Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute for
Dynamic Systems and Control
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

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	$[^{\mathrm{K}}/_{\mathrm{s}}]$

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 $f^* : \mathcal{X} \rightarrow \mathcal{Y}$ by means of an approximated function $f \in \mathcal{F}$ using a finite set of observations.

Definition (Supervised dataset)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} \subseteq \mathbb{R}^t$ be the input and output spaces, respectively. Let X be the 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 $\underline{X}$ paired with its f^*-mapped output values.*

$$D = \{(\boldsymbol{x}_n, f^*(\boldsymbol{x}_n))\}_{n \in [N]}$$

2.1.1 Empirical risk minimization

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

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

where $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ denotes a loss function. Glivenko-Cantelli theorem allow us to estimate the expected risk with its empirical (plug-in) equivalent 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*

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

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

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

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 usually tweaqued to include a regularization term penalizing complexity.

Definition (Regularized empirical risk)**.** *Let $\Omega : \mathcal{F} \to \mathbb{R}$ be a complexity measure. 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.

The RRM problem now includes a hyperparameter λ, that will be selected so that the generalization error is minimized.

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 shows the data transformation performed by a single node.

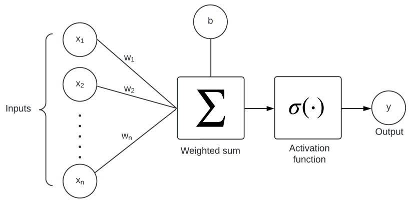

Figure 2.1: The output of a node is computed by applying a non-linear activation function $\sigma(\cdot)$ to the weighted sum of its inputs.

Let $\boldsymbol{x}_k \in \mathbb{R}^{d_k}$ be the input to the layer k, 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.

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

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 algorithms, 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 Generalization error

As we mentioned in the first lines of this chapter, the input of learning algorithms are realizations of the random variable X defined over the input space $\mathcal{X}$. The implicit randomness embedded in the sampling process extends to the outcome of algorithms, even when performing a deterministic sequence of operations. An alternative intuition of generalization arises from this perspective, in the sense that a learning algorithm should learn the same function f when trained on different realizations of the same experiment. This intuition will be formalized in this section.

2.3.1 Posterior distribution

The function $f \in \mathcal{F}$ was defined mapping $\mathcal{X} \longmapsto \mathcal{Y}$, in the sense that an observation $x \in \mathcal{X}$ is mapped to a prediction $f(x) \in \mathcal{Y}$. In order to formalize the randomness of the data sampling process, we will redefine $\mathcal{X}$ so that its elements are realizations of the simple random sample $\underline{\mathrm{X}}$, which are then mapped to a set of outcomes θ.

Definition (Distribution of $\underline{\mathrm{X}}$)**.** *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 refer to this probability distribution as $\mathbf{P}_X$ to avoid notation clutter.

Definition (Hypothesis class)**.** *A data science algorithm learns a function f implementing the following mapping:*

$$f : \mathcal{X} \longmapsto \Theta$$
$$\boldsymbol{x} = (x_1, \ldots, x_N) \sim X \longmapsto (f(x_1), \ldots, f(x_N)) = \theta$$

The output space of hypothesis Θ represents all possible outcomes of a function f learned on a dataset sampled from $\mathcal{X}$.

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 equivalent, 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 $\mathcal{F}_\Gamma$ derived from 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 $\boldsymbol{x}$ of $\underline{\mathrm{X}}$ 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)**.** *Let $\tau \in \mathbb{T}$ be a transformation of the sampling experiment X. The dependency of the data on the experimental design will be captured by the index τ, and we will implicitely consider $\underline{X}$ to be a sample of $\tau \circ X$. We require a sufficiently rich set of experiments $\mathbb{T}$ such that every hypothesis $\theta \in \Theta$ is the (most likely) outcome of some realization $\boldsymbol{x}$.*

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

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

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

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

$\mathbf{P}^f$ *establishes the stochastic relation between data and hypothesis.*

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 given $\underline{\mathrm{X}}$ as

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

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 generalization error within this framework, we will proceed in an analogous way as we did in the previuous section. We will consider two datasets D_t and D_v that arise from two different sampling realizations. In this case, however, we will make explicit the fact that they encode different instantiations of the randomness associated with the sampling, and thus denote $\boldsymbol{x}'$ and $\boldsymbol{x}''$ as realizations of the two simple random samples $\underline{\mathrm{X}}', \underline{\mathrm{X}}'' \overset{\text{iid}}{\sim} X \mid \tau$, respectively, and associate each dataset with one of them. The τ indexing is now explicit, since both samples are conditionally independent given the experiment:

$$\mathbf{P}(\underline{\mathrm{X}}', \underline{\mathrm{X}}'') = \mathbf{P}(\underline{\mathrm{X}}' \mid \tau)\mathbf{P}(\underline{\mathrm{X}}'' \mid \tau)$$

The selection principles for a posterior distribution will be the following:

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

Definition (Description length)**.** *Let $\mathcal{F}_{\boldsymbol{\Gamma}}(\cdot)$ be the function class containing all functions represented by the parametrization $\boldsymbol{\Gamma}$. Let $\mathbf{P}_{\boldsymbol{\Gamma}}$ be the universal distribution relative to $\mathcal{F}_{\boldsymbol{\Gamma}}$ fulfilling the minimum description length principle. The description length of a function $f_\gamma \in \mathcal{F}_{\boldsymbol{\Gamma}}$ 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.

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

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

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

It amounts to the expected normalized loss over the validation data $\boldsymbol{x}''$ computed over the distribution on the hypothesis expressed by the posterior on the training data $\boldsymbol{x}'$. Intuitively, a lower generalization error is achieved when good quality posteriors on $\boldsymbol{x}''$ are assigned a high probability by the posterior on $\boldsymbol{x}'$.

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.

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

□

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

$$\log(\mathbb{E}[\cdot]) \leq \mathbb{E}[\log(\cdot)]$$

2.3.3 Maximum posterior agreement

Describe max posterior agreeement problem. All Joachim, but be brief.

2.4 Posterior agreement kernel

2.4.1 Classification problem

Statement of the classification problem. Follow structure from

2.5 DS algorithms, NNs and distribution shift

MOST COMPLICATED SECTION.
- Cite Joachim, introduction to DS algorithms. Introduce notation that will follow the rest of the thesis. Particularize for NNs (work to do).
- Explain function representation problem in NNs, why it is important. (look at phd zotero)
- Read about info theory, why description length
- Start with two datasets, training and validation, and their distributions (also Joachim)
- Derive the OOD-error from the previous theory. More detail than Joachim.

2.6 From OOD error to PA

- Definitions and proofs (ommitted by Joachim) that lead to PA
- Intuition behind PA, plot.
- Read more about previous work from Joachim, where PA has worked.
- Include (briefly) the binary symmetric channel as an example, as it is important for Joachim and will be referenced later.

2.7 The PA kernel

- Follow derivation overleaf, include factorization theorem.
- Explain change of perspective, think of (binary) classification as a binary symmetric channel.

2.7.1 Properties

(all in overleaf or papers)
- Positive definiteness, explain why important
- Symmetry, explain why important
- Convexity analysis.
- Problem with binary classification

2.7.2 Example

- Analytical results obtained, overleaf.

2.8 Generalization error in the hypothesis space

- Alternative formulation
- Explain why important, and formalize the data augmentation strategy (presentation)

Chapter 3

Results and discussion

Blah, blah ...

Chapter 4

Discussion

Blah, blah ...

Appendix A

Something

Blah, blah ...

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, 24. 4. 2024: ______________________________