K} S_{(jk),(iq)} = \sum_{k=1}^{K}\sum_{q=1}^{K}\sum_{j=1}^{K}\sum_{i=1}^{K}(F'_j + F''_k) E_{(jk),(iq)} \Delta_{(jk),(iq)}$$

$$E_{(jk),(iq)} = e'_j e''_k e'_i e''_q = E_{(ik),(jq)} = E_{(jq),(ik)} = E_{(iq),(jk)}$$

$$\Delta_{(jk),(iq)} = (F'_j - F'_i) + (F''_k - F''_q) = -\Delta_{(iq),(jk)}$$

The symmetry arises when adding two elements that have mirror indexes in both $\boldsymbol{x}'$ and $\boldsymbol{x}''$.

$$\begin{aligned} S_{(jk),(iq)} + S_{(iq),(jk)} &= (F'_j + F''_k) E_{(jk),(iq)} \Delta_{(jk),(iq)} + (F'_i + F''_q) E_{(iq),(jk)} \Delta_{(iq),(jk)} \\ &= E_{(jk),(iq)} \Delta_{(jk),(iq)} [(F'_j + F''_k) - (F'_i + F''_q)] = E_{(jk),(iq)} \Delta^2_{(jk),(iq)} > 0 \end{aligned}$$

Given that symmetries are independent for $\boldsymbol{x}'$ and $\boldsymbol{x}''$, we end up with a similar expression:

$$\frac{d^2}{d\beta^2}G_2(\beta) = \sum_{k=1}^{K}\sum_{q<k}^{K}\sum_{j=1}^{K}\sum_{i<j}^{K} E_{(jk),(iq)} \Delta^2_{(jk),(iq)} > 0$$

Even if a further simplified version can be obtained, this one will allow us to complete the proof. We can now define the function $G(\beta)$ as the sum of the two terms:

$$\frac{d^2}{d\beta^2}G(\beta) = \sum_{n=1}^{N}\left[\sum_{k=1}^{K}\sum_{q<k}^{K}\sum_{j=1}^{K}\sum_{i<j}^{K} E_{(jk),(iq)} \Delta^2_{(jk),(iq)} - \sum_{k=1}^{K}\sum_{q<k}^{K} E_{(qq),(kk)} \Delta^2_{(qq),(kk)}\right]$$

where we can clearly see that the particular case $\{k = j, q = i\}$ cancels the negative terms:

$$\begin{aligned} \frac{d^2}{d\beta^2}F^n(\beta) &= \sum_{k=1}^{K}\sum_{q<k}^{K}\sum_{j=\{1:K\}\setminus\{k\}}\sum_{i=\{1:K|i<j\}\setminus\{k\}} E_{(jk),(iq)} \Delta^2_{(jk),(iq)} \\ &+ \sum_{k=1}^{K}\sum_{q<k}^{K} E_{(qq),(kk)} \Delta^2_{(qq),(kk)} - \sum_{k=1}^{K}\sum_{q<k}^{K} E_{(qq),(kk)} \Delta^2_{(qq),(kk)} = \\ &= \sum_{k=1}^{K}\sum_{q<k}^{K}\sum_{j=\{1:K\}\setminus\{k\}}\sum_{i=\{1:K|i<j\}\setminus\{k\}} E_{(jk),(iq)} \Delta^2_{(jk),(iq)} > 0 \end{aligned}$$

Which proves that $G(\beta)$ is absolutely convex in $\mathbb{R}^+$:

$$\frac{d^2}{d\beta^2}G(\beta) = \sum_{n=1}^{N} \frac{d^2}{d\beta^2}G^n(\beta) = \sum_{n=1}^{N} \left[\sum_{k=1}^{K} \sum_{q<k}^{K} \sum_{j=\{1:K\}\setminus\{k\}} \sum_{i=\{1:K|i<j\}\setminus\{k\}} E_{(jk),(iq)} \Delta^2_{(jk),(iq)} \right] > 0$$

We must note that on the limit $\beta \to \infty$ the curvature is not defined, so it will be always a good practice to start the numerical procedure at a value $\beta_0 = 0^+$:

$$\lim_{\beta \to 0^+} \frac{d^2}{d\beta^2}G(\beta) > 0$$

□

Appendix B

Again Something

Blah, blah ...

Bibliography

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute for Dynamic Systems and Control
Prof. Dr. R. D'Andrea, Prof. Dr. L. Guzzella

Title of work:

Improved robustness of deep learning models through posterior agreement-based model selection

Thesis type and date:

Master Thesis, September 2024

Supervision:

Dr. Joao Borges de Sa Carvalho, Dr. Alessandro Torcinovich
Prof. Dr. Joachim M. Buhmann

Student:

Name:	Victor Jimenez Rodriguez
E-mail:	vjimenez@student.ethz.ch
Legi-Nr.:	97-906-739
Semester:	5

Statement regarding plagiarism:

By signing this statement, I affirm that I have read and signed the Declaration of Originality, independently produced this paper, and adhered to the general practice of source citation in this subject-area.

Declaration of Originality:

`http://www.ethz.ch/faculty/exams/plagiarism/confirmation_en.pdf`

Zurich, 1. 5. 2024: _____