Effective Fine-tuning via Base Model Rescaling: A Random Matrix Theory Perspective

Master's Thesis report

15 April 2025 - 15 October 2025

Aymane EL FIRDOUSSI

Academic supervisors
Alexandre D'ASPREMONT (MVA, ENS Paris)
Olivier FERCOQ (Télécom Paris)

Lab supervisors
El Mahdi CHAYTI (EPFL)
Martin JAGGI (EPFL)

Acknowledgement

I would like to express my sincere gratitude to my internship supervisors, El Mahdi Chayti and Professor Martin Jaggi, for their warm welcome to the Machine Learning and Optimization (MLO) Lab at EPFL and for their invaluable guidance throughout the project. El Mahdi closely supervised my work on a daily basis, and our numerous discussions were a constant source of insight and inspiration. Over time, I came to regard him not only as a mentor but also as a close friend. I am grateful for the strong relationship we built, which we look forward to continuing beyond this internship. I also wish to thank Professor Jaggi for his constructive remarks and thoughtful criticism, which helped us reach the full potential of this project. My gratitude extends as well to all members of the MLO Lab for their team spirit, dynamism, and hospitality, which made me feel welcome and fully integrated during my stay at EPFL. Finally, I would like to acknowledge my professors at ENS Paris an Télécom Paris, in particular Prof. Alexandre d'Aspremont and Prof. Olivier Fercoq, for the solid knowledge imparted during coursework. Their teaching has laid a strong foundation for my growth and development, and their contributions have been integral to my academic and professional journey.

Abstract

Fine-Tuning has proven to be highly effective in adapting pre-trained models to perform better on new desired tasks with minimal data samples. Among the most widely used approaches are reparameterization methods, which update a target module by augmenting its frozen weight matrix with an additional trainable weight matrix. The most prominent example is Low Rank Adaption (LoRA) (Hu et al., 2022) which gained significant attention in recent years. In this work, we introduce a new class of reparameterization methods for transfer learning, designed to enhance the generalization ability of fine-tuned models. We establish the effectiveness of our approach in a high-dimensional binary classification setting using tools from Random Matrix Theory, and further validate our theoretical findings through more realistic experiments such as fine-tuning large language models. Finally, we extend our analysis to multi-source and regression transfer settings, highlighting the generality and robustness of our approach. Overall, this work provides both theoretical insight and practical algorithms that bridge Random Matrix Theory and efficient model adaptation.

Contents

1 Introduction and contributions **5**

2 Related work **6**

3 Theoretical setting and mathematical background **8**
3.1 Theoretical Setting . . . 8
3.2 RMT Background . . . 10

4 Main Theoretical Results **12**
4.1 RMT Assumptions . . . 12
4.2 Theoretical performance with Ridge source classifier . . . 12
4.3 Theoretical performance with arbitrary source classifier . . . 15
4.4 Conclusion of our theory . . . 17

5 Experiments **17**
5.1 Within our theoretical model: Linear Binary Classification . . . 17
5.2 Beyond our theoretical model: Supervised Fine-tuning for LLMs . . . 19

6 Multi-source Transfer Learning **22**
6.1 Asymptotic distribution of the test . . . 22
6.2 Characterization of the optimal scaling factors . . . 23

7 The case of regression: fine-tuning a weight matrix **24**

8 Discussion and Conclusion **28**

A Useful results **33**
A.1 General lemmas . . . 33
A.2 Deterministic equivalents . . . 33

B RMT Analysis of the fine-tuned classifier **39**
B.1 Test Expectation . . . 39
B.2 Test Variance . . . 40
B.3 Finding optimal scaling parameter . . . 45

C RMT analysis for arbitrary source classifier **45**
C.1 Test Expectation . . . 46
C.2 Test variance . . . 46

D Extension to Multi-Source Transfer Learning **49**
D.1 Test Expectation 49
D.2 Test variance 50

E Extension to Linear Regression Transfer **53**
E.1 Preliminary results 53
E.2 Test error 54
E.3 Optimal scaling parameter 60

F LLMs experimental details **60**
F.1 Hyperparameters 60
F.2 Values of scaling parameters 60

1 Introduction and contributions

Large foundational models have driven major advances in artificial intelligence across domains such as computer vision and natural language processing. Examples include transformer-based models (Vaswani et al., 2017) operating in natural language domain, known as Large Language Models (LLMs), such as `Gemini` (Team et al., 2023) and `Llama` (Grattafiori et al., 2024), and on the vision domain such as Vision Transformers (Dosovitskiy et al., 2020). Such models are specifically known for their relatively large size and massive training corpus, which makes them more powerful and adapted for many use cases. However, even with their extensive pre-training, these large models may not excel at some specific tasks without further adjustment. To achieve improvements of this kind, a process known as Fine-Tuning is often needed. Fine-tuning involves adapting a pre-trained model to a target task by continuing its training on task-specific data. Unlike training from scratch, fine-tuning leverages the general representations learned during pre-training and refines them to capture task-relevant information, thereby improving performance while reducing data and computational requirements. The most common class of fine-tuning methods is Supervised Fine-Tuning (SFT), which relies on labeled data in that process, and one of the most popular lightweight SFT methods is Low-Rank Adaptation (LoRA) (Hu et al., 2022), which updates the desired module by adding a low-rank perturbation to the original (frozen) weight matrix.
In this project, we study fine-tuning through the lens of Random Matrix Theory (RMT), where we introduce a theoretical framework to understand and improve transfer learning. Leveraging the theoretical findings, our key practical idea in the context of LoRA is to scale the frozen weights row-wise with a vector $\boldsymbol{\alpha}$ before adaptation, thereby adding a new degree of freedom to the fine-tuning process. We show that this modification leads to an optimal scaling factor $\boldsymbol{\alpha}^*$, which is typically different from the standard choice ($\boldsymbol{\alpha} = 1$). We analyze this framework in a high-dimensional binary classification setting under a Gaussian Mixture Model, proving the existence of an optimal $\boldsymbol{\alpha}^*$ while providing its closed-form expression in terms of scalar data-dependent quantities. We then validate our theoretical insights on real tasks, including transfer learning benchmarks and large language model fine-tuning, in addition to extending the theoretical results to other classification and regression settings.

Summary of contributions. Our work on fine-tuning is novel and presents many contributions to the community, which we summarize as follows:

1. We introduce a new class of Supervised Fine-Tuning algorithms characterized by an additional scaling parameter $\boldsymbol{\alpha}$.

2. We theoretically prove the existence of an optimal parameter $\boldsymbol{\alpha} \neq 1$ and derive its expression in binary classification.

3. We propose an algorithm for finding such optimal $\boldsymbol{\alpha}$ for complex scenarios such as fine-tuning language models.

2 Related work

Transfer Learning foundations. Transfer Learning (TL) studies how knowledge acquired in a source task or domain can be reused to improve learning in a related target task. Early surveys (Pan & Yang, 2009; Weiss et al., 2016) outlined key settings such as domain adaptation and multitask learning. A foundational study by Ben-David et al. (2010) established generalization bounds that relate target error to source error and distributional divergence, providing theoretical criteria for effective transfer. Building on this, Maurer et al. (2016) showed that shared representations across tasks can reduce sample complexity in multitask settings, further emphasizing the role of representation learning. Tripuraneni et al. (2020) analyzed the impact of task diversity on TL and show that by learning a shared feature representation from diverse tasks, the amount of data needed for a new task is greatly reduced, scaling only with the complexity of the new task itself, rather than the complexity of the entire system. Other works such as (Hanneke & Kpotufe, 2024; Zhang et al., 2021; Klivans et al., 2024; Kpotufe & Martinet, 2021; Cai & Wei, 2021; Reeve et al., 2021) have tackled TL theoretically each from a different perspective and on a different setting (regression or classification).

Fine-Tuning pre-trained models. With the advent of large-scale pretraining, fine-tuning has become the dominant strategy for transfer learning. The most popular fine-tuning techniques are Supervised Fine-Tuning (SFT) and fine-tuning with Reinforcement Learning (RL). RL-based approaches such as RLHF (Ouyang et al., 2022), DPO (Rafailov et al., 2023), GRPO (Ramesh et al., 2024; Guo et al., 2025) and other variants are specifically effective on reasoning and mathematics tasks, where they often outperform SFT (Shenfeld et al., 2025). In this paper, however, we only focus on SFT techniques. In fact, SFT extends the training of the given pre-trained model using labeled data. However, as the size of used pre-trained models is generally large, a common approach to fine-tuning is to modify a small fraction of the model's parameters while leaving most of them unmodified. This strategy, known as Parameter-Efficient Fine-Tuning (PEFT) (Xu et al., 2023), aims to achieve strong performance with minimal parameter updates. PEFT methods are usually grouped into three categories: additive, selective, and reparameterized (Ji et al., 2025).

Additive Fine-Tuning. The most popular additive fine-tuning approach is Adapters (Houlsby et al., 2019; He et al., 2021), which adds a minimal number of new trainable parameters that are strategically positioned within the model architecture, while keeping the rest of the model frozen. Variants explore different placement strategies, scaling, and modular reuse (Pfeiffer et al., 2020; Karimi Mahabadi et al., 2021). These added layers/modules act as computational bottlenecks, refining the model's output while leveraging the existing pre-trained parameters.

Selective Fine-Tuning. Unlike additive PEFT, selective PEFT does not add extra layers or modules to the original model, but updates a specific subset of the existing parameters within the model. This is achieved for instance by applying a binary mask to the model's parameters, where each element of the mask is either 0 or 1, indicating whether the corresponding parameter should be updated during fine-tuning.

Popular selective techniques include Diff pruning (Guo et al., 2020), FishMask (Sung et al., 2021) and PaFi (Liao et al., 2023).

Reparameterized Fine-Tuning. Reparameterization-based fine-tuning adapts a model by expressing its parameters in an alternative form, commonly through a low-rank decomposition, to reduce training costs, while the full weight matrices are reconstructed for inference. The most common technique in this class is Low Rank Adaptation (LoRA) (Hu et al., 2022), which introduces small, trainable matrices operating alongside the pre-trained weights to inject task-specific updates without burdening the inference process. Many extensions were proposed to enhance the efficiency of LoRA by either acting on the initialization of the low rank modules (Hayou et al., 2024a), their learning rates (Hayou et al., 2024b), normalizing the updates (Liu et al., 2024), setting adaptive ranks (Kim et al., 2024; Lu et al., 2024), finding optimal placements for LoRA modules (Hayou et al., 2025), and more (Zhang et al., 2023b; Dettmers et al., 2023; Kopiczko et al., 2023; Zhang et al., 2023a; Tian et al., 2024; Jiang et al., 2024).

While prior work has proposed numerous variants of LoRA that adjust ranks, placements, or normalization schemes, little attention has been paid to the scaling of frozen weights themselves. Our work is complementary to these approaches: rather than modifying the structure of the low-rank modules, we focus on the scaling dynamics of the pre-trained component and provide the first theoretical analysis of its impact using Random Matrix Theory.

3 Theoretical setting and mathematical background

It is common in Machine Learning research that in order to prove the effectiveness of some method or algorithm, we theoretically analyze it in simple settings and then use the obtained results to build insights and intuitions on more complex settings (such as LLMs). Thus, to prove the effectiveness of our new family of fine-tuning algorithms, we will theoretically analyze a binary classification setting under a Gaussian Mixture Model (GMM) using tools from Random Matrix Theory (RMT). Through this analysis, we will prove the existence of an optimal scaling parameter $\alpha^\star$ and derive its exact theoretical formulation for these settings.

3.1 Theoretical Setting

The goal is to fine-tune a linear classifier, initially pretrained on a dataset called **source**, in order to perform a **target** task given a relatively small target data corpus.

Pre-training phase. We consider that we are given pairs of pre-training (source) data samples $\{(\tilde{\boldsymbol{x}}_i, \tilde{y}_i)\}_{i=1}^N$ that are distributed, for $\tilde{\boldsymbol{x}}_i \in \mathscr{C}_a$ with $a \in \{1, 2\}$, as follows:

$$\tilde{\boldsymbol{x}}_i \in \mathscr{C}_a \quad \Leftrightarrow \quad \begin{cases} \tilde{\boldsymbol{x}}_i = \boldsymbol{\mu}_a + \tilde{\boldsymbol{z}}_i, \quad \tilde{\boldsymbol{z}}_i \sim \mathscr{N}(\mathbf{0}, \mathbf{I}_p), \\ \tilde{y}_i = (-1)^a. \end{cases} \tag{1}$$

For convenience and without loss of generality, we further assume that $\boldsymbol{\mu}_a = (-1)^a \boldsymbol{\mu}$ for some vector $\boldsymbol{\mu} \in \mathbb{R}^p$. This setting can be recovered by subtracting $\frac{\boldsymbol{\mu}_1 + \boldsymbol{\mu}_2}{2}$ from each data point, as such $\boldsymbol{\mu} = \frac{\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1}{2}$ and therefore the SNR $\|\boldsymbol{\mu}\|$ controls the difficulty of the classification problem, in the sense that large values of $\|\boldsymbol{\mu}\|$ yield a simple classification problem whereas when $\|\boldsymbol{\mu}\| \to 0$, the classification becomes impossible.

Remark 3.1 (On the data model)**.** *Note that the above data assumption can be relaxed (generalized) to considering $\boldsymbol{x}_i = \boldsymbol{\mu}_a + \mathbf{C}_a^{\frac{1}{2}} \boldsymbol{z}_i$ where $\mathbf{C}_a$ is some semi-definite covariance matrix and $\boldsymbol{z}_i$ are random vectors with i.i.d entries of mean 0, variance 1 and bounded fourth order moment. In fact, in the high-dimensional regime when $n, p \to \infty$, the asymptotic performance of the classifier considered subsequently is universal in the sense that it depends only on the statistical means and covariances of the data (Louart & Couillet, 2018; Seddik et al., 2020; Dandi et al., 2024). However, such a general setting comes at the expense of more complex formulas, making the above isotropic assumption more convenient for readability and better interpretation of our findings.*

Denoting $\tilde{\mathbf{X}} = [\tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_N] \in \mathbb{R}^{p \times N}$ the data matrix and $\tilde{\boldsymbol{y}} = [\tilde{y}_1, \ldots, \tilde{y}_N]^\top \in \mathbb{R}^N$ the corresponding labels vector, we have in matrix form:

$$\tilde{\mathbf{X}} = \boldsymbol{\mu} \tilde{\boldsymbol{y}}^\top + \tilde{\mathbf{Z}}, \tag{2}$$

where $\tilde{\mathbf{Z}}$ is a random matrix with $\mathscr{N}(0, 1)$ i.i.d. entries.
We then consider training a classifier, called $\tilde{\boldsymbol{w}}$, on this source dataset by optimizing:

$$\min_{\boldsymbol{w}} \frac{1}{N} \sum_{i=1}^N \ell(\boldsymbol{w}^\top \boldsymbol{x}_i, y_i) + \tilde{\gamma} \|\boldsymbol{w}\|_2^2 \tag{3}$$

for some loss function ℓ and a positive regularization parameter $\tilde{\gamma} > 0$. Taking a generic loss function, such as the binary cross entropy, leads to **intractable** solution $\tilde{\boldsymbol{w}}$. Fortunately, Mai & Liao (2024) show that in the case of a Gaussian mixture data model or more generally a data distribution with finite fourth-order moment (remark 3.1), it is possible to optimize such classifier using the squared (L^2) loss function, which also gives a closed-form solution to this problem. Thus, taking $\ell(x, y) = (x - y)^2$ leads to the following optimization problem:

$$\tilde{\boldsymbol{w}} = \arg\min_{\boldsymbol{v}} \frac{1}{N} \left\| \tilde{\mathbf{X}}^\top \boldsymbol{v} - \tilde{\boldsymbol{y}} \right\|_2^2 + \tilde{\gamma} \|\boldsymbol{v}\|_2^2, \tag{4}$$

Which gives us the following solution:

$$\tilde{\boldsymbol{w}} = \frac{1}{N} \mathbf{R} \tilde{\mathbf{X}} \tilde{\boldsymbol{y}}, \quad \mathbf{R} = \left(\frac{1}{N} \tilde{\mathbf{X}} \tilde{\mathbf{X}}^\top + \tilde{\gamma} \mathbf{I}_p \right)^{-1} \tag{5}$$

Fine-tuning phase. During the fine-tuning phase, we suppose that we are given pairs of target data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ with $y_i \in \{-1, 1\}$ that are distributed such that $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{p \times n}$ is given by:

$$\mathbf{X} = \boldsymbol{\mu}_\beta \boldsymbol{y}^\top + \mathbf{Z}, \quad \boldsymbol{\mu}_\beta = \beta \boldsymbol{\mu} + \boldsymbol{\mu}^\perp, \tag{6}$$

where $\mathbf{Z}$ is a random matrix with $\mathcal{N}(0, 1)$ i.i.d. entries, $\boldsymbol{\mu}^\perp$ is an orthogonal vector to $\boldsymbol{\mu}$ and the factor $\beta \in \mathbb{R}$ quantifies the **alignment** between the source and target data, as we have that: $\langle \boldsymbol{\mu}_\beta, \boldsymbol{\mu} \rangle = \beta \|\boldsymbol{\mu}\|^2$. The goal is to leverage the original classifier $\tilde{\boldsymbol{w}}$ to train a new classifier on this target dataset. The standard reparameterization approach for doing so is modeled by adding a trainable classifier to $\tilde{\boldsymbol{w}}$, and then training it on the target data, i.e solving:

$$\min_{\boldsymbol{v}} \frac{1}{n} \left\| \mathbf{X}^\top (\tilde{\boldsymbol{w}} + \boldsymbol{v}) - \boldsymbol{y} \right\|_2^2 + \gamma \|\boldsymbol{v}\|_2^2$$

However, we can generalize this method even further by introducing a scaling parameter to the pre-trained classifier $\tilde{\boldsymbol{w}}$, which adds up a new degree of freedom to this learning process and makes a better use of the pretraining phase. Thus, leveraging the pre-trained weights $\tilde{\boldsymbol{w}} \in \mathbb{R}^p$, we consider the training of adapter weights $\boldsymbol{a}$ as:

$$\boldsymbol{a} = \arg\min_{\boldsymbol{v}} \frac{1}{n} \left\| \mathbf{X}^\top (\alpha \tilde{\boldsymbol{w}} + \boldsymbol{v}) - \boldsymbol{y} \right\|_2^2 + \gamma \|\boldsymbol{v}\|_2^2, \tag{7}$$

for a scalar $\alpha \in \mathbb{R}$. Solving the previous minimization problem, $\boldsymbol{a}$ expresses as:

$$\boldsymbol{a} = \frac{1}{n} \left(\frac{1}{n} \mathbf{X}\mathbf{X}^\top + \gamma \mathbf{I}_p \right)^{-1} \left(\mathbf{X}\boldsymbol{y} - \alpha \mathbf{X}\mathbf{X}^\top \tilde{\boldsymbol{w}} \right). \tag{8}$$

We define the resolvent matrices $\mathbf{Q}$ and $\mathbf{R}$ by:

$$\mathbf{Q} = \left(\frac{1}{n} \mathbf{X}\mathbf{X}^\top + \gamma \mathbf{I}_p \right)^{-1}, \quad \mathbf{R} = \left(\frac{1}{N} \tilde{\mathbf{X}} \tilde{\mathbf{X}}^\top + \tilde{\gamma} \mathbf{I}_p \right)^{-1}, \tag{9}$$

Then our obtained fine-tuned classifier $\boldsymbol{w}_\alpha$ writes:

$$\boldsymbol{w}_\alpha = \alpha \tilde{\boldsymbol{w}} + \boldsymbol{a} = \frac{1}{n} \mathbf{Q}(\gamma) \mathbf{X}\boldsymbol{y} + \alpha\gamma \mathbf{Q} \tilde{\boldsymbol{w}}$$

We denote by $\boldsymbol{w} \equiv \boldsymbol{w}_0$ the classifier obtained through learning directly on target data (without fine-tuning), which is given by:

$$\boldsymbol{w} = \frac{1}{n}\mathbf{Q}(\gamma)\mathbf{X}\boldsymbol{y} \tag{No-FT}$$

Then we finally get the expression of our α-Fine-tuned classifier as follows:

$$\boldsymbol{w}_\alpha = \boldsymbol{w} + \alpha\gamma\mathbf{Q}\tilde{\boldsymbol{w}} \tag{α-FTC}$$

Remark 3.2 (About the interpretability of our fine-tuned classifier)**.** *Remark that the parameter α introduced in the expression of the fine-tuned classifier $\boldsymbol{w}_\alpha$ characterizes the contribution of each training dataset (source and target) to the test performance on the target task. In fact, since the prediction of the class label does not change by multiplying $\boldsymbol{w}_\alpha$ by a positive constant, then by taking a positive α and for $\rho = \frac{\alpha}{1+\alpha} \in (0,1)$, the fine-tuned classifier is equivalent to this convex weighted classifier:*

$$\boldsymbol{w}_\rho = \rho\tilde{\boldsymbol{w}} + (1-\rho)\boldsymbol{a}$$

and therefore, this new parameter ρ can be interpreted as the percentage of the contribution of the source task to the test performance on the target task.

Remark 3.3 (About the regularization parameter γ)**.** *We remark from the expression of $\boldsymbol{w}_\alpha$ in (α-FTC) that the weight decay γ is essential to have the dependence of $\boldsymbol{w}_\alpha$ on α. In fact, taking $\gamma \to 0$ leads to a fine-tuned classifier of the form:*

$$\boldsymbol{w}_\alpha = (\mathbf{X}\mathbf{X}^\top)^+\mathbf{X}\boldsymbol{y}$$

*where $(\mathbf{X}\mathbf{X}^\top)^+$ is the Moore-Penrose inverse of the symmetric semi-definite matrix $\mathbf{X}\mathbf{X}^\top$. Therefore, the obtained classifier does **not** depend on α here, nor on the pre-trained model $\tilde{\boldsymbol{w}}$. Additionally, having such a regularization technique is essential in transfer learning since the target dataset is generally much smaller than the pre-training one, and therefore the fine-tuning process can easily lead to overfitting in the absence of a regularization technique.*

3.2 RMT Background

To theoretically study the fine-tuned classifier $\boldsymbol{w}_\alpha$, we can leverage tools from Random Matrix Theory. In mathematical terms, the understanding of the asymptotic performance of the classifier $\boldsymbol{w}_\alpha$ boils down to the characterization of the statistical behavior of the *resolvent matrices* $\mathbf{Q}(z)$ and $\mathbf{R}(z)$ introduced in (9). In the following, we will recall some important notions and results from random matrix theory, which will be at the heart of our analysis. We start by defining the main object, which is the resolvent matrix.

Definition 3.4 (Resolvent)**.** *For a symmetric matrix $\mathbf{M} \in \mathbb{R}^{p\times p}$, the resolvent $\mathbf{Q}_M(z)$ of $\mathbf{M}$ is defined for $z \in \mathbb{C}\backslash\mathscr{S}(\mathbf{M})$ as:*

$$\mathbf{Q}_M(z) = (\mathbf{M} - z\mathbf{I}_p)^{-1},$$

where $\mathscr{S}(\mathbf{M})$ is the set of eigenvalues or spectrum of $\mathbf{M}$.

In fact, the study of the asymptotic performance of $\boldsymbol{w}_\alpha$ involves the estimation of linear forms of the resolvents $\mathbf{Q}$ and $\mathbf{R}$ in (9), such as $\frac{1}{n}\operatorname{Tr}\mathbf{Q}$ and $\boldsymbol{a}^\top\mathbf{Q}\boldsymbol{b}$ with $\boldsymbol{a},\boldsymbol{b}\in\mathbb{R}^p$ of bounded Euclidean norms. Therefore, the notion of a *deterministic equivalent* (Hachem et al., 2007) is crucial as it allows the design of a **deterministic** matrix, having (in probability or almost surely) asymptotically the same *scalar observations* as the random ones in the sense of *linear forms*. A rigorous definition is provided below.

Definition 3.5 (Deterministic equivalent (Hachem et al., 2007))**.** *We say that $\bar{\mathbf{Q}}\in\mathbb{R}^{p\times p}$ is a deterministic equivalent for the random resolvent matrix $\mathbf{Q}\in\mathbb{R}^{p\times p}$ if, for any bounded linear form $u:\mathbb{R}^{p\times p}\to\mathbb{R}$, we have that, as $p\to\infty$:*

$$u(\mathbf{Q})\xrightarrow{a.s.}u(\bar{\mathbf{Q}}),$$

where the convergence is in the almost sure sense.

In particular, a deterministic equivalent for the resolvents $\mathbf{Q}(z)$ and $\mathbf{R}(z)$ defined in (9) is given by the following Lemma.

Lemma 3.6 (Deterministic equivalent of $\mathbf{Q}$ and $\mathbf{R}$)**.** *Under the high-dimensional regime, when $p,n,N\to\infty$ with $\frac{p}{n}\to\eta\in(0,\infty)$ and $\frac{p}{N}\to\tilde{\eta}\in(0,\infty)$ and assuming $\|\boldsymbol{\mu}\|=\mathcal{O}(1)$, a deterministic equivalent for $\mathbf{Q}\equiv\mathbf{Q}(\gamma)$ and for $\mathbf{R}\equiv\mathbf{R}(\gamma)$, previously defined in (9), denoted $\bar{\mathbf{Q}}$ and $\bar{\mathbf{R}}$ respectively, are given by:*

$$\bar{\mathbf{Q}}(\gamma)=\left(\frac{\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top+\mathbf{I}_p}{1+\delta_Q}+\gamma\mathbf{I}_p\right)^{-1},\quad \bar{\mathbf{R}}(\gamma)=\left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top+\mathbf{I}_p}{1+\delta_R}+\gamma\mathbf{I}_p\right)^{-1}.$$

Where:

$$\delta_Q=\frac{1}{n}\operatorname{Tr}\bar{\mathbf{Q}}=\frac{\eta-\gamma-1+\sqrt{(\eta-\gamma-1)^2+4\eta\gamma}}{2\gamma},\quad \delta_R=\frac{\tilde{\eta}-\tilde{\gamma}-1+\sqrt{(\tilde{\eta}-\tilde{\gamma}-1)^2+4\tilde{\eta}\tilde{\gamma}}}{2\tilde{\gamma}}.$$

Proof. We will prove the deterministic equivalent of $\mathbf{Q}$, and the proof of $\bar{\mathbf{R}}$ can be derived similarly. In general, we want to find a deterministic equivalent $\bar{\mathbf{Q}}$ of the same form of $\mathbf{Q}$, i.e we consider $\bar{\mathbf{Q}}(\gamma)=(\mathbf{S}+\gamma\mathbf{I}_p)^{-1}$ and we want to find a deterministic matrix $\mathbf{S}\in\mathbb{R}^{p\times p}$ such that for any linear form u:

$$u(\mathbf{Q})\xrightarrow{\text{a.s.}}u(\bar{\mathbf{Q}}),$$

Or more simply:

$$u(\mathbb{E}[\mathbf{Q}]-\bar{\mathbf{Q}})\to 0.$$

We have that:

$$\begin{aligned}\mathbb{E}[\mathbf{Q}]-\bar{\mathbf{Q}}&=\mathbb{E}[\mathbf{Q}-\bar{\mathbf{Q}}]\\&=\mathbb{E}[\mathbf{Q}\left(\mathbf{S}-\frac{1}{n}\mathbf{X}\mathbf{X}^\top\right)\bar{\mathbf{Q}}]\\&=\mathbb{E}\left[\left(\mathbf{Q}\mathbf{S}-\frac{1}{n}\sum_{i=1}^n\mathbf{Q}\boldsymbol{x}_i\boldsymbol{x}_i^\top\right)\bar{\mathbf{Q}}\right]\end{aligned}$$

And since: $\mathbf{Q}\boldsymbol{x}_i = \frac{\mathbf{Q}_{-i}\boldsymbol{x}_i}{1+\delta_Q}$ and that we want $\mathbb{E}[\mathbf{Q}] = \bar{\mathbf{Q}}$ in linear forms, we get that:

$$\begin{aligned}
\mathbb{E}\left[\left(\mathbf{QS} - \frac{1}{n}\sum_{i=1}^{n}\mathbf{Q}\boldsymbol{x}_i\boldsymbol{x}_i^\top\right)\bar{\mathbf{Q}}\right] &= \bar{\mathbf{Q}}\mathbf{S}\bar{\mathbf{Q}} - \frac{1}{n}\sum_{i=1}^{n}\frac{1}{1+\delta_Q}\,\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top]\bar{\mathbf{Q}} \\
&= \bar{\mathbf{Q}}\mathbf{S}\bar{\mathbf{Q}} - \frac{1}{n}\sum_{i=1}^{n}\frac{1}{1+\delta_Q}\bar{\mathbf{Q}}(\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top + \mathbf{I}_p)\bar{\mathbf{Q}} \qquad (\boldsymbol{x}_i \perp\!\!\!\perp \mathbf{Q}_{-i}) \\
&= \bar{\mathbf{Q}}\left(\mathbf{S} - \frac{\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top + \mathbf{I}_p}{1+\delta_Q}\right)\bar{\mathbf{Q}}
\end{aligned}$$

Finally, it suffices to take: $\mathbf{S} = \frac{\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top + \mathbf{I}_p}{1+\delta_Q}$ to get the desired result. □

4 Main Theoretical Results

After having defined the setting and needed background, we will now present our main technical results, which describe the asymptotic behavior of the fine-tuned classifier defined in (α-FTC).

4.1 RMT Assumptions

We provide our results under the following growth rate assumptions (classical assumptions in Random Matrix Theory).

Assumption 4.1 (Growth Rates)**.** *Suppose that as* $p, n, N \to \infty$*:*

$$\textit{1)}\ \tfrac{p}{n} \to \eta \in [0,\infty), \qquad \textit{2)}\ \tfrac{p}{N} \to \tilde{\eta} \in [0,\infty), \qquad \textit{3)}\ \|\boldsymbol{\mu}\| = \mathscr{O}(1), \qquad \textit{4)}\ \|\boldsymbol{\mu}_\beta\| = \mathscr{O}(1).$$

The first and second assumptions simply state that our analysis considers both the low ($\eta, \tilde{\eta} \ll 1$) and high ($\eta, \tilde{\eta} \gg 1$) dimensional regimes. The third and last assumptions are also fundamental and state that the norm of the source $\boldsymbol{\mu}$ and target $\boldsymbol{\mu}_\beta$ data means do not scale with the dimension p, which makes the classification problem neither easy ($\|\boldsymbol{\mu}\| \to \infty$) nor impossible ($\|\boldsymbol{\mu}\| \to 0$) in high dimensions.

4.2 Theoretical performance with Ridge source classifier

Having stated the main assumptions, we are now in a position to present our main technical findings about the theoretical test performance of the fine-tuned classifier α-FTC. But beforehand, let us define some scalar quantities that will be useful in our derivations:

$$\lambda_Q = \|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q), \quad \lambda_R = \|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R), \quad h = 1 - \frac{\eta}{(1+\gamma(1+\delta_Q))^2},$$
$$\tilde{h} = 1 - \frac{\tilde{\eta}}{(1+\tilde{\gamma}(1+\delta_R))^2}$$

Our main theorem below describes the behavior of the decision function of our fine-tuned classifier.

Theorem 4.2 (Gaussianity of the fine-tuned Ridge model)**.** *Let $\boldsymbol{w}_\alpha$ be the fine-tuned classifier as defined in (α-FTC) and suppose that Assumption 4.1 holds. The decision function $\boldsymbol{w}_\alpha^\top \boldsymbol{x}$, on some test sample $\boldsymbol{x} \in \mathscr{C}_a$ independent of $\mathbf{X}$, satisfies:*

$$\boldsymbol{w}_\alpha^\top \boldsymbol{x} \xrightarrow{\mathscr{D}} \mathcal{N}\left((-1)^a m_\alpha,\, \nu_\alpha - m_\alpha^2\right),$$

where:

$$m_\alpha = \frac{1}{\lambda_Q}\left(\|\boldsymbol{\mu}_\beta\|^2 + \frac{\alpha\beta\gamma(1+\delta_Q)}{\lambda_R}\|\boldsymbol{\mu}\|^2\right),$$
$$\nu_\alpha = T_1 + \alpha T_2 + \alpha^2 T_3.$$

With:

$$T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q} - 2(1-h)\right) + \frac{1-h}{h},$$
$$T_2 = \frac{2\gamma\beta(1+\delta_Q)\|\boldsymbol{\mu}\|^2}{\lambda_R\lambda_Q}\left(1 - \frac{\gamma(1+\delta_Q)}{h\lambda_Q}\right),$$
$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\times$$
$$\left[\frac{\|\boldsymbol{\mu}\|^2}{\lambda_R^2}\left(\frac{\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q^2} + \frac{1-h}{\eta}\left(1 + \frac{\beta^2\|\boldsymbol{\mu}\|^2\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q^2} - \frac{2\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q} + (1-\tilde{h})\left(1 - \frac{2\|\boldsymbol{\mu}\|^2}{\lambda_R}\right)\right)\right)\right]$$

In simple terms, Theorem 4.2 states that the decision function of the classifier (α-FTC) is asymptotically equivalent to the thresholding of two monovariate Gaussian random variables with respective means m_α and $-m_\alpha$ and standard deviation $\nu_\alpha - m_\alpha^2$, where the statistics m_α and ν_α are expressed in terms of the scalar quantities defined above. This behavior is highlighted in Figure 1 which depicts the histogram of the decision function for the different values of α and β along with the theoretical Gaussian distributions as per Theorem 4.2. The gaussian distribution comes from Lyupanov's Central Limit theorem, so we only needed to compute the first and second order moments of the decision function $\boldsymbol{w}_\alpha^\top \boldsymbol{x}$ (for a test sample $\boldsymbol{x}$) to prove the above theorem, which was presented in details in Appendix B.

Having characterized the distribution of the decision function of $\boldsymbol{w}_\alpha$, we can now estimate its generalization performance, such as its test accuracy. In fact, the theoretical test missclassification of $\boldsymbol{w}_\alpha$ is equal to the shaded area between the two histograms (intersection) in Figure 1, and since the histograms are of Gaussian laws, we have the exact formula to compute this desired quantity which we state in the following proposition:

Proposition 4.3 (Asymptotic test accuracy of $\boldsymbol{w}_\alpha$)**.** *The asymptotic test accuracy of $\boldsymbol{w}_\alpha$ defined in (α-FTC), under Assumptions 4.1 as the number of test samples $n_{test} \to \infty$, is given by:*

$$\mathscr{A}_{test} \xrightarrow{a.s.} 1 - \varphi\left((\nu_\alpha - m_\alpha^2)^{-\frac{1}{2}} m_\alpha\right), \quad \textit{where: } \varphi(x) = \frac{1}{\sqrt{2\pi}}\int_x^{+\infty} e^{-\frac{t^2}{2}}\,\mathrm{dt}.$$

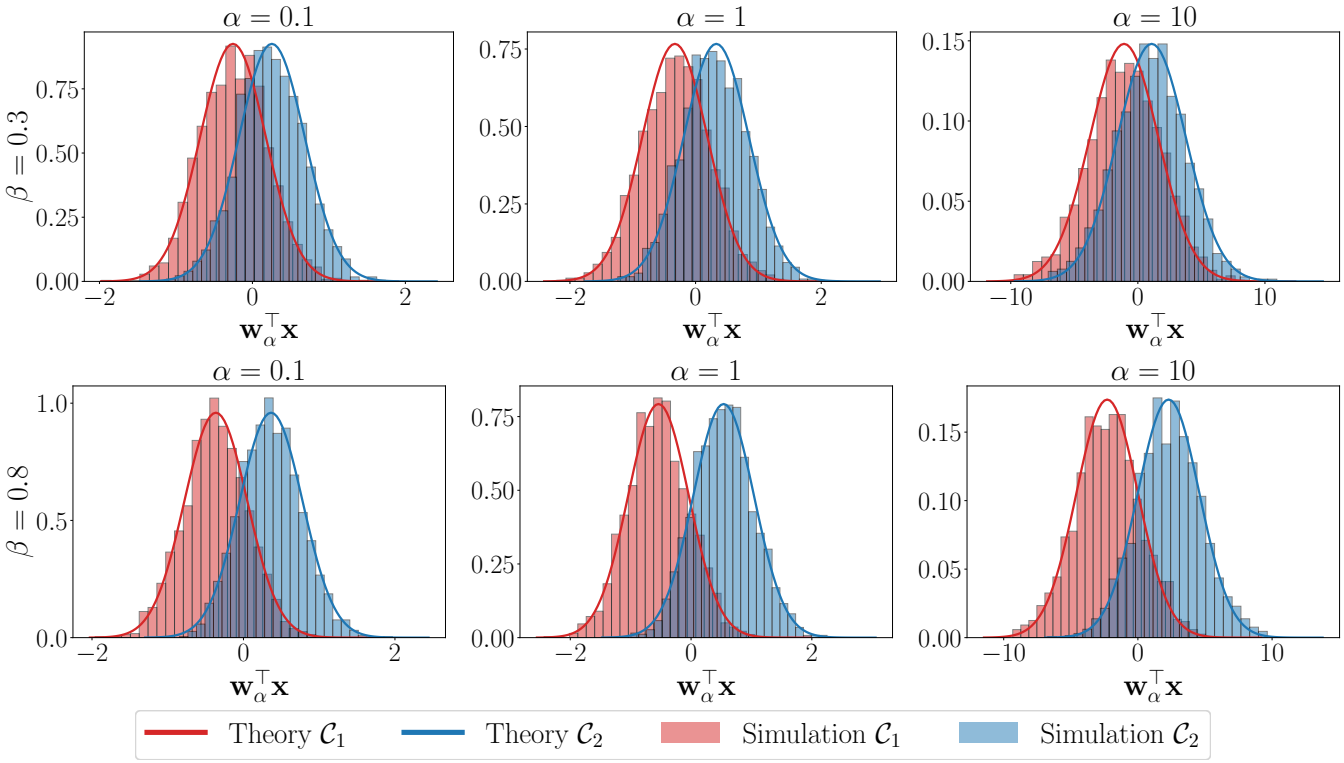

Figure 1: Distribution of the decision function $\boldsymbol{w}_\alpha^\top \boldsymbol{x}$ for different values of α (per column) and β (per row) for a data model given by: $\boldsymbol{\mu}_\beta = \beta\boldsymbol{\mu} + \sqrt{1-\beta^2}\boldsymbol{\mu}^\perp$. Here we have $N = 5000$, $n = 200$, $p = 400$, $\|\boldsymbol{\mu}\| = 1.5$, $\|\boldsymbol{\mu}^\perp\| = 1$, $\gamma = \tilde{\gamma} = 1$. The theoretical Gaussian distributions are predicted as per Theorem 4.2.

Therefore, thanks to Proposition 4.3, we now have the exact formula of the theoretical **test accuracy** of our classifier $\boldsymbol{w}_\alpha$, which can be used to simulate the dynamics of the test accuracy with respect to the parameters of the setting (like α, β and γ, as it was done in Figure 2), and also to characterize the expression of the optimal and worst parameters of the model (for instance, the α parameter) to use for the fine-tuning process. In particular, we will derive the theoretical expressions of the extrema of α that lead to the best and the worst test accuracies on the target task (proof in Appendix B).

Theorem 4.4 (Optimal α)**.** *Maximizing the term* $\left((\nu_\alpha - m_\alpha^2)^{-\frac{1}{2}} m_\alpha\right)$ *in terms of* α *leads to optimum test accuracy* $\mathscr{A}_{test}$*, and gives a unique maximizer* $\alpha^\star$ *given by:*

$$\alpha^\star = \frac{\lambda_R T_2 \|\boldsymbol{\mu}_\beta\|^2 - 2\beta\gamma T_1(1+\delta_Q)\|\boldsymbol{\mu}\|^2}{\beta\gamma T_2(1+\delta_Q)\|\boldsymbol{\mu}\|^2 - 2\lambda_R T_3 \|\boldsymbol{\mu}_\beta\|^2}$$

Plus, solving $(\nu_\alpha - m_\alpha^2)^{-\frac{1}{2}} m_\alpha = 0$ *leads to the unique minimizer* $\bar{\alpha}$ *of* $\mathscr{A}_{test}$*, which is given by:*

$$\bar{\alpha} = -\frac{\lambda_R \|\boldsymbol{\mu}_\beta\|^2}{\beta\gamma(1+\delta_Q)\|\boldsymbol{\mu}\|^2}$$

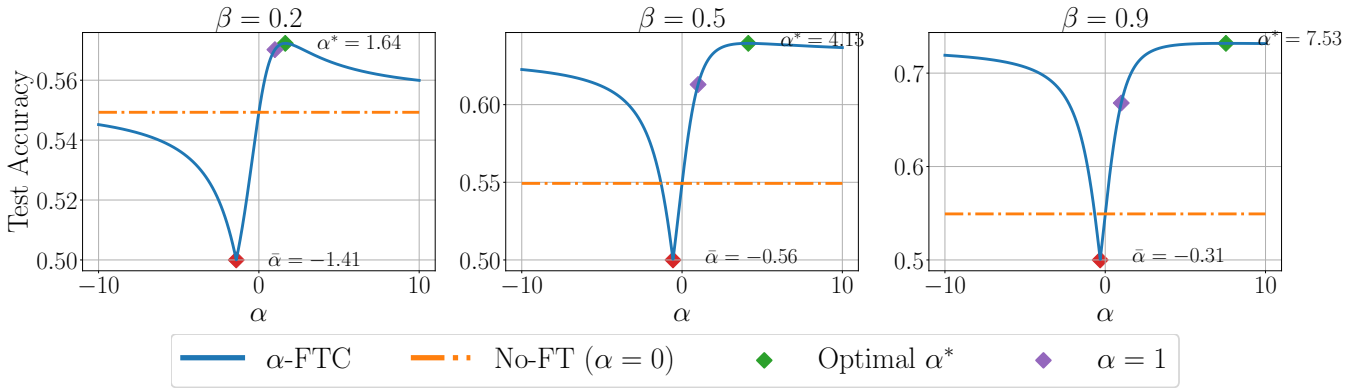

Figure 2: Theoretical Test Accuracy variation with α for $N = 5000$, $n = 40$, $p = 1000$, and the theoretical model is modified to take β in $(0, 1)$: $\boldsymbol{\mu}_\beta = \beta\boldsymbol{\mu} + \sqrt{1-\beta^2}\boldsymbol{\mu}^\perp$, where $\|\boldsymbol{\mu}\| = \|\boldsymbol{\mu}^\perp\| = 0.8$. Finally the regularization parameters are: $\tilde{\gamma} = 2$ and $\gamma = 10^{-1}$.

Theorem 4.4 gives us the exact expression of the optimal scaling parameter α^*, which we can examine its dynamics with respect to the other parameters of the considered model. For instance, Figure 3 clearly depicts the non-trivial contribution of the dimension p to the choice of α. It is clear that $\alpha^\star$ is non-decreasing with the alignment β between the source and target tasks, but its effect gets amplified with the dimension p of the problem. Notably, the influence of α is more pronounced in low-resource settings ($p \gg n$) compared to cases where sufficient fine-tuning data is available. This further underscores the crucial role of α in effectively leveraging the pre-trained model and source data. Additionally, as $\beta \to 0$, we also remark that $\alpha^\star \to 0$, which means that fine-tuning has no added value when the source and target tasks are **unrelated** and orthogonal.

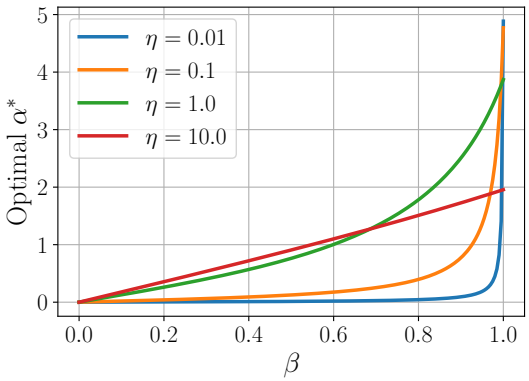

Figure 3: Variations of the optimal parameter $\alpha^\star$ with respect to the alignment between the source $\boldsymbol{\mu}$ and target $\boldsymbol{\mu}_\beta$ dataset means. These latter were chosen of norm 1, $N = 2000$, $n = 200$ and $\gamma = \tilde{\gamma} = 1$.

Additionally, Figure 2 shows the evolution of the theoretical test accuracy with the parameter α for different source datasets (i.e, different alignments β). In particular, we observe the existence of an optimal parameter $\alpha^\star$ that is generally different from 1 (standard approach), and as can be previously anticipated, its impact on the test accuracy is more visible in the case of a higher alignment factor β, which means in this case that we highly leverage the base model to better generalize on the new task (see Remark 3.2).

4.3 Theoretical performance with arbitrary source classifier

We can also extend our previous results in the case of an arbitrary source classifier $\tilde{\boldsymbol{w}}$ (instead of Ridge), which gives us an analysis of our method in the case where we don't have access to the pre-training data. In fact, we assume here that we don't know the distribution of the source data, and thus the alignment term

β will be defined differently, and in this case, it is given by the dot product between the source classifier weights $\tilde{\boldsymbol{w}}$ and the target data mean $\boldsymbol{\mu}_\beta$, i.e $\beta = \langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle$. We will show in the following that we get the same observations and insights as of the previous section.

Under the same assumptions 4.1, we state our main theorem of this section which also describes the asymptotic distribution of the fine-tuned classifier in this setting.

Theorem 4.5 (Gaussianity of the fine-tuned model for an arbitrary $\tilde{\boldsymbol{w}}$)**.** *Let $\boldsymbol{w}_\alpha$ be the fine-tuned classifier as defined in (α-FTC) and suppose that Assumption 4.1 holds. The decision function $\boldsymbol{w}_\alpha^\top \boldsymbol{x}$, on some test sample $\boldsymbol{x} \in \mathscr{C}_a$ independent of $\mathbf{X}$, satisfies:*

$$\boldsymbol{w}_\alpha^\top \boldsymbol{x} \xrightarrow{\mathscr{D}} \mathcal{N}\left((-1)^a m_\alpha,\, \nu_\alpha - m_\alpha^2\right),$$

where:

$$m_\alpha = \frac{\|\boldsymbol{\mu}_\beta\|^2 + \alpha\gamma(1+\delta_Q)\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)},$$
$$\nu_\alpha = T_1 + \alpha T_2 + \alpha^2 T_3.$$

with:

$$T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q} - 2(1-h)\right) + \frac{1-h}{h},$$
$$T_2 = \frac{2\gamma(1+\delta_Q)\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda} - (1-h)\right),$$
$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\left(\frac{\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle^2}{\lambda_Q^2} + \frac{1-h}{\eta}\|\tilde{\boldsymbol{w}}\|^2 + \frac{(1-h)\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle^2}{\eta\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q} - 2\right)\right).$$

The new **alignment** term β quantifies how well is the source classifier performing on the target domain, in the sense that larger values of $|\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle|$ essentially mean that $\tilde{\boldsymbol{w}}$ can better classifies the target data, which justifies the choice of this metric as an alignment measurement. By examining the contribution of this term in the expression of m_α and ν_α in Theorem 4.5 and by looking at the dynamics of α^* (which will be defined later) with respect to it as in Figure 4, we observe that this new alignment term operates in the same way as the previous β defined in the case of a known source data distribution.
Again, thanks to Proposition 4.3, we have the theoretical formula of the test accuracy of this new fine-tuned classifier $\boldsymbol{w}_\alpha$, which we can use to compute the optimal parameter α^* that maximizes the test accuracy, which is the object of the following theorem 4.6.

Theorem 4.6 (Optimal α for arbitrary $\tilde{\boldsymbol{w}}$)**.** *Maximizing the term $\left((\nu_\alpha - m_\alpha^2)^{-\frac{1}{2}} m_\alpha\right)$ with respect to α leads to optimal test accuracy $\mathscr{A}_{test}$ and gives a unique maximizer α^* given by:*

$$\alpha^* = \frac{\eta(1+\gamma(1+\delta_Q))\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle}{\gamma(1+\delta_Q)\left(\lambda\|\boldsymbol{\mu}_\beta\|^2\|\tilde{\boldsymbol{w}}\|^2 - (\lambda-\eta)\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle^2\right)}$$

Solving $m_\alpha = 0$ leads to the unique minimizer $\bar{\alpha}$ of $\mathscr{A}_{test}$ which is given by:

$$\bar{\alpha} = \frac{-\|\boldsymbol{\mu}_\beta\|^2}{\gamma(1+\delta_Q)\langle \tilde{\boldsymbol{w}}, \boldsymbol{\mu}_\beta \rangle}$$

Similarly, for the arbitrary classifier $\tilde{\boldsymbol{w}}$, we observe the same scaling behavior of the optimal parameter α^* with respect to the alignment term. Notably, the influence of α is more pronounced in low-resource settings ($p \gg n$) compared to cases where sufficient fine-tuning data is available. This further underscores the crucial role of α in effectively leveraging the pre-trained model and source data.
Therefore, we have shown in this whole section that adding a scaling parameter to the base model helps better leverage the source task in Transfer Learning, though needs to be carefully chosen.

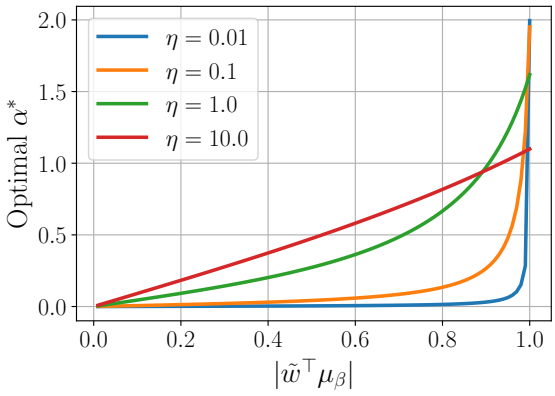

Figure 4: Variations of the optimal parameter α^* with respect to the alignment between the source classifier $\tilde{\boldsymbol{w}}$ and the target dataset mean $\boldsymbol{\mu}_\beta$. These latter were chosen of norm 1, $n = 200$ and $\gamma = 1$.

4.4 Conclusion of our theory

The theoretical framework developed in this section provides precise predictions on how the scaling parameter α influences generalization in transfer learning. In particular, the Random Matrix Theory analysis reveals a non-trivial optimal α^* that depends on alignment between source and target tasks and the dimensional regime. To verify whether these analytical insights hold in practice, we now turn to empirical validation. In the next section, we test our predictions on both controlled linear models and complex real-world fine-tuning scenarios such as Large Language Models (LLMs).

5 Experiments

In this section, we present some experiments on real datasets to validate our approach and prove the effectiveness of scaling the base model. We start by fine-tuning linear models on the Amazon Review dataset (Blitzer et al., 2007) to verify our theoretical findings. After that, we formalize our new class of reparameterization methods and verify its efficiency by experiments on fine-tuning LLMs on the GLUE benchmark (Wang et al., 2018).

5.1 Within our theoretical model: Linear Binary Classification

Here we present our experiments on the Amazon Review dataset (Blitzer et al., 2007) to validate our theory. This dataset includes several binary classification tasks corresponding to positive versus negative reviews of `books`, `dvd`, `electronics`, and `kitchen`. We apply the standard scaler from `scikit-learn` (Pedregosa et al., 2011) and estimate $\|\boldsymbol{\mu}\|$, $\|\boldsymbol{\mu}^\perp\|$ and β with the normalized data. Figure 5 depicts the variation in test accuracy of three transfer tasks with respect to the parameter α and gives a comparison between the three main schemes: $\alpha = 0$ (i.e., learning directly on the target data without using previous source knowledge), $\alpha = 1$ (classical approach) and with the optimal $\alpha^\star$ obtained using the theoretical formula in Theorem 4.4. Depending on the tasks, we see a clear improvement in the test accuracy for $\alpha^\star$ compared to the other

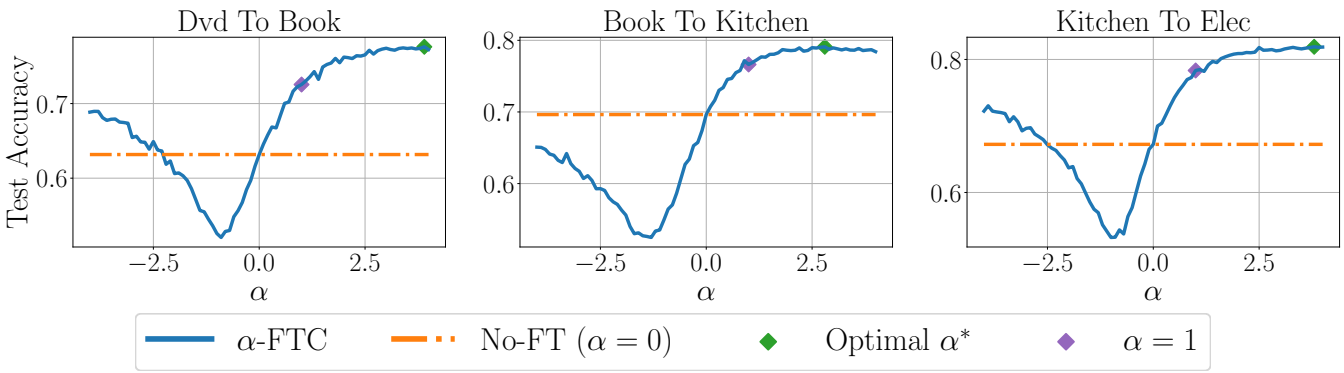

Figure 5: Test accuracy variation with α for different transfer learning schemes from the Amazon Review dataset (Blitzer et al., 2007). The considered parameters here are: $N = 2000$, $n = 40$, $p = 400$, $\gamma = 10^{-1}$ and $\tilde{\gamma} = 2$.

schemes, which further highlights the impact of this scaling parameter. Table 1 summarizes the results obtained for all the possible transfer tasks between the sub-datasets.

Table 1: Test accuracy (in %) comparison over Amazon review datasets (Blitzer et al., 2007) for $N = 2000$, $n = 40$, $p = 400$, and optimal regularization parameters $\gamma = \tilde{\gamma} = 1$. As theoretically anticipated, our new fine-tuning approach yields better classification accuracy than training directly on the target dataset ($\alpha = 0$) or using $\alpha = 1$. The results were computed for 3 random seeds.

Source Dataset	**Target Dataset**	$\boldsymbol{\alpha = 0}$	$\boldsymbol{\alpha = 1}$	**Optimal** $\alpha^\star$
`Books`	`Dvd` ($\beta = 0.8$)	64.12 ± 0.03	75.67 ± 0.24	77.35 ± 0.14 ($\alpha^\star = 2.47$)
	`Electronics` ($\beta = 0.71$)	68.61 ± 0.74	76.65 ± 0.02	77.12 ± 0.17 ($\alpha^\star = 1.68$)
	`Kitchen` ($\beta = 0.79$)	69.24 ± 0.95	78.19 ± 0.05	78.96 ± 0.26 ($\alpha^\star = 1.9$)
`Dvd`	`Books` ($\beta = 0.78$)	63.43 ± 0.67	75.22 ± 0.24	77.59 ± 0.07 ($\alpha^\star = 2.47$)
	`Electronics` ($\beta = 0.71$)	68.61 ± 0.74	76.72 ± 0.17	76.88 ± 0.42 ($\alpha^\star = 1.69$)
	`Kitchen` ($\beta = 0.78$)	69.24 ± 0.95	78.11 ± 0.23	78.72 ± 0.54 ($\alpha^\star = 1.88$)
`Electronics`	`Books` ($\beta = 0.51$)	63.43 ± 0.67	72.2 ± 0.1	73.29 ± 0.13 ($\alpha^\star = 1.67$)
	`Dvd` ($\beta = 0.52$)	64.12 ± 0.03	72.41 ± 0.16	73.48 ± 0.17 ($\alpha^\star = 1.69$)
	`Kitchen` ($\beta = 0.9$)	69.24 ± 0.95	81.58 ± 0.15	83.02 ± 0.1 ($\alpha^\star = 2.29$)
`Kitchen`	`Books` ($\beta = 0.52$)	63.43 ± 0.67	72.86 ± 0.1	74.27 ± 0.14 ($\alpha^\star = 1.84$)
	`Dvd` ($\beta = 0.53$)	64.12 ± 0.03	73.15 ± 0.08	74.15 ± 0.09 ($\alpha^\star = 1.82$)
	`Electronics` ($\beta = 0.83$)	68.61 ± 0.74	80.14 ± 0.02	81.89 ± 0.18 ($\alpha^\star = 2.31$)

We note that our approach yields optimal results for all transfer tasks, which clearly validates our theoretical results and underscores the efficiency of our method in terms of its generalization capabilities. This can also be observed in Figure 5, which shows that the optimal test accuracy is obtained for a parameter α that is not necessarily equal to 1.

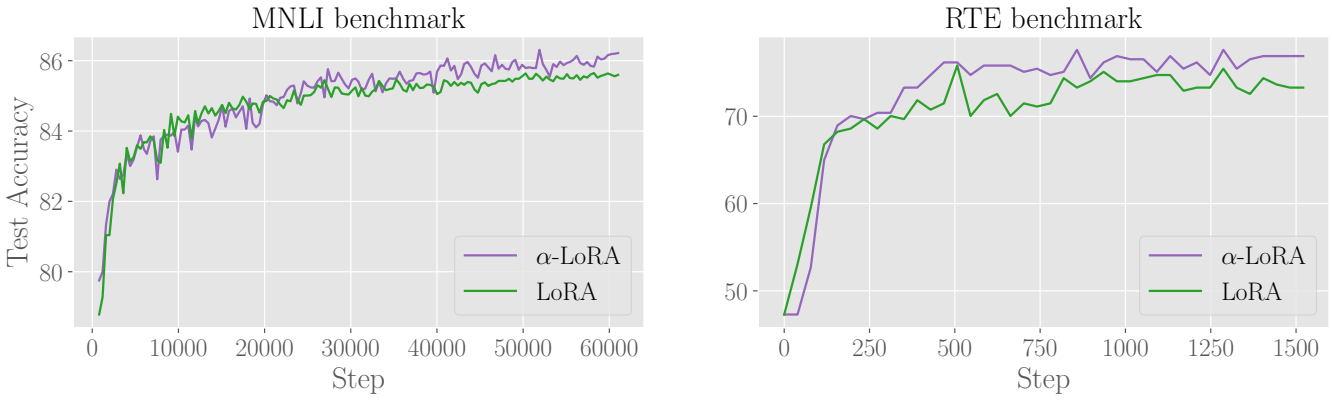

Figure 6: Test accuracy evolution of `roberta-base` finetuned on MNLI and RTE for a single fixed seed (seed 5 for MNLI and seed 123 for RTE).

5.2 Beyond our theoretical model: Supervised Fine-tuning for LLMs

To go beyond linear models, we now fine-tune language models, specifically the `roberta-base` BERT model (Liu et al., 2019)), on downstream classification tasks taken from GLUE tasks (Wang et al., 2018). To adapt our theoretical insights from the linear model to complex multi-layered architectures like LLMs, we generalize the scalar scaling parameter α to a vector $\boldsymbol{\alpha}$, i.e applying a scaling parameter to each output dimension of the target module. This extension provides finer-grained control, allowing the model to rescale the contribution of the frozen base weights on a per-output-neuron basis. This added flexibility is crucial for capturing the intricate functional specialization within different dimensions of a neural network's hidden states. Consequently, the update rule for a weight matrix $\mathbf{W}^\star$ is modified from a simple scalar product to a row-wise scaling operation, as detailed below:

$$\boxed{\mathbf{W}_{\text{new}} = \boldsymbol{\alpha} \odot \mathbf{W}^\star + \mathbf{W}} \tag{10}$$

where $\odot$ is the element-wise product between vectors, $\mathbf{W}^\star \in \mathbb{R}^{d_{out} \times d_{in}}$ is the original layer weights (frozen during training), $\boldsymbol{\alpha} \in \mathbb{R}^{d_{out}}$ (each element in the output dimension is then multiplied by a scalar), and $\mathbf{W} \in \mathbb{R}^{d_{out} \times d_{in}}$ is the trainable weight matrix. This generalization from a scalar α to a vector is adequate with our theoretical study in the previous section since the weight matrix $\mathbf{W}^*$ is comprised of d_{out} vectors (rows), and thus a *scalar* α is applied to each row. Additionally, to further make our fine-tuning method lightweight, $\mathbf{W}$ can be approximated with a low-rank matrix: $\mathbf{W} = \mathbf{AB}$, where: $\mathbf{A} \in \mathbb{R}^{d_{out} \times r}$ and $\mathbf{B} \in \mathbb{R}^{r \times d_{in}}$, a method that we call α-LoRA. We then report in Table 2 the test performance obtained using standard LoRA and our α-LoRA method evaluated on six GLUE tasks: MNLI, QNLI, MRPC, RTE, SST-2, and QQP.

We note that from Table 2 and Figure 6, our method leads to higher generalization performance compared to standard LoRA across all GLUE benchmarks, which further validates our theoretical findings of the previous section, and proves the usefulness of scaling the base model weights.

Table 2: Test accuracy comparison over GLUE classification tasks (Wang et al., 2018) using `roberta-base` model. As theoretically anticipated, our new fine-tuning approach yields better test classification accuracy than the standard LoRA method ($\boldsymbol{\alpha} = 1$). Details about these experiments are presented in Appendix F.

Method	MNLI	QNLI	MRPC	RTE	SST-2	QQP
LoRA	85.77 ± 0.16	91.95 ± 0.03	88.40 ± 0.31	74.01 ± 1.64	94.00 ± 0.11	88.80 ± 0.02
α-LoRA	**86.12 ± 0.06**	**92.20 ± 0.13**	**89.46 ± 0.53**	**77.62 ± 0.59**	**94.38 ± 0.01**	**88.86 ± 0.03**

Finding the parameters $\boldsymbol{\alpha}$. We designed a practical heuristic algorithm to automatically update $\boldsymbol{\alpha}$ during training. In fact, we consider each vector $\boldsymbol{\alpha}$ as a trainable parameter and update these vectors once every T step (tunable hyperparameter) by sampling a new batch, different from the one used to train the reparametrization weights $\mathbf{W}$, and then taking a gradient step over this new batch with either `Adam` or `AdamW`. The full pseudo-code of our algorithm is given by the following.

Algorithm 1 α-LoRA Fine-Tuning

Require: Base model weights $\{\mathbf{W}_i^*\}_{i=1}^N$, fine-tuning dataset $\mathscr{D} = \{B_j\}_{j=1}^b$ divided into batches, update period T, optimizers `optim` (for LoRA modules) and `optim_alpha` (for $\boldsymbol{\alpha} = \{\alpha_i\}_{i=1}^N$), number of epochs n.

for $k = 1 \dots n$ **do**
  for batch B_j in $\mathscr{D}$ **do**
    Update LoRA modules $\{(A_i, B_i)\}_{i=1}^N$ with a gradient step on B using `optim`.
    if $j \bmod T = 0$ **then**
      Sample a fresh batch B_α from $\mathscr{D}$
      Update $\boldsymbol{\alpha}$ with a gradient step on B_α using `optim_alpha`.
    end if
  end for
end for

The design choices of our algorithm can be justified by the following:

- Because the vectors $\boldsymbol{\alpha}$ applied to each module lie in the whole Euclidean space $\mathbb{R}^d$, it is not possible to find such a parameter through a simple grid search, as this will give a very costly and impractical algorithm.
- Additionally, finding theoretical formulas for each vector $\boldsymbol{\alpha}$ is very hard, if not impossible. Therefore, it is crucial to have an algorithm that updates the vectors $\boldsymbol{\alpha}$ automatically.
- Finally, because we want to optimize the **generalization** performance of our fine-tuning method, training $\boldsymbol{\alpha}$ in the same way as the reparametrization weights $\mathbf{W}$ can easily lead to overfitting of the model, which justifies sampling of new batches to update $\boldsymbol{\alpha}$ and the update rate T. Our specific choices are detailed for reproducibility in Appendix F.

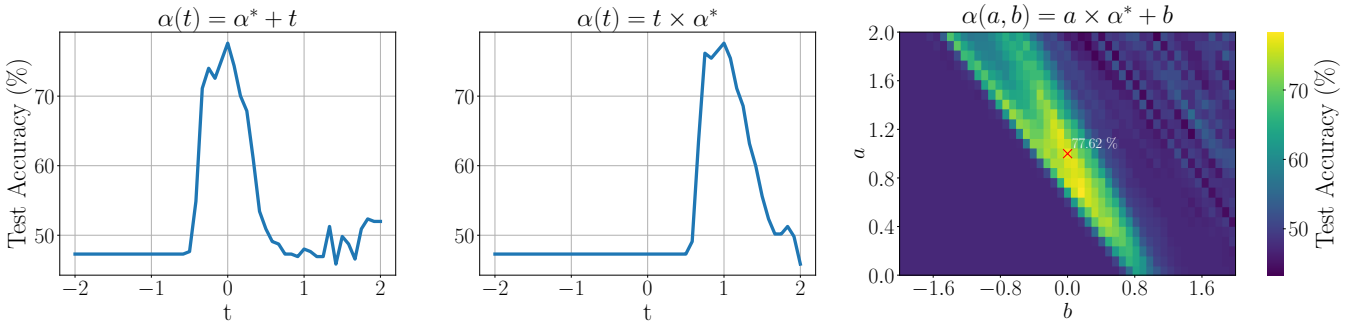

Figure 7: Test accuracy of `roberta-base` finetuned on RTE for different values of $\boldsymbol{\alpha}$ in the neighborhood of the obtained $\boldsymbol{\alpha}^\star$. The values of the parameters $\boldsymbol{\alpha}^\star$ in this experiment range between 0.85 and 1.14.

Figure 7 shows that our algorithm leads to optimal scaling vectors $\boldsymbol{\alpha}^\star$ in their neighborhood, which proves the effectiveness of our algorithm and the fine-tuning method in general. The pseudo-code 1 of our algorithm is detailed in the Appendix F.

Overhead induced by the additional parameters $\boldsymbol{\alpha}$. We note that the number of additional trainable parameters $\boldsymbol{\alpha}$ induced by our algorithm 1 is negligible compared to the standard approach (fixed $\boldsymbol{\alpha} = 1$), for example in the case of our experiments with `roberta-base` model, the increase in the number of trainable parameters is only of 0.02%. Additionally, investigating the resulting values of these learned $\boldsymbol{\alpha}$ vectors as reported in Figures 8, 10 and 11, we notice that we get similar values for query and value matrices, thus we can use a shared parameter for both weight matrices (or for the whole attention module more generally), reducing the overhead even further.

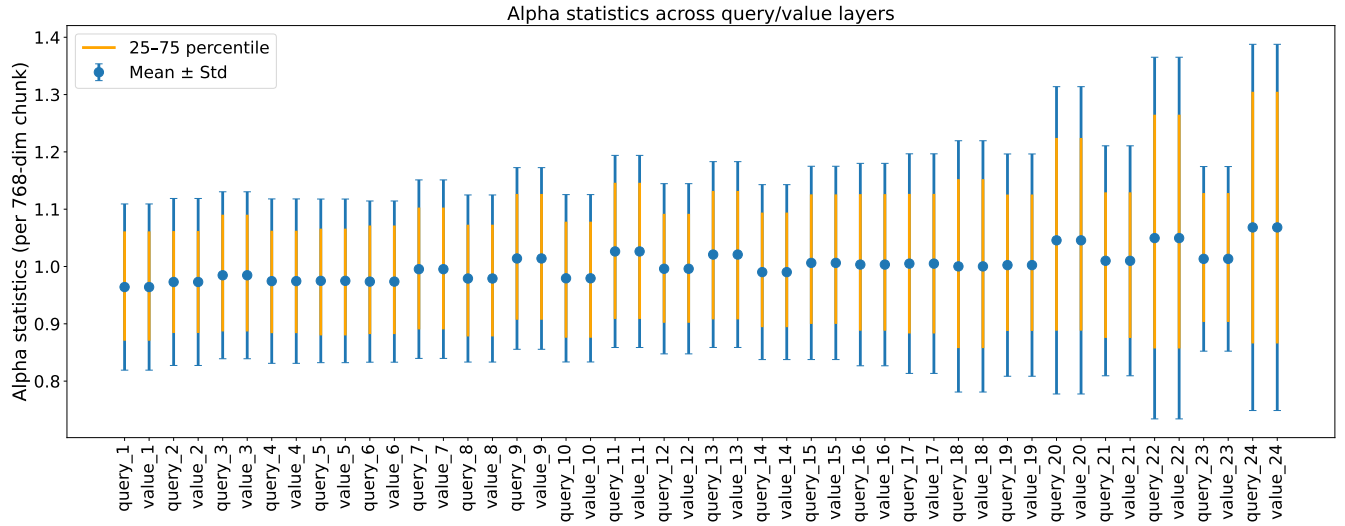

Figure 8: Statistics of the vectors $\boldsymbol{\alpha}$ for the MNLI benchmark

6 Multi-source Transfer Learning

Building on the single-source formulation, we extend our theoretical framework to multi-source transfer settings, where multiple pre-trained models contribute to a single target task. This setting is increasingly relevant with the rise of mixture-of-experts and multi-domain pretraining.

6.1 Asymptotic distribution of the test

In this section, we present an extension of our theoretical work of section 4 to the case of Transfer Learning using multiple source classifiers. Given T source classifiers $\{\boldsymbol{w}_t\}_{t=1}^T$ and a single target task, the goal is to fine-tune a mixture of these classifiers on the target task. Specifically, we want to find the optimal fine-tuned classifier $\boldsymbol{w}_\Omega$ that is written as:

$$\boldsymbol{w}_\Omega = \sum_{t=1}^T \alpha_t \boldsymbol{w}_t + \boldsymbol{a}$$

where $\alpha_t \in \mathbb{R}$ and $\boldsymbol{a}$ is an adapter trained on the target dataset as follows:

$$\boldsymbol{a} = \arg\min_{\boldsymbol{v}} \frac{1}{n}\|\mathbf{X}^\top(\sum_{t=1}^T \alpha_t \boldsymbol{w}_t + \boldsymbol{v}) - \boldsymbol{y}\|^2 + \gamma\|\boldsymbol{v}\|^2$$

Then, $\boldsymbol{a}$ expresses as:

$$\boldsymbol{a} = \frac{1}{n}\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + \gamma\mathbf{I}_p\right)^{-1}\left(\mathbf{X}\boldsymbol{y} - \mathbf{X}\mathbf{X}^\top\sum_{t=1}^T \alpha_t \boldsymbol{w}_t\right)$$

Thus, our new fine-tuned classifier writes as:

$$\boldsymbol{w}_\Omega = \sum_{t=1}^T \alpha_t \boldsymbol{w}_t + \boldsymbol{a} = \frac{1}{n}\mathbf{Q}\mathbf{X}\boldsymbol{y} + \gamma\sum_{t=1}^T \alpha_t \mathbf{Q}\boldsymbol{w}_t \tag{11}$$

Using the same RMT tools, we establish the asymptotic distribution of the decision function of the classifier $\boldsymbol{m}_\Omega$ for an arbitrary test sample $\boldsymbol{x} \sim \mathscr{N}((-1)^a\boldsymbol{\mu}_\beta, \mathbf{I}_p)$ in the following theorem.

Theorem 6.1 (Test performance for multi-source transfer learning)**.** *Let $\boldsymbol{w}_\Omega$ be the multi-source fine-tuned classifier as defined in (D) and suppose Assumption 4.1 hold. The decision function $\boldsymbol{w}_\Omega^\top\boldsymbol{x}$ on some test sample $\boldsymbol{x} \in \mathscr{C}_a$ independent of $\mathbf{X}$ for arbitrary source classifiers $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_T$, satisfies:*

$$\boldsymbol{w}_\Omega^\top\boldsymbol{x} \xrightarrow{\mathscr{D}} \mathscr{N}\left((-1)^a m_\Omega,\, \nu_\Omega - m_\Omega^2\right),$$

where:

$$m_\Omega = \frac{\|\boldsymbol{\mu}_\beta\|^2 + \gamma(1+\delta_Q)\sum_{t=1}^T \alpha_t\langle\boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)},$$
$$\nu_\Omega = T_1 + T_2 + T_3.$$

with:

$$T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\lambda_Q} - 2(1-h)\right) + \frac{1-h}{h}$$

$$T_2 = \frac{2\gamma(1+\delta_Q)}{h\lambda_Q}\sum_{t=1}^{T}\alpha_t\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q}-(1-h)\right)\langle \boldsymbol{w}_t,\boldsymbol{\mu}_\beta\rangle$$

$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\times$$

$$\sum_{t,k=1}^{T}\alpha_t\alpha_k\left[\frac{\langle \boldsymbol{w}_t,\boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k,\boldsymbol{\mu}_\beta\rangle}{\lambda_Q^2}+\frac{1}{(1+\gamma(1+\delta_Q))^2}\left(\langle \boldsymbol{w}_t,\boldsymbol{w}_k\rangle+\frac{\langle \boldsymbol{w}_t,\boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k,\boldsymbol{\mu}_\beta\rangle}{\lambda_Q}(\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q}-2)\right)\right]$$

Theorem 6.1 gives the distribution of the fine-tuned classifier obtained through a mixture of many source models. This latter will be very useful to characterize the optimal scaling factors that can be used to maximize the transfer generalization capabilities.

6.2 Characterization of the optimal scaling factors

We now use Proposition 4.3 to characterize the extrema scaling factors $\{\alpha_t\}_{t=1}^T$, but beforehand, let us write the first and second order moments m_Ω and ν_Ω in vectorized forms in order for a better readability of the results.

Denote by $\boldsymbol{\alpha}=(\alpha_t)_{t=1}^T\in\mathbb{R}^T$ and the matrix of source classifiers: $\mathbf{W}=(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_T)\in\mathbb{R}^{p\times T}$. Then, the quantities m_Ω, T_2 and T_3 write as:

$$m_\Omega = \frac{\left(\|\boldsymbol{\mu}_\beta\|^2+\gamma(1+\delta_Q)\boldsymbol{\alpha}^\top\mathbf{W}^\top\boldsymbol{\mu}_\beta\right)}{\|\boldsymbol{\mu}_\beta\|^2+1+\gamma(1+\delta_Q)} \tag{12}$$

$$T_2 = \frac{2\gamma(1+\delta_Q)}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q}-(1-h)\right)\boldsymbol{\alpha}^\top\mathbf{W}^\top\boldsymbol{\mu}_\beta \tag{13}$$

$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\boldsymbol{\alpha}^\top\mathbf{M}\boldsymbol{\alpha} \tag{14}$$

Worst scaling factors. Finding the worst scaling factors (that lead to a test accuracy of 50%) boils down to solving the equation $m_\Omega=0$, which in turn leads to the following condition:

$$\bar{\boldsymbol{\alpha}}^\top\mathbf{W}^\top\boldsymbol{\mu}_\beta = -\frac{\|\boldsymbol{\mu}_\beta\|^2}{\gamma(1+\delta_Q)} \tag{15}$$

which also means that:

$$\sum_{t=1}^{T}\bar{\alpha}_t\boldsymbol{w}_t+\frac{\boldsymbol{\mu}_\beta}{\gamma(1+\delta_Q)}\perp\boldsymbol{\mu}_\beta$$

Best scaling factors. The theoretical test accuracy writes as follows:

$$\mathscr{A}_{\text{test}}(\boldsymbol{\alpha})=\varphi\left(\frac{a_1+\boldsymbol{\alpha}^\top\boldsymbol{v}_1}{\sqrt{a_2+\boldsymbol{\alpha}^\top\boldsymbol{v}_2+\boldsymbol{\alpha}^\top\tilde{\mathbf{M}}\boldsymbol{\alpha}}}\right)$$

where:

$$a_1=\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q},\quad \boldsymbol{v}_1=\frac{\gamma(1+\delta_Q)}{\lambda_Q}\mathbf{W}^\top\boldsymbol{\mu}_\beta,\quad a_2=T_1-a_1^2=\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{h\lambda_Q}-\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q}-\frac{2(1-h)}{h}\right)+\frac{1-h}{h}$$

$$\boldsymbol{v}_2=\frac{(1+\delta_Q)}{\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{h\lambda_Q}-\frac{2\gamma\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q}-\frac{1-h}{h}\right)\mathbf{W}^\top\boldsymbol{\mu}_\beta,$$

$$\tilde{M} = \frac{\gamma^2(1+\delta_Q)^2(1-h)}{h}\left(\frac{1}{\eta}\mathbf{W}^\top\mathbf{W} + \left(\frac{1}{\lambda_Q^2} + \frac{1}{\eta\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q} - 2\right)\right)\mathbf{W}^\top\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top\mathbf{W}^\top\right)$$

And therefore, since φ is non-decreasing, maximizing this test accuracy boils down to maximizing the term inside it, i.e we want to find $\boldsymbol{\alpha}^*$ that satisfies:

$$\boldsymbol{\alpha}^* \in \arg\max_{\boldsymbol{\alpha}} \frac{a_1 + \boldsymbol{\alpha}^\top\boldsymbol{v}_1}{\sqrt{a_2 + \boldsymbol{\alpha}^\top\boldsymbol{v}_2 + \boldsymbol{\alpha}^\top\tilde{M}\boldsymbol{\alpha}}} = \arg\max_{\boldsymbol{\alpha}} g(\boldsymbol{\alpha})$$

We compute the gradient of g with respect to $\boldsymbol{\alpha}$ to find the extremum values of these mixing parameters:

$$\nabla_\alpha g(\boldsymbol{\alpha}) = \frac{\sqrt{a_2 + \boldsymbol{\alpha}^\top\boldsymbol{v}_2 + \boldsymbol{\alpha}\tilde{\mathbf{M}}\boldsymbol{\alpha}}\ \boldsymbol{v}_1 - (a_1 + \boldsymbol{\alpha}^\top\boldsymbol{v}_1)\frac{\boldsymbol{v}_2 + 2\tilde{\mathbf{M}}\boldsymbol{\alpha}}{\sqrt{a_2 + \boldsymbol{\alpha}^\top\boldsymbol{v}_2 + \boldsymbol{\alpha}^\top\tilde{\mathbf{M}}\boldsymbol{\alpha}}}}{a_2 + \boldsymbol{\alpha}^\top\boldsymbol{v}_2 + \boldsymbol{\alpha}^\top\tilde{\mathbf{M}}\boldsymbol{\alpha}}$$

Thus the roots $\boldsymbol{\alpha}$ of $\nabla g(\boldsymbol{\alpha})$ satisfy the following equation:

$$\boxed{(a_2 + \boldsymbol{\alpha}^\top\boldsymbol{v}_2 + \boldsymbol{\alpha}^\top\tilde{\mathbf{M}}\boldsymbol{\alpha})\boldsymbol{v}_1 - (a_1 + \boldsymbol{\alpha}^\top\boldsymbol{v}_1)(\boldsymbol{v}_2 + 2\tilde{\mathbf{M}}\boldsymbol{\alpha}) = 0}$$

We summarize these findings in the following theorem:

Theorem 6.2 (Optimal α for the mixture of source classifiers)**.** *Under Assumptions 4.1, the optimal scaling factors $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_T)^T$ satisfy the following identity:*

$$(a_2 + \boldsymbol{\alpha}^\top\boldsymbol{v}_2 + \boldsymbol{\alpha}^\top\tilde{\mathbf{M}}\boldsymbol{\alpha})\boldsymbol{v}_1 - (a_1 + \boldsymbol{\alpha}^\top\boldsymbol{v}_1)(\boldsymbol{v}_2 + 2\tilde{\mathbf{M}}\boldsymbol{\alpha}) = 0$$

Whereas the worst coefficients satisfy:

$$\boldsymbol{\alpha}^\top\mathbf{W}^\top\boldsymbol{\mu}_\beta = -\frac{\|\boldsymbol{\mu}_\beta\|^2}{\gamma(1+\delta_Q)}$$

We next show that similar principles hold in continuous-output settings such as regression, suggesting that α-scaling reflects a general statistical phenomenon rather than a classification-specific artifact.

7 The case of regression: fine-tuning a weight matrix

Let consider now analyzing a linear regression task where the finetuning process is done using an adapter matrix (instead of vector), which is described by the following setting.

Source task. Assume we are given a source regression dataset $\{(\tilde{\boldsymbol{x}}_i, \tilde{\boldsymbol{y}}_i)\}_{i=1}^N$ such that there exists $\mathbf{W}_s \in \mathbb{R}^{d\times p}$:

$$\tilde{\boldsymbol{x}}_i \sim \mathscr{N}(0, \mathbf{I}_p), \quad \tilde{\boldsymbol{y}}_i = \mathbf{W}_s\tilde{\boldsymbol{x}}_i + \tilde{\boldsymbol{z}}_i \in \mathbb{R}^d, \quad \text{where: } \tilde{\boldsymbol{z}}_i \sim \mathscr{N}(0, \tilde{\sigma}^2\mathbf{I}_d) \tag{16}$$

Target task. Now we want to learn a target task characterized by a matrix $\mathbf{W}_t$. In fact, we consider having a target dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ defined as:

$$\boldsymbol{x}_i \sim \mathscr{N}(0, \mathbf{I}_p), \quad \boldsymbol{y}_i = \mathbf{W}_t\boldsymbol{x}_i + \boldsymbol{z}_i \in \mathbb{R}^d, \quad \text{where: } \boldsymbol{z}_i \sim \mathscr{N}(0, \sigma^2\mathbf{I}_d) \tag{17}$$

Again we denote by: $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{p \times n}$ and $\mathbf{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n] \in \mathbb{R}^{d \times n}$. We want to learn such target task by considering weights of the form: $\alpha \mathbf{W}_s + \mathbf{A}$ such that $\mathbf{A}$ is learned from the target data by minimizing the following loss function:

$$\min_{\mathbf{V}} \mathscr{L}(\mathbf{V}; \mathbf{X}, \mathbf{Y}) = \min_{\mathbf{V}} \frac{1}{n} \sum_{i=1}^{n} \|(\alpha \mathbf{W}_s + \mathbf{V})\boldsymbol{x}_i - \boldsymbol{y}_i\|_2^2 + \gamma \|\mathbf{V}\|_F^2 \tag{18}$$

Where $\|.\|_F$ denotes the Frobenius norm of a matrix. The gradient of the loss function with respect to $\mathbf{V}$ is given by:

$$\nabla \mathscr{L}(\mathbf{V}) = \frac{2}{n} \sum_{i=1}^{n} \left((\alpha \mathbf{W}_s + \mathbf{V})\boldsymbol{x}_i - \boldsymbol{y}_i\right) \boldsymbol{x}_i^\top + 2\gamma \mathbf{V} = \frac{2}{n} \left(\alpha \mathbf{W}_s + \mathbf{V}\right) \mathbf{X}\mathbf{X}^\top - \frac{2}{n} \mathbf{Y}\mathbf{X}^\top + 2\gamma \mathbf{V}$$

Thus, the minimizer $\mathbf{A}$ of the loss function $\mathscr{L}(\mathbf{V})$ is given by:

$$\mathbf{A} = \frac{1}{n} \mathbf{Y}\mathbf{X}^\top \mathbf{Q} - \frac{\alpha}{n} \mathbf{W}_s \mathbf{X}\mathbf{X}^\top \mathbf{Q} \tag{19}$$

where:

$$\mathbf{Q} = \left(\frac{1}{n} \mathbf{X}\mathbf{X}^\top + \gamma \mathbf{I}_p\right)^{-1} \tag{20}$$

Finally, the fine-tuned regressor is given by:

$$\mathbf{W}_\alpha = \alpha \mathbf{W}_s + \mathbf{A} = \frac{1}{n} \mathbf{Y}\mathbf{X}^\top \mathbf{Q} + \alpha\gamma \mathbf{W}_s \mathbf{Q} \tag{21}$$

To evaluate the efficiency of this fine-tuning process and showcase the impact of the parameter α, we can compute the theoretical test error of this regressor defined as follows:

$$E_{\text{test}} = \mathbb{E}\left[\|\mathbf{W}_\alpha \boldsymbol{x} - \boldsymbol{y}\|^2\right] \tag{22}$$

for test samples $(\boldsymbol{x}, \boldsymbol{y})$ independent of the training set. Denote by:

$$\lambda = 1 + \gamma(1 + \delta)$$

The following theorem gives the theoretical expression of the test error of the fine-tuned regressor $\mathbf{W}_\alpha$.

Theorem 7.1 (Theoretical test error.)**.** *Under the high-dimensional regime, i.e* $\frac{p}{n} \to \eta \in [0, +\infty)$*, the theoretical test error of the fine-tuned regressor* $\mathbf{W}_\alpha$ *defined in* (22) *is given by:*

$$E_{test} = T_1 + \alpha T_2 + \alpha^2 T_3,$$

where:

$$T_1 = \frac{(\lambda - 1)^2}{\lambda^2 - \eta} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_t^\top) + \frac{\sigma^2 . d . \lambda^2}{\lambda^2 - \eta}$$
$$T_2 = \frac{2\gamma(1 + \delta)(1 - \lambda)}{\lambda^2 - \eta} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top)$$
$$T_3 = \frac{(\gamma(1 + \delta))^2}{\lambda^2 - \eta} \operatorname{Tr}(\mathbf{W}_s \mathbf{W}_s^\top)$$

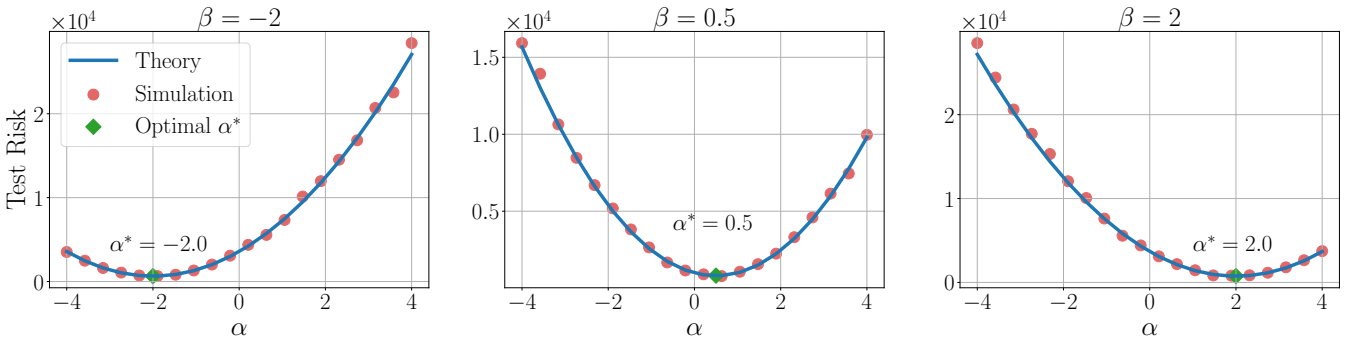

Figure 9: Test risk variation with α for a transfer learning setting starting from a fixed (random) source regressor $\mathbf{W}_s$ to a target task of the form: $\mathbf{W}_t = \beta \mathbf{W}_s + \mathbf{W}_s^{\perp}$. The considered parameters here are: $n = 20$, $p = 200$, $d = 4$, $\gamma = 10^{-2}$ and $\sigma = 0.5$.

The proof of this theorem is presented in Appendix E. In particular, we can easily derive the optimal α to use in the fine-tuning process which we present in the following theorem.

Theorem 7.2 (Optimal regression α^*.)**.** *Under the same assumptions of the previous theorem, the optimal α^* that minimizes the theoretical test error E_{test} of the fine-tuned regressor is given by:*

$$\alpha^* = \frac{\mathrm{Tr}(\mathbf{W}_t \mathbf{W}_s^\top)}{\mathrm{Tr}(\mathbf{W}_s \mathbf{W}_s^\top)}$$

Again, the proof of this theorem is provided in Appendix E. We remark here for instance that the optimal parameter α^* does **not** depend on the dimensionality of the problem, nor on the number of fine-tuning samples n, which is an **interesting and unexpected** property that was not observed in the previously studied classification setting. Additionally, α^* can also be interpreted as a normalized alignment score between the source and target tasks. In fact, we know that the Frobenius dot product between two matrices $\mathbf{A}$ and $\mathbf{B}$ is given by: $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{Tr}(\mathbf{A}\mathbf{B}^\top)$, hence: $\alpha^* = \frac{\langle \mathbf{W}_t, \mathbf{W}_s \rangle}{\|\mathbf{W}_s\|^2}$. Therefore the optimal fine-tuning parameter α to choose is exactly the alignment score between the source and target tasks as defined earlier in (6) This is further shown in Figure 9.

Estimating α^*. From our derivations in Appendix E, we can derive a consistent estimator of α^*. In fact, for any source samples $(\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{y}}_1)$ and $(\tilde{\boldsymbol{x}}_2, \tilde{\boldsymbol{y}}_2)$, and target samples $(\boldsymbol{x}, \boldsymbol{y})$, we have that:

$$\mathbb{E}[\tilde{\boldsymbol{y}}_1^\top \boldsymbol{y} \boldsymbol{x}^\top \tilde{\boldsymbol{x}}_1] = \mathrm{Tr}(\mathbf{W}_t \mathbf{W}_s^\top), \quad \mathbb{E}[\tilde{\boldsymbol{y}}_1^\top \tilde{\boldsymbol{y}}_2 \tilde{\boldsymbol{x}}_2^\top \tilde{\boldsymbol{x}}_1] = \mathrm{Tr}(\mathbf{W}_s \mathbf{W}_s^\top)$$

Therefore:

$$\boxed{\alpha^* = \frac{\mathbb{E}[\tilde{\boldsymbol{y}}_1^\top \boldsymbol{y} \boldsymbol{x}^\top \tilde{\boldsymbol{x}}_1]}{\mathbb{E}[\tilde{\boldsymbol{y}}_1^\top \tilde{\boldsymbol{y}}_2 \tilde{\boldsymbol{x}}_2^\top \tilde{\boldsymbol{x}}_1]}} \tag{23}$$

And therefore, we can deploy Monte Carlo methods to estimate these two expectations.

Conceptual reflection. This work reframes fine-tuning as a problem of balancing knowledge transfer rather than merely parameter adaptation. By introducing a scaling parameter that explicitly governs the contribution of pre-trained representations, we provide a theoretical and algorithmic mechanism to control how prior knowledge is reused. This interpretation connects transfer learning to broader principles in optimization and statistical physics, where equilibrium between old and new information determines generalization.

8 Discussion and Conclusion

In this thesis, we introduced a new theoretical and algorithmic framework for fine-tuning pre-trained models through an additional scaling degree of freedom. By reparameterizing the adaptation process with a learnable scaling parameter, we demonstrated—both analytically and empirically—that fine-tuning can achieve superior generalization compared to standard low-rank approaches such as LoRA.

Our Random Matrix Theory analysis revealed the existence of an optimal scaling factor that minimizes generalization error in high-dimensional transfer settings. Interestingly, this optimal value is often distinct from the conventional scaling ($\alpha = 1$) used in prior work, providing a rigorous theoretical justification for adaptive rescaling during fine-tuning. The theory not only yields interpretable expressions linking α to alignment between source and target tasks but also extends naturally to multi-source and regression frameworks.

Empirically, our proposed α-LoRA method consistently improves performance on benchmark transfer tasks and LLM fine-tuning experiments, supporting the theoretical predictions. This dual validation—mathematical and empirical—underscores the relevance of RMT-based analysis for guiding fine-tuning design.

Nevertheless, our study also highlights several limitations. The theoretical results rely on simplifying assumptions such as Gaussian data distributions and linearized architectures, which do not capture the full complexity of modern deep networks. Future work could aim to: (i) relax these assumptions to handle more realistic data and architectures, (ii) develop efficient estimators for α in large-scale scenarios, and (iii) explore synergistic combinations of α-scaling with other advanced adapter techniques (e.g., DoRA, MoRA, or LoRA+). Beyond methodological extensions, an intriguing avenue is the use of some theoretical tools (in addition to RMT) tools to predict fine-tuning dynamics or to design other adaptive algorithms that automatically adjust α during training.

In summary, this work takes a first step toward a principled, theoretically grounded understanding of fine-tuning, bridging Random Matrix Theory and practical transfer learning. We hope it inspires future research into interpretable and mathematically guided adaptation mechanisms for large-scale learning systems.

References

Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. A theory of learning from different domains. In *Machine Learning*, volume 79, pp. 151–175. Springer, 2010.

John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pp. 440–447, 2007.

T Tony Cai and Hongji Wei. Transfer learning for nonparametric classification. *The Annals of Statistics*, 49(1):100–128, 2021.

Yatin Dandi, Ludovic Stephan, Florent Krzakala, Bruno Loureiro, and Lenka Zdeborová. Universality laws for gaussian mixtures in generalized linear models. *Advances in Neural Information Processing Systems*, 36, 2024.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Aymane El Firdoussi and Mohamed El Amine Seddik. High-dimensional learning with noisy labels. *arXiv preprint arXiv:2405.14088*, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Demi Guo, Alexander M Rush, and Yoon Kim. Parameter-efficient transfer learning with diff pruning. *arXiv preprint arXiv:2012.07463*, 2020.

Walid Hachem, Philippe Loubaton, and Jamal Najim. Deterministic equivalents for certain functionals of large random matrices. 2007.

Steve Hanneke and Samory Kpotufe. A more unified theory of transfer learning. *arXiv preprint arXiv:2408.16189*, 2024.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. The impact of initialization on lora finetuning dynamics. *Advances in Neural Information Processing Systems*, 37:117015–117040, 2024a.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. Lora+: Efficient low rank adaptation of large models. *arXiv preprint arXiv:2402.12354*, 2024b.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. Plop: Precise lora placement for efficient finetuning of large models. *arXiv preprint arXiv:2506.20629*, 2025.

Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*, 2021.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Wenlong Ji, Weizhe Yuan, Emily Getzen, Kyunghyun Cho, Michael I Jordan, Song Mei, Jason E Weston, Weijie J Su, Jing Xu, and Linjun Zhang. An overview of large language models for statisticians. *arXiv preprint arXiv:2502.17814*, 2025.

Ting Jiang, Shaohan Huang, Shengyue Luo, Zihan Zhang, Haizhen Huang, Furu Wei, Weiwei Deng, Feng Sun, Qi Zhang, Deqing Wang, and otxhers. Mora: High-rank updating for parameter-efficient fine-tuning. *arXiv preprint arXiv:2405.12130*, 2024.

Rabeeh Karimi Mahabadi, James Henderson, and Sebastian Ruder. Compacter: Efficient low-rank hypercomplex adapter layers. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 1022–1035, 2021.

Minsoo Kim, Sihwa Lee, Wonyong Sung, and Jungwook Choi. Ra-lora: Rank-adaptive parameter-efficient fine-tuning for accurate 2-bit quantized large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pp. 15773–15786, 2024.

Adam Klivans, Konstantinos Stavropoulos, and Arsen Vasilyan. Testable learning with distribution shift. In *The Thirty Seventh Annual Conference on Learning Theory*, pp. 2887–2943. PMLR, 2024.

Dawid J Kopiczko, Tijmen Blankevoort, and Yuki M Asano. Vera: Vector-based random matrix adaptation. *arXiv preprint arXiv:2310.11454*, 2023.

Samory Kpotufe and Guillaume Martinet. Marginal singularity and the benefits of labels in covariate-shift. *The Annals of Statistics*, 49(6):3299–3323, 2021.

Baohao Liao, Yan Meng, and Christof Monz. Parameter-efficient fine-tuning without introducing new latency. *arXiv preprint arXiv:2305.16742*, 2023.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. In *Forty-first International Conference on Machine Learning*, 2024.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL http://arxiv.org/abs/1907.11692.

Cosme Louart and Romain Couillet. Concentration of measure and large random matrices with an application to sample covariance matrices. *arXiv preprint arXiv:1805.08295*, 2018.

Haodong Lu, Chongyang Zhao, Jason Xue, Lina Yao, Kristen Moore, and Dong Gong. Adaptive rank, reduced forgetting: Knowledge retention in continual learning vision-language models with dynamic rank-selective lora. *arXiv preprint arXiv:2412.01004*, 2024.

Xiaoyi Mai and Zhenyu Liao. The breakdown of gaussian universality in classification of high-dimensional mixtures. *arXiv preprint arXiv:2410.05609*, 2024.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32, 2016.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. Adapterfusion: Non-destructive task composition for transfer learning. In *European Chapter of the Association for Computational Linguistics (EACL)*, 2020.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Shyam Sundhar Ramesh, Yifan Hu, Iason Chaimalas, Viraj Mehta, Pier Giuseppe Sessa, Haitham Bou Ammar, and Ilija Bogunovic. Group robust preference optimization in reward-free rlhf. *Advances in Neural Information Processing Systems*, 37:37100–37137, 2024.

Henry WJ Reeve, Timothy I Cannings, and Richard J Samworth. Adaptive transfer learning. *The Annals of Statistics*, 49(6):3618–3649, 2021.

Mohamed El Amine Seddik, Cosme Louart, Mohamed Tamaazousti, and Romain Couillet. Random matrix theory proves that deep learning representations of gan-data behave as gaussian mixtures. In *International Conference on Machine Learning*, pp. 8573–8582. PMLR, 2020.

Idan Shenfeld, Jyothish Pari, and Pulkit Agrawal. Rl's razor: Why online reinforcement learning forgets less. *arXiv preprint arXiv:2509.04259*, 2025.

Yi-Lin Sung, Varun Nair, and Colin A Raffel. Training neural networks with fixed sparse masks. *Advances in Neural Information Processing Systems*, 34:24193–24205, 2021.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Cheng-Zhong Xu. Hydralora: An asymmetric lora architecture for efficient fine-tuning. *Advances in Neural Information Processing Systems*, 37:9565–9584, 2024.

Nilesh Tripuraneni, Michael Jordan, and Chi Jin. On the theory of transfer learning: The importance of task diversity. *Advances in neural information processing systems*, 33:7852–7862, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big data*, 3(1):9, 2016.

Lingling Xu, Haoran Xie, Si-Zhao Joe Qin, Xiaohui Tao, and Fu Lee Wang. Parameter-efficient fine-tuning methods for pretrained language models: A critical review and assessment. *arXiv preprint arXiv:2312.12148*, 2023.

Guojun Zhang, Han Zhao, Yaoliang Yu, and Pascal Poupart. Quantifying and improving transferability in domain generalization. *Advances in Neural Information Processing Systems*, 34:10957–10970, 2021.

Longteng Zhang, Lin Zhang, Shaohuai Shi, Xiaowen Chu, and Bo Li. Lora-fa: Memory-efficient low-rank adaptation for large language models fine-tuning. *arXiv preprint arXiv:2308.03303*, 2023a.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*, 2023b.

Supplementary material

Notations. Here are two notations that we will use along the whole analysis:

$$\lambda_Q = \|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q), \quad \lambda_R = \|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R) \tag{24}$$

A Useful results

A.1 General lemmas

Here we will list useful lemmas used in our analysis.

Lemma A.1 (Resolvent identity)**.** *For invertible matrices* $\mathbf{A}$ *and* $\mathbf{B}$*, we have:*

$$\mathbf{A}^{-1} - \mathbf{B}^{-1} = \mathbf{A}^{-1}(\mathbf{B} - \mathbf{A})\mathbf{B}^{-1}.$$

Lemma A.2 (Sherman-Morisson)**.** *For* $\mathbf{A} \in \mathbb{R}^{p\times p}$ *invertible and* $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^p$*,* $\mathbf{A} + \boldsymbol{u}\boldsymbol{v}^\top$ *is invertible if and only if:* $1 + \boldsymbol{v}^\top \mathbf{A}^{-1}\boldsymbol{u} \neq 0$*, and:*

$$(\mathbf{A} + \boldsymbol{u}\boldsymbol{v}^\top)^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\boldsymbol{u}\boldsymbol{v}^\top\mathbf{A}^{-1}}{1 + \boldsymbol{v}^\top\mathbf{A}^{-1}\boldsymbol{u}}.$$

Besides,

$$(\mathbf{A} + \boldsymbol{u}\boldsymbol{v}^\top)^{-1}\boldsymbol{u} = \frac{\mathbf{A}^{-1}\boldsymbol{u}}{1 + \boldsymbol{v}^\top\mathbf{A}^{-1}\boldsymbol{u}}.$$

A.2 Deterministic equivalents

Recall the expression of the resolvents defined in equation (9):

$$\mathbf{Q} = \left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + \gamma\mathbf{I}_p\right)^{-1}, \mathbf{R} = \left(\frac{1}{N}\tilde{\mathbf{X}}\tilde{\mathbf{X}}^\top + \tilde{\gamma}\mathbf{I}_p\right)^{-1}$$

We define the matrices $\mathbf{Q}_{-i}$ and $\mathbf{R}_{-i}$ as the resolvents obtained by removing the contribution of the i^{th} sample, i.e:

$$\mathbf{Q}_{-i} = \left(\frac{1}{n}\sum_{k\neq i}\boldsymbol{x}_k\boldsymbol{x}_k^\top + \gamma\mathbf{I}_p\right)^{-1}, \quad \mathbf{R}_{-i} = \left(\frac{1}{N}\sum_{k\neq i}\tilde{\boldsymbol{x}}_k\tilde{\boldsymbol{x}}_k^\top + \tilde{\gamma}\mathbf{I}_p\right)^{-1}$$

then we have that:

$$\mathbf{Q} = \left(\mathbf{Q}_{-i}^{-1} + \frac{1}{n}\boldsymbol{x}_i\boldsymbol{x}_i^\top\right)^{-1}, \quad \mathbf{R} = \left(\mathbf{R}_{-i}^{-1} + \frac{1}{N}\tilde{\boldsymbol{x}}_i\tilde{\boldsymbol{x}}_i^\top\right)^{-1}$$

Thus by Sherman-Morisson's lemma:

$$\mathbf{Q} = \mathbf{Q}_{-i} - \frac{1}{n}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{1+\delta_Q}, \quad \mathbf{R} = \mathbf{R}_{-i} - \frac{\frac{1}{N}\mathbf{R}_{-i}\tilde{\boldsymbol{x}}_i\tilde{\boldsymbol{x}}_i^\top\mathbf{R}_{-i}}{1+\delta_R}$$

where:

$$\delta_Q = \frac{1}{n}\operatorname{Tr}\bar{\mathbf{Q}} = \frac{\eta - \gamma - 1 + \sqrt{(\eta-\gamma-1)^2 + 4\eta\gamma}}{2\gamma}, \quad \delta_R = \frac{1}{N}\operatorname{Tr}\bar{\mathbf{R}} = \frac{\tilde{\eta} - \tilde{\gamma} - 1 + \sqrt{(\tilde{\eta}-\tilde{\gamma}-1)^2 + 4\tilde{\eta}\gamma}}{2\tilde{\gamma}}$$

Thus, we get that:

$$\mathbf{Q}\boldsymbol{x}_i = \frac{\mathbf{Q}_{-i}\boldsymbol{x}_i}{1+\delta_Q}, \quad \mathbf{R}\tilde{\boldsymbol{x}}_i = \frac{\mathbf{R}_{-i}\tilde{\boldsymbol{x}}_i}{1+\delta_R} \tag{25}$$

Using the above identities, we can easily prove the deterministic equivalents of $\mathbf{Q}$ and $\mathbf{R}$ stated in Lemma 3.6, which we will do in the following.

Lemma A.3 (Deterministic equivalent of $\mathbf{Q}$ and $\mathbf{R}$)**.** *Under the high-dimensional regime and the assumptions 4.1, a deterministic equivalent for $\mathbf{Q} \equiv \mathbf{Q}(\gamma)$ and for $\mathbf{R} \equiv \mathbf{R}(\gamma)$, denoted $\bar{\mathbf{Q}}$ and $\bar{\mathbf{R}}$ respectively, as defined in (9) are given by:*

$$\bar{\mathbf{Q}}(\gamma) = \left(\frac{\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top + \mathbf{I}_p}{1+\delta_Q} + \gamma\mathbf{I}_p\right)^{-1}, \quad \bar{\mathbf{R}}(\gamma) = \left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top + \mathbf{I}_p}{1+\delta_R} + \gamma\mathbf{I}_p\right)^{-1}.$$

Where:

$$\delta_Q = \frac{1}{n}\operatorname{Tr}\bar{\mathbf{Q}} = \frac{\eta-\gamma-1+\sqrt{(\eta-\gamma-1)^2+4\eta\gamma}}{2\gamma}, \quad \delta_R = \frac{1}{N}\operatorname{Tr}\bar{\mathbf{R}} = \frac{\tilde{\eta}-\tilde{\gamma}-1+\sqrt{(\tilde{\eta}-\tilde{\gamma}-1)^2+4\tilde{\eta}\gamma}}{2\tilde{\gamma}}.$$

Lemma A.4 (Trace identities)**.** *Let $\bar{\mathbf{Q}}, \bar{\mathbf{R}} \in \mathbb{R}^{p\times p}$ be the deterministic matrices defined in lemma 3.6. Then:*

$$\frac{1}{n}\frac{\operatorname{Tr}((\Sigma_\beta\bar{\mathbf{Q}})^2)}{(1+\delta_Q)^2} = \frac{\eta}{(1+\gamma(1+\delta_Q))^2}, \quad \frac{1}{N}\frac{\operatorname{Tr}((\Sigma\bar{\mathbf{R}})^2)}{(1+\delta_R)^2} = \frac{\tilde{\eta}}{(1+\tilde{\gamma}(1+\delta_R))^2}.$$

And:

$$\frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}^2\bar{\mathbf{Q}}^2) = \tilde{\eta}\left(\frac{(1+\delta_R)(1+\delta_Q)}{(1+\tilde{\gamma}(1+\delta_R))(1+\gamma(1+\delta_Q))}\right)^2$$

Lemma A.5 (Relevant Identities)**.** *Let $\bar{\mathbf{Q}}, \bar{\mathbf{R}} \in \mathbb{R}^{p\times p}$ be the deterministic matrices defined in lemma 3.6. Then we have the following identities:*

$$\boldsymbol{\mu}_\beta^\top\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta = \frac{(1+\delta_Q)\|\boldsymbol{\mu}_\beta\|^2}{\|\boldsymbol{\mu}_\beta\|^2+1+\gamma(1+\delta_Q)}, \quad \boldsymbol{\mu}_\beta^\top\bar{\mathbf{Q}}^2\boldsymbol{\mu}_\beta = \left(\frac{(1+\delta_Q)\|\boldsymbol{\mu}_\beta\|}{\|\boldsymbol{\mu}_\beta\|^2+1+\gamma(1+\delta_Q)}\right)^2,$$

$$\boldsymbol{\mu}^\top\bar{\mathbf{R}}\boldsymbol{\mu} = \frac{(1+\delta_R)\|\boldsymbol{\mu}\|^2}{\|\boldsymbol{\mu}\|^2+1+\tilde{\gamma}(1+\delta_R)}, \quad \boldsymbol{\mu}^\top\bar{\mathbf{R}}^2\boldsymbol{\mu} = \left(\frac{(1+\delta_R)\|\boldsymbol{\mu}\|}{\|\boldsymbol{\mu}\|^2+1+\tilde{\gamma}(1+\delta_R)}\right)^2,$$

$$\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta = \frac{(1+\delta_R)(1+\delta_Q)\beta\|\boldsymbol{\mu}\|^2}{(\|\boldsymbol{\mu}\|^2+1+\tilde{\gamma}(1+\delta_R))(\|\boldsymbol{\mu}_\beta\|^2+1+\gamma(1+\delta_Q))},$$

$$\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}^2\boldsymbol{\mu}_\beta = \frac{(1+\delta_R)}{(\|\boldsymbol{\mu}\|^2+1+\tilde{\gamma}(1+\delta_R))}\left(\frac{(1+\delta_Q)}{(\|\boldsymbol{\mu}_\beta\|^2+1+\gamma(1+\delta_Q))}\right)^2\beta\|\boldsymbol{\mu}\|^2,$$

And finally:

$$\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}^2\bar{\mathbf{R}}\boldsymbol{\mu}$$
$$= \left(\frac{(1+\delta_R)(1+\delta_Q)\|\boldsymbol{\mu}\|}{(1+\gamma(1+\delta_Q))(\|\boldsymbol{\mu}\|^2+1+\tilde{\gamma}(1+\delta_R))}\right)^2\left(1+\frac{\beta^3\|\boldsymbol{\mu}\|^4}{(\|\boldsymbol{\mu}_\beta\|^2+1+\gamma(1+\delta_Q))^2}-\frac{2\beta^2\|\boldsymbol{\mu}\|^2}{\|\boldsymbol{\mu}_\beta\|^2+1+\gamma(1+\delta_Q)}\right).$$

Proof. The proof of all these identities relies on the following results:

$$\begin{aligned}
\bar{\mathbf{R}} &= \left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top}{1+\delta_R} + \left(\tilde{\gamma} + \frac{1}{1+\delta_R}\right)\mathbf{I}_p\right)^{-1} \\
&= (1+\delta_R)\left(\boldsymbol{\mu}\boldsymbol{\mu}^\top + (1+\tilde{\gamma}(1+\delta_R)\mathbf{I}_p)\right)^{-1} \\
&= \frac{1+\delta_R}{1+\tilde{\gamma}(1+\delta_R)}\left(\frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top}{1+\tilde{\gamma}(1+\delta_R)} + \mathbf{I}_p\right)^{-1} \\
&= \frac{1+\delta_R}{1+\tilde{\gamma}(1+\delta_R)}\left(\mathbf{I}_p - \frac{\boldsymbol{\mu}\boldsymbol{\mu}^\top}{\|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R)}\right) \qquad \text{(lemma A.2)}
\end{aligned}$$

where the last equality is obtained using Sherman-Morisson's identity (lemma A.2). Hence,

$$(\bar{\mathbf{R}})^2 = \frac{(1+\delta_R)^2}{(1+\tilde{\gamma}(1+\delta_R))^2}\left(\mathbf{I}_p + \frac{(\boldsymbol{\mu}\boldsymbol{\mu}^\top)^2}{(\|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R))^2} - \frac{2\boldsymbol{\mu}\boldsymbol{\mu}^\top}{\|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R)}\right).$$

And the same for $\bar{\mathbf{Q}}$:

$$\bar{\mathbf{Q}} = \frac{1+\delta_Q}{1+\gamma(1+\delta_Q)}\left(\mathbf{I}_p - \frac{\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)}\right),$$

$$(\bar{\mathbf{Q}})^2 = \frac{(1+\delta_Q)^2}{(1+\gamma(1+\delta_Q))^2}\left(\mathbf{I}_p + \frac{(\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top)^2}{(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))^2} - \frac{2\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)}\right).$$

And using the second identity in Sherman-Morisson's lemma A.2:

$$\bar{\mathbf{R}}\boldsymbol{\mu} = \frac{(1+\delta_R)}{\|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R)}\boldsymbol{\mu}, \quad \bar{\mathbf{Q}}\boldsymbol{\mu}_\beta = \frac{(1+\delta_Q)}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)}\boldsymbol{\mu}_\beta$$

□

Lemma A.6 (Expectation some classifiers)**.** *Let $\tilde{\boldsymbol{w}}$ and $\boldsymbol{w}$ be the classifiers defined earlier (α-FTC). We have that:*

$$\mathbb{E}[\tilde{\boldsymbol{w}}] = \frac{1}{1+\delta_R}\bar{\mathbf{R}}\boldsymbol{\mu}, \quad \mathbb{E}[\boldsymbol{w}] = \frac{1}{1+\delta_Q}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta.$$

Proof.

$$\begin{aligned}
\mathbb{E}[\tilde{\boldsymbol{w}}] &= \frac{1}{N}\sum_{i=1}^N \mathbb{E}[\tilde{y}_i \mathbf{R}\tilde{\boldsymbol{x}}_i] \\
&= \frac{1}{N}\sum_{i=1}^N \frac{1}{1+\delta_R}\mathbb{E}[\tilde{y}_i \mathbf{R}_{-i}\tilde{\boldsymbol{x}}_i] \\
&= \frac{1}{1+\delta_R}\bar{\mathbf{R}}\boldsymbol{\mu}
\end{aligned}$$

The proof of $\mathbb{E}[\boldsymbol{w}]$ is similar to this latter. □

Lemma A.7 (Deterministic equivalent)**.** *For any positive semi-definite matrix* $\mathbf{A}$*, we have:*

$$\mathbf{QAQ} \leftrightarrow \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n}\frac{\operatorname{Tr}(\Sigma_\beta \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})}{(1+\delta_Q)^2}\,\mathbb{E}[\mathbf{Q}\Sigma_\beta\mathbf{Q}],$$

and:

$$\mathbf{RAR} \leftrightarrow \bar{\mathbf{R}}\mathbf{A}\bar{\mathbf{R}} + \frac{1}{N}\frac{\operatorname{Tr}(\Sigma\bar{\mathbf{R}}\mathbf{A}\bar{\mathbf{R}})}{(1+\delta_R)^2}\,\mathbb{E}[\mathbf{R}\Sigma\mathbf{R}].$$

In particular for every $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^p$*:*

$$\boldsymbol{a}^\top \mathbb{E}[\mathbf{Q}\Sigma_\beta\mathbf{Q}]\boldsymbol{b} = \frac{1}{h}\boldsymbol{a}^\top\bar{\mathbf{Q}}\Sigma_\beta\bar{\mathbf{Q}}\boldsymbol{b}, \quad \boldsymbol{a}^\top \mathbb{E}[\mathbf{R}\Sigma\mathbf{R}]\boldsymbol{b} = \frac{1}{\tilde{h}}\boldsymbol{a}^\top\bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}\boldsymbol{b}.$$

Proof. The proof is derived similarly as in the appendix of Firdoussi & Seddik (2024). Again, the proof is similar for both $\mathbf{Q}$ and $\mathbf{R}$.
Let $\bar{\mathbf{Q}}$ be a deterministic equivalent of $\mathbf{Q}$. The following equations and identities are valid in terms of linear forms. We have that:

$$\begin{aligned}
\mathbb{E}[\mathbf{QAQ}] &= \mathbb{E}[\bar{\mathbf{Q}}\mathbf{AQ}] + \mathbb{E}[(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{AQ}] \\
&= \bar{\mathbf{Q}}(\mathbb{E}[\mathbf{AQ}] + \mathbf{A}\,\mathbb{E}[\mathbf{Q}-\bar{\mathbf{Q}}]) + \mathbb{E}[(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{AQ}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \mathbb{E}[(\mathbf{Q}-\bar{\mathbf{Q}})\mathbf{AQ}]
\end{aligned}$$

Using lemma A.1, we have that:

$$\begin{aligned}
\mathbf{Q}-\bar{\mathbf{Q}} &= \mathbf{Q}(\bar{\mathbf{Q}}^{-1}-\mathbf{Q}^{-1})\bar{\mathbf{Q}} \\
&= \mathbf{Q}\left(\frac{\Sigma_\beta}{1+\delta_Q} - \frac{1}{n}\mathbf{X}\mathbf{X}^\top\right)\bar{\mathbf{Q}} \\
&= \mathbf{Q}\left(\mathbf{S} - \frac{1}{n}\mathbf{X}\mathbf{X}^\top\right)\bar{\mathbf{Q}}
\end{aligned}$$

Thus:

$$\begin{aligned}
\mathbb{E}[\mathbf{QAQ}] &= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \mathbb{E}[\mathbf{Q}(\mathbf{S}-\frac{1}{n}\mathbf{X}\mathbf{X}^\top)\bar{\mathbf{Q}}\mathbf{AQ}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \mathbb{E}[\mathbf{QS}\bar{\mathbf{Q}}\mathbf{AQ}] - \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{AQ}]
\end{aligned}$$

We have that:

$$\begin{aligned}
\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{AQ}] &= \frac{1}{1+\delta_Q}\,\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{AQ}] \\
&= \frac{1}{1+\delta_Q}\left(\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{Q}_{-i}] - \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{n(1+\delta_Q)}]\right) \\
&= \frac{1}{1+\delta_Q}\left(\mathbb{E}[\mathbf{Q}_{-i}\Sigma_\beta\bar{\mathbf{Q}}\mathbf{AQ}_{-i}] - \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{n(1+\delta_Q)}]\right) \\
&= \frac{1}{1+\delta_Q}\left(\mathbb{E}[\mathbf{Q}\Sigma_\beta\bar{\mathbf{Q}}\mathbf{AQ}] - \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{n(1+\delta_Q)}]\right)
\end{aligned}$$

Therefore, by replacing the obtained expression of $\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\mathbf{Q}]$ in the equation of $\mathbb{E}[\mathbf{Q}\mathbf{A}\mathbf{Q}]$, we get that:

$$\begin{aligned}
\mathbb{E}[\mathbf{Q}\mathbf{A}\mathbf{Q}] &= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n^2(1+\delta_Q)^2}\sum_{i=1}^{n}\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\bar{\mathbf{Q}}\mathbf{A}\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n^2(1+\delta_Q)^2}\sum_{i=1}^{n}\operatorname{Tr}(\Sigma_\beta\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\,\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n^2(1+\delta_Q)^2}\sum_{i=1}^{n}\operatorname{Tr}(\Sigma_\beta\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})\,\mathbb{E}[\mathbf{Q}_{-i}\Sigma_\beta\mathbf{Q}_{-i}] \\
&= \bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}} + \frac{1}{n}\frac{\operatorname{Tr}(\Sigma_\beta\bar{\mathbf{Q}}\mathbf{A}\bar{\mathbf{Q}})}{(1+\delta_Q)^2}\,\mathbb{E}[\mathbf{Q}\Sigma_\beta\mathbf{Q}]
\end{aligned}$$

Which finally concludes the proof. □

Now we will provide the result of a useful quantity that we will be using for computing the variance.

Lemma A.8 (Expectation of $\tilde{\boldsymbol{w}}^\top\mathbf{A}\tilde{\boldsymbol{w}}$)**.** *Let $\mathbf{A} \in \mathbb{R}^{p\times p}$ be a random matrix independent of $\tilde{\boldsymbol{w}}$. We have that:*

$$\mathbb{E}[\tilde{\boldsymbol{w}}^\top\mathbf{A}\tilde{\boldsymbol{w}}] = \frac{1}{(1+\delta_R)^2}\left(\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{R}\mathbf{A}\mathbf{R}]\boldsymbol{\mu} - \frac{2}{N(1+\delta_R)}\operatorname{Tr}(\Sigma\,\mathbb{E}[\mathbf{R}\mathbf{A}\mathbf{R}])\boldsymbol{\mu}^\top\bar{\mathbf{R}}\boldsymbol{\mu} + \frac{1}{N}\operatorname{Tr}(\Sigma\,\mathbb{E}[\mathbf{R}\mathbf{A}\mathbf{R}])\right)$$

Proof. We have that:

$$\begin{aligned}
\mathbb{E}[\tilde{\boldsymbol{w}}^\top\mathbf{A}\tilde{\boldsymbol{w}}] &= \frac{1}{N^2}\sum_{i,j=1}^{N}\mathbb{E}[\tilde{y}_i\tilde{y}_j\tilde{\boldsymbol{x}}_i^\top\mathbf{R}\mathbf{A}\mathbf{R}\tilde{\boldsymbol{x}}_j] \\
&= \frac{1}{N^2}\sum_{i\neq j}\mathbb{E}[\tilde{y}_i\tilde{y}_j\tilde{\boldsymbol{x}}_i^\top\mathbf{R}\mathbf{A}\mathbf{R}\tilde{\boldsymbol{x}}_j] + \frac{1}{N^2}\sum_{i=1}^{N}\mathbb{E}[\tilde{\boldsymbol{x}}_i^\top\mathbf{R}\mathbf{A}\mathbf{R}\tilde{\boldsymbol{x}}_i]
\end{aligned}$$

We have for $i \neq j$:

$$\begin{aligned}
&\mathbb{E}[\tilde{y}_i\tilde{y}_j\tilde{\boldsymbol{x}}_i^\top\mathbf{R}\mathbf{A}\mathbf{R}\tilde{\boldsymbol{x}}_j] = \frac{1}{(1+\delta_R)^2}\,\mathbb{E}[\tilde{y}_i\tilde{y}_j\tilde{\boldsymbol{x}}_i\mathbf{R}_{-i}\mathbf{A}\mathbf{R}_{-i}\tilde{\boldsymbol{x}}_j] \\
&= \frac{1}{(1+\delta_R)^2}\,\mathbb{E}\left[\tilde{y}_i\tilde{y}_j\tilde{\boldsymbol{x}}_i^\top\left(\mathbf{R}_{-ij} - \frac{\frac{1}{N}\mathbf{R}_{-ij}\tilde{\boldsymbol{x}}_j\tilde{\boldsymbol{x}}_j^\top\mathbf{R}_{-ij}}{1+\delta_R}\right)\mathbf{A}\left(\mathbf{R}_{-ij} - \frac{\frac{1}{N}\mathbf{R}_{-ij}\tilde{\boldsymbol{x}}_i\tilde{\boldsymbol{x}}_i^\top\mathbf{R}_{-ij}}{1+\delta_R}\right)\tilde{\boldsymbol{x}}_j\right] \\
&= A_{11} - A_{12} - A_{13} + A_{14}
\end{aligned}$$

So let us compute each term independently:

$$\begin{aligned}
A_{11} &= \frac{1}{(1+\delta_R)^2}\,\mathbb{E}[\tilde{y}_i\tilde{y}_j\tilde{\boldsymbol{x}}_i^\top\mathbf{R}_{-ij}\mathbf{A}\mathbf{R}_{-ij}\tilde{\boldsymbol{x}}_j] \\
&= \frac{1}{(1+\delta_R)^2}\boldsymbol{\mu}^\top\mathbb{E}[\mathbf{R}\mathbf{A}\mathbf{R}]\boldsymbol{\mu}
\end{aligned}$$

And :

$$A_{12} = \frac{1}{N(1+\delta_R)^3}\,\mathbb{E}[\tilde{y}_i\tilde{y}_j\tilde{\boldsymbol{x}}_i^\top\mathbf{R}_{-ij}\mathbf{A}\mathbf{R}_{-ij}\tilde{\boldsymbol{x}}_i\tilde{\boldsymbol{x}}_i^\top\mathbf{R}_{-ij}\tilde{\boldsymbol{x}}_j]$$

$$= \frac{1}{N(1+\delta_R)^3} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{RAR}])\, \mathbb{E}[\tilde{y}_i \tilde{y}_j \tilde{\boldsymbol{x}}_i^\top \mathbf{R}_{-ij} \tilde{\boldsymbol{x}}_j]$$
$$= \frac{1}{N(1+\delta_R)^3} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{RAR}]) \boldsymbol{\mu}^\top \bar{\mathbf{R}} \boldsymbol{\mu}$$

And also we can easily observe that:

$$A_{13} = A_{12}, \quad A_{14} = \mathscr{O}(N^{-1}).$$

Thus:

$$\mathbb{E}[\tilde{y}_i \tilde{y}_j \tilde{\boldsymbol{x}}_i^\top \mathbf{RAR} \tilde{\boldsymbol{x}}_j] = \frac{1}{(1+\delta_R)^2} \left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{RAR}] \boldsymbol{\mu} - \frac{2}{N(1+\delta_R)} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{RAR}]) \boldsymbol{\mu}^\top \bar{\mathbf{R}} \boldsymbol{\mu} \right)$$

And for the second term in the equation of $\mathbb{E}[\tilde{\boldsymbol{w}} \mathbf{A} \tilde{\boldsymbol{w}}]$, we have:

$$\begin{aligned}
\mathbb{E}[\tilde{\boldsymbol{x}}_i^\top \mathbf{RAR} \tilde{\boldsymbol{x}}_i] &= \frac{1}{(1+\delta_R)^2} \mathbb{E}[\tilde{\boldsymbol{x}}_i^\top \mathbf{R}_{-i} \mathbf{A} \mathbf{R}_{-i} \tilde{\boldsymbol{x}}_i] \\
&= \frac{1}{(1+\delta_R)^2} \mathbb{E}[\operatorname{Tr}(\tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^\top \mathbf{R}_{-i} \mathbf{A} \mathbf{R}_{-i})] \\
&= \frac{1}{(1+\delta_R)^2} \operatorname{Tr}(\mathbb{E}[\tilde{\boldsymbol{x}}_i \tilde{\boldsymbol{x}}_i^\top]\, \mathbb{E}[\mathbf{R}_{-i} \mathbf{A} \mathbf{R}_{-i}]) \\
&= \frac{1}{(1+\delta_R)^2} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{RAR}])
\end{aligned}$$

Hence, finally:

$$\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{A} \tilde{\boldsymbol{w}}] = \frac{1}{(1+\delta_R)^2} \left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{RAR}] \boldsymbol{\mu} - \frac{2}{N(1+\delta_R)} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{RAR}]) \boldsymbol{\mu}^\top \bar{\mathbf{R}} \boldsymbol{\mu} + \frac{1}{N} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{RAR}]) \right)$$

□

Lemma A.9 (Commutativity)**.** *Let* $\bar{\mathbf{R}}$ *and* $\bar{\mathbf{Q}}$ *be the resolvent matrices defined in lemma 3.6. We have that:*

$$\bar{\mathbf{Q}} \Sigma_\beta = \Sigma_\beta \bar{\mathbf{Q}}, \quad \bar{\mathbf{R}} \Sigma = \Sigma \bar{\mathbf{R}}.$$

Proof. We will just prove it for $\bar{\mathbf{Q}}$ and Σ_β because the other proof of the second identity is similar. We know that:

$$\Sigma_\beta = (1+\delta_Q)(\bar{\mathbf{Q}}^{-1} - \gamma \mathbf{I}_p)$$

Thus:

$$\bar{\mathbf{Q}} \Sigma_\beta = (1+\delta_Q) \bar{\mathbf{Q}} (\bar{\mathbf{Q}}^{-1} - \gamma \mathbf{I}_p) = (1+\delta_Q)(\mathbf{I}_p - \gamma \bar{\mathbf{Q}})$$
$$\Sigma_\beta \bar{\mathbf{Q}} = (1+\delta_Q)(\bar{\mathbf{Q}}^{-1} - \gamma \mathbf{I}_p) \bar{\mathbf{Q}} = (1+\delta_Q)(\mathbf{I}_p - \gamma \bar{\mathbf{Q}})$$

which concludes the proof. □

B RMT Analysis of the fine-tuned classifier

Let $\boldsymbol{x} \sim \mathscr{N}((-1)^a \boldsymbol{\mu}_\beta, \mathbf{I}_p)$ independent of the fine-tuning dataset $\mathbf{X}$. We recall that:

$$\boldsymbol{w}_\alpha = \boldsymbol{w} + \alpha \tilde{\boldsymbol{w}} - \frac{\alpha}{n} \mathbf{Q}(\gamma) \mathbf{X} \mathbf{X}^\top \tilde{\boldsymbol{w}},$$

where:

$$\boldsymbol{w} = \frac{1}{n} \mathbf{Q}(\gamma) \mathbf{X} \boldsymbol{y}, \quad \tilde{\boldsymbol{w}} = \frac{1}{N} \mathbf{R}(\tilde{\gamma}) \tilde{\mathbf{X}} \tilde{\boldsymbol{y}}$$

B.1 Test Expectation

We have that:

$$\mathbb{E}[\boldsymbol{w}_\alpha^\top \boldsymbol{x}] = \mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}] + \alpha \, \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] - \frac{\alpha}{n} \, \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X} \mathbf{X}^\top \mathbf{Q} \boldsymbol{x}] \tag{26}$$

Let us compute each term of this previous sum.

First, using lemma A.6, we have that, since $\boldsymbol{x}$ is independent of $\mathbf{X}$ and of $\tilde{\mathbf{X}}$:

$$\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}] = \mathbb{E}[\boldsymbol{w}]^\top \, \mathbb{E}[\boldsymbol{x}] = \frac{(-1)^a}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta$$

$$\mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] = \mathbb{E}[\tilde{\boldsymbol{w}}]^\top \, \mathbb{E}[\boldsymbol{x}] = \frac{(-1)^a}{1+\delta_R} \boldsymbol{\mu}^\top \bar{\mathbf{R}} \boldsymbol{\mu}_\beta$$

And we have that:

$$\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X} \mathbf{X}^\top \mathbf{Q} \boldsymbol{x}] = \mathbb{E}[\tilde{\boldsymbol{w}}]^\top \, \mathbb{E}[\mathbf{X} \mathbf{X}^\top \mathbf{Q}] \, \mathbb{E}[\boldsymbol{x}]$$

And:

$$\begin{aligned}
\mathbb{E}[\mathbf{X} \mathbf{X}^\top \mathbf{Q}] &= \sum_{i=1}^n \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}] \\
&= \sum_{i=1}^n \frac{1}{1+\delta_Q} \, \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_i] \\
&= \sum_{i=1}^n \frac{1}{1+\delta_Q} \, \mathbb{E}[\boldsymbol{x}_i \boldsymbol{x}_i^\top] \bar{\mathbf{Q}} \\
&= \frac{n}{1+\delta_Q} \Sigma_\beta \bar{\mathbf{Q}}
\end{aligned}$$

Thus:

$$\begin{aligned}
\frac{1}{n} \, \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X} \mathbf{X}^\top \mathbf{Q} \boldsymbol{x}] &= \frac{(-1)^a}{(1+\delta_R)} \frac{1}{(1+\delta_Q)} \boldsymbol{\mu}^\top \bar{\mathbf{R}} \Sigma_\beta \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \\
&= \frac{(-1)^a}{1+\delta_R} \boldsymbol{\mu}^\top \bar{\mathbf{R}} (\mathbf{I}_p - \gamma \bar{\mathbf{Q}}) \boldsymbol{\mu}_\beta
\end{aligned}$$

Finally:

$$\mathbb{E}[\boldsymbol{w}_\alpha^\top \boldsymbol{x}] = (-1)^a \left(\frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta + \frac{\alpha}{1+\delta_R} \boldsymbol{\mu}^\top \bar{\mathbf{R}} \boldsymbol{\mu}_\beta - \frac{\alpha}{1+\delta_R} \boldsymbol{\mu}^\top \bar{\mathbf{R}} (\mathbf{I}_p - \gamma \bar{\mathbf{Q}}) \boldsymbol{\mu}_\beta \right)$$

$$= (-1)^a \left(\frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta + \frac{\alpha\gamma}{1+\delta_R} \boldsymbol{\mu}^\top \bar{\mathbf{R}} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right)$$

And using the identities in lemma A.5:

$$\mathbb{E}[\boldsymbol{w}_\alpha^\top \boldsymbol{x}] = \frac{(-1)^a}{(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))} \left(\|\boldsymbol{\mu}_\beta\|^2 + \frac{\alpha\gamma(1+\delta_Q)}{(\|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R))} \beta \|\boldsymbol{\mu}\|^2 \right) \tag{27}$$

$$= \frac{(-1)^a}{\lambda_Q} \left(\|\boldsymbol{\mu}_\beta\|^2 + \frac{\alpha\beta\gamma(1+\delta_Q)}{\lambda_R} \|\boldsymbol{\mu}\|^2 \right) \tag{28}$$

B.2 Test Variance

To compute the variance of $\boldsymbol{w}_\alpha^\top \boldsymbol{x}$, it suffices to compute the second moment: $\mathbb{E}[(\boldsymbol{w}_\alpha^\top \boldsymbol{x})^2]$.

$$\mathbb{E}[(\boldsymbol{w}_\alpha^\top \boldsymbol{x})^2] = \mathbb{E}[(\boldsymbol{w}^\top \boldsymbol{x} + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2 + \frac{\alpha^2}{n^2} (\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2 - \frac{2\alpha}{n} \tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}(\boldsymbol{w}^\top \boldsymbol{x} + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{x})] \tag{29}$$

First term: We have that, as proved in (Firdoussi & Seddik, 2024):

$$\begin{aligned}
\mathbb{E}[(\boldsymbol{w}^\top \boldsymbol{x})^2] &= \frac{1}{h(1+\delta_Q)} \left(\frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\Sigma}_\beta \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - 2(1-h) \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right) + \frac{1-h}{h} \\
&= \frac{1}{h(1+\delta_Q)} \left(\frac{1}{1+\delta_Q} \left((\boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta)^2 + \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}}^2 \boldsymbol{\mu}_\beta \right) - 2(1-h) \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right) + \frac{1-h}{h} \\
&= \frac{\|\boldsymbol{\mu}_\beta\|^2}{h(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))} \left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)} - 2(1-h) \right) + \frac{1-h}{h}
\end{aligned}$$

And:

$$\begin{aligned}
\mathbb{E}[(\tilde{\boldsymbol{w}}^\top x)^2] &= \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x} \tilde{\boldsymbol{w}}^\top \boldsymbol{x}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x} \boldsymbol{x}^\top \tilde{\boldsymbol{w}}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{\Sigma}_\beta \tilde{\boldsymbol{w}}]
\end{aligned}$$

Therefore by lemma A.8:

$$\mathbb{E}[(\tilde{\boldsymbol{w}}^\top x)^2] = \frac{1}{(1+\delta_R)^2} \left(\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{R}\boldsymbol{\Sigma}_\beta \mathbf{R}] \boldsymbol{\mu} - \frac{2}{(1+\delta_R)} \frac{1}{N} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{R}\boldsymbol{\Sigma}_\beta \mathbf{R}]) \boldsymbol{\mu}^\top \bar{\mathbf{R}} \boldsymbol{\mu} + \frac{1}{N} \operatorname{Tr}(\Sigma\, \mathbb{E}[\mathbf{R}\boldsymbol{\Sigma}_\beta \mathbf{R}]) \right) \tag{30}$$

And, we have that:

$$\begin{aligned}
\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x} \tilde{\boldsymbol{w}}^\top \boldsymbol{x}] &= \mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x} \boldsymbol{x}^\top \tilde{\boldsymbol{w}}] \\
&= \mathbb{E}[\boldsymbol{w}]^\top \boldsymbol{\Sigma}_\beta\, \mathbb{E}[\tilde{\boldsymbol{w}}] \\
&= \frac{1}{(1+\delta_Q)(1+\delta_R)} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\Sigma}_\beta \bar{\mathbf{R}} \boldsymbol{\mu} \\
&= \frac{1}{(1+\delta_R)} \boldsymbol{\mu}_\beta^\top (\mathbf{I}_p - \gamma \bar{\mathbf{Q}}) \bar{\mathbf{R}} \boldsymbol{\mu} \\
&= \frac{1}{(1+\delta_R)} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{R}} \boldsymbol{\mu} - \frac{\gamma}{(1+\delta_R)} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \bar{\mathbf{R}} \boldsymbol{\mu}
\end{aligned}$$

And since $\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] = \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x}\boldsymbol{w}^\top \boldsymbol{x}]$, then:

$$\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] = \frac{1}{(1+\delta_R)}\boldsymbol{\mu}_\beta^\top \bar{\mathbf{R}}\boldsymbol{\mu} - \frac{\gamma}{(1+\delta_R)}\boldsymbol{\mu}_\beta^\top \bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu}$$

and thus:

$$\boldsymbol{\mu}_\beta^\top \bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu} = \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}}\bar{\mathbf{R}}\boldsymbol{\mu} \tag{31}$$

Second term: Now let us compute the expectation of the second term in (41):

$$\begin{aligned}
\frac{1}{n^2}\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2] &= \frac{1}{n^2}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}] \\
&= \frac{1}{n^2}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\tilde{\boldsymbol{w}}] \\
&= \frac{1}{n^2}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\Sigma_\beta \mathbf{X}\mathbf{X}^\top \mathbf{Q}\tilde{\boldsymbol{w}}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}^\top (\mathbf{I}_p - \gamma\mathbf{Q})\Sigma_\beta(\mathbf{I}_p - \gamma\mathbf{Q})\tilde{\boldsymbol{w}}]
\end{aligned}$$

Therefore, by lemma A.8:

$$\begin{aligned}
\frac{1}{n^2}\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2] &= \frac{1}{(1+\delta_R)^2}\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{R}(\mathbf{I}_p - \gamma\mathbf{Q})\Sigma_\beta(\mathbf{I}_p - \gamma\mathbf{Q})\mathbf{R}]\boldsymbol{\mu} \\
&+ \frac{\mathrm{Tr}(\Sigma\,\mathbb{E}[\mathbf{R}(\mathbf{I}_p - \gamma\mathbf{Q})\Sigma_\beta(\mathbf{I}_p - \gamma\mathbf{Q})\mathbf{R}])}{N(1+\delta_R)^2}\left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top \bar{\mathbf{R}}\boldsymbol{\mu}\right)
\end{aligned}$$

Third term: Now we want to compute $\frac{2\alpha}{n}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}(\boldsymbol{w}^\top \boldsymbol{x} + \alpha\tilde{\boldsymbol{w}}^\top \boldsymbol{x})]$. So we have that:

$$\begin{aligned}
\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{w}^\top \boldsymbol{x}] &= \mathbb{E}[\tilde{\boldsymbol{w}}]^\top \mathbb{E}[\mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{w}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}]^\top \mathbb{E}[\mathbf{X}\mathbf{X}^\top \mathbf{Q}\Sigma_\beta \boldsymbol{w}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}]^\top \mathbb{E}[\frac{1}{n}\mathbf{X}\mathbf{X}^\top \mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{X}\boldsymbol{y}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}]^\top \mathbb{E}[(\mathbf{Q}^{-1} - \gamma\mathbf{I}_p)\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{X}\boldsymbol{y}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}]^\top \mathbb{E}[(\mathbf{I}_p - \gamma\mathbf{Q})\Sigma_\beta \mathbf{Q}\mathbf{X}\boldsymbol{y}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}]^\top (\mathbb{E}[\Sigma_\beta \mathbf{Q}\mathbf{X}\boldsymbol{y}] - \gamma\,\mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{X}\boldsymbol{y}])
\end{aligned}$$

And we have that:

$$\begin{aligned}
\mathbb{E}[\Sigma_\beta \mathbf{Q}\mathbf{X}\boldsymbol{y}] &= \sum_{i=1}^n \mathbb{E}[y_i \Sigma_\beta \mathbf{Q}\boldsymbol{x}_i] \\
&= \frac{n}{(1+\delta_Q)}\mathbb{E}[y_i \Sigma_\beta \mathbf{Q}_{-i}\boldsymbol{x}_i] \\
&= \frac{n}{(1+\delta_Q)}\Sigma_\beta \bar{\mathbf{Q}}\boldsymbol{\mu}_\beta \\
&= n(\mathbf{I}_p - \gamma\bar{\mathbf{Q}})\boldsymbol{\mu}_\beta
\end{aligned}$$

And:

$$\mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{X}\boldsymbol{y}] = \sum_{i=1}^n \mathbb{E}[y_i \mathbf{Q}\Sigma_\beta \mathbf{Q}\boldsymbol{x}_i]$$

$$
\begin{aligned}
&= \frac{n}{(1+\delta_Q)}\,\mathbb{E}[y_i \mathbf{Q}\Sigma_\beta \mathbf{Q}_{-i}\boldsymbol{x}_i] \\
&= \frac{n}{(1+\delta_Q)}\,\mathbb{E}\left[y_i\left(\mathbf{Q}_{-i} - \frac{\frac{1}{n}\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top \mathbf{Q}_{-i}}{1+\delta_Q}\right)\Sigma_\beta \mathbf{Q}_{-i}\boldsymbol{x}_i\right] \\
&= \frac{n}{(1+\delta_Q)}\left(\mathbb{E}[y_i \mathbf{Q}_{-i}\Sigma_\beta \mathbf{Q}_{-i}\boldsymbol{x}_i] - \frac{1}{n(1+\delta_Q)}\,\mathbb{E}[y_i \mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top \mathbf{Q}_{-i}\Sigma_\beta \mathbf{Q}_{-i}\boldsymbol{x}_i]\right) \\
&= \frac{n}{(1+\delta_Q)}\left(\mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}]\boldsymbol{\mu}_\beta - \frac{1}{n(1+\delta_Q)}\operatorname{Tr}(\Sigma_\beta\,\mathbb{E}[\mathbf{Q}\Sigma_\beta\mathbf{Q}])\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta\right) \\
&= \frac{n}{h(1+\delta_Q)}\bar{\mathbf{Q}}\Sigma_\beta\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta - \frac{n(1-h)}{h}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta \\
&= n\left(\frac{1}{h}(\mathbf{I}_p - \gamma\bar{\mathbf{Q}})\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta - \frac{1-h}{h}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta\right) \\
&= n(\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta - \frac{\gamma}{h}\bar{\mathbf{Q}}^2\boldsymbol{\mu}_\beta)
\end{aligned}
$$

Thus:

$$
\frac{1}{n}\,\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{w}^\top\boldsymbol{x}] = \frac{1}{(1+\delta_R)}\boldsymbol{\mu}^\top\bar{\mathbf{R}}\left(\mathbf{I}_p - 2\gamma\bar{\mathbf{Q}} + \frac{\gamma^2}{h}\bar{\mathbf{Q}}^2\right)\boldsymbol{\mu}_\beta \tag{32}
$$

Let us now compute the remaining term:

$$
\begin{aligned}
\frac{1}{n}\,\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\tilde{\boldsymbol{w}}^\top\boldsymbol{x}] &= \frac{1}{n}\,\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}^\top\tilde{\boldsymbol{w}}] \\
&= \frac{1}{n}\,\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\Sigma_\beta\tilde{\boldsymbol{w}}] \\
&= \mathbb{E}[\tilde{\boldsymbol{w}}^\top(\mathbf{I}_p - \gamma\mathbf{Q})\Sigma_\beta\tilde{\boldsymbol{w}}]
\end{aligned}
$$

And again by lemma A.8:

$$
\frac{1}{n}\,\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\tilde{\boldsymbol{w}}^\top\boldsymbol{x}] = \frac{1}{(1+\delta_R)^2}\boldsymbol{\mu}^\top\,\mathbb{E}[\mathbf{R}(\mathbf{I}_p - \gamma\mathbf{Q})\Sigma_\beta\mathbf{R}]\boldsymbol{\mu} + \frac{\operatorname{Tr}(\Sigma\,\mathbb{E}[\mathbf{R}(\mathbf{I}_p - \gamma\mathbf{Q})\Sigma_\beta\mathbf{R}])}{N(1+\delta_R)^2}\left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top\bar{\mathbf{R}}\boldsymbol{\mu}\right)
$$

Now let us group all the results as follows.

Terms without α: There is only one term which is:

$$
\begin{aligned}
T_1 = \mathbb{E}[(\boldsymbol{w}^\top\boldsymbol{x})^2] &= \frac{1}{h(1+\delta_Q)}\left((2h-1)\boldsymbol{\mu}_\beta^\top\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta - \gamma\boldsymbol{\mu}_\beta^\top\bar{\mathbf{Q}}^2\boldsymbol{\mu}_\beta\right) + \frac{1-h}{h} \\
&= \frac{\|\boldsymbol{\mu}_\beta\|^2}{h(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)} - 2(1-h)\right) + \frac{1-h}{h}
\end{aligned}
$$

Terms in α: There are two: $2\,\mathbb{E}[\boldsymbol{w}^\top\boldsymbol{x}\tilde{\boldsymbol{w}}^\top\boldsymbol{x}]$ and $\frac{2}{n}\,\mathbb{E}[\tilde{\boldsymbol{w}}^\top\mathbf{X}\mathbf{X}^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{w}^\top\boldsymbol{x}]$:

$$
\begin{aligned}
T_2 &= 2\,\mathbb{E}[\boldsymbol{w}^\top\boldsymbol{x}\tilde{\boldsymbol{w}}^\top\boldsymbol{x}] - \frac{2}{n}\,\mathbb{E}[\tilde{\boldsymbol{w}}^\top\mathbf{X}\mathbf{X}^\top\mathbf{Q}\boldsymbol{x}\boldsymbol{w}^\top\boldsymbol{x}] \\
&= \frac{2}{(1+\delta_R)}\left(\boldsymbol{\mu}_\beta\bar{\mathbf{R}}\boldsymbol{\mu} - \gamma\boldsymbol{\mu}_\beta^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu} - \boldsymbol{\mu}^\top\bar{\mathbf{R}}(\mathbf{I}_p - 2\gamma\bar{\mathbf{Q}} + \frac{\gamma^2}{h}\bar{\mathbf{Q}}^2)\boldsymbol{\mu}_\beta\right) \\
&= \frac{2\gamma}{(1+\delta_R)}\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\left(\mathbf{I}_p - \frac{\gamma}{h}\bar{\mathbf{Q}}\right)\boldsymbol{\mu}_\beta
\end{aligned}
$$

And using lemma A.5:

$$
T_2 = \frac{2\gamma(1+\delta_Q)\beta\|\boldsymbol{\mu}\|^2}{(\|\boldsymbol{\mu}\|^2 + 1 + \tilde{\gamma}(1+\delta_R))\,(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))}\left(1 - \frac{\gamma(1+\delta_Q)}{h(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))}\right)
$$

Terms in α^2 : we have three terms: $\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2]$, $\frac{1}{n^2}\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2]$ and $\frac{-2}{n}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\tilde{\boldsymbol{w}}^\top \boldsymbol{x}]$:

$$\begin{aligned}
T_3 &= \mathbb{E}[(\tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2] + \frac{1}{n^2}\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2] - \frac{2}{n}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] \\
&= \frac{\gamma}{(1+\delta_R)^2}\boldsymbol{\mu}^\top \left(\mathbb{E}[\mathbf{R}\bar{\mathbf{Q}}\Sigma_\beta \mathbf{R}] - \mathbb{E}[\mathbf{R}\Sigma_\beta \bar{\mathbf{Q}}\mathbf{R}] + \gamma\,\mathbb{E}[\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}]\right)\boldsymbol{\mu} \\
&+ \frac{\gamma}{N(1+\delta_R)^2}\left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top \bar{\mathbf{R}}\boldsymbol{\mu}\right)\operatorname{Tr}\left(\Sigma(\mathbb{E}[\mathbf{R}\bar{\mathbf{Q}}\Sigma_\beta \mathbf{R}] - \mathbb{E}[\mathbf{R}\Sigma_\beta \bar{\mathbf{Q}}\mathbf{R}] + \gamma\,\mathbb{E}[\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}])\right) \\
&= \frac{\gamma^2}{(1+\delta_R)^2}\left[\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}]\boldsymbol{\mu} + \left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top \bar{\mathbf{R}}\boldsymbol{\mu}\right)\frac{1}{N}\operatorname{Tr}(\Sigma\,\mathbb{E}[\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}])\right]
\end{aligned}$$

where the last equality is gotten using lemma A.9.
We also have that:

$$\begin{aligned}
\frac{1}{N}\operatorname{Tr}(\Sigma\,\mathbb{E}[\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}]) &= \frac{1}{N}\operatorname{Tr}(\mathbb{E}[\Sigma\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}]) \\
&= \frac{1}{N}\mathbb{E}[\operatorname{Tr}(\Sigma\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R})] \\
&= \frac{1}{N}\mathbb{E}[\operatorname{Tr}(\mathbf{R}\Sigma\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q})] \\
&= \frac{1}{N}\operatorname{Tr}(\mathbb{E}[\mathbf{R}\Sigma\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}]) \\
&= \frac{1}{N}\operatorname{Tr}(\mathbb{E}[\mathbf{R}\Sigma\mathbf{R}]\,\mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}]) \\
&= \frac{1}{h\tilde{h}}\frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}\bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}})
\end{aligned}$$

And:

$$\begin{aligned}
\boldsymbol{\mu}^\top \mathbb{E}[\mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}]\boldsymbol{\mu} &= \operatorname{Tr}(\mathbb{E}[\boldsymbol{\mu}^\top \mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q}\mathbf{R}\boldsymbol{\mu}]) \\
&= \mathbb{E}[\operatorname{Tr}(\mathbf{R}\boldsymbol{\mu}\boldsymbol{\mu}^\top \mathbf{R}\mathbf{Q}\Sigma_\beta \mathbf{Q})] \\
&= \operatorname{Tr}(\mathbb{E}[\mathbf{R}\boldsymbol{\mu}\boldsymbol{\mu}^\top \mathbf{R}]\,\mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}]) \\
&= \frac{1}{h}\operatorname{Tr}(\mathbb{E}[\mathbf{R}\boldsymbol{\mu}\boldsymbol{\mu}^\top \mathbf{R}]\bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}})
\end{aligned}$$

Thus:

$$T_3 = \frac{\gamma^2}{h(1+\delta_R)^2}\left[\operatorname{Tr}(\mathbb{E}[\mathbf{R}\boldsymbol{\mu}\boldsymbol{\mu}^\top \mathbf{R}]\bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}}) + \left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top \bar{\mathbf{R}}\boldsymbol{\mu}\right)\frac{1}{\tilde{h}}\frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}\bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}})\right]$$

Now remains to compute $\mathbb{E}[\mathbf{R}\boldsymbol{\mu}\boldsymbol{\mu}^\top \mathbf{R}]$. For that, we use lemma A.7:

$$\begin{aligned}
\mathbb{E}[\mathbf{R}\boldsymbol{\mu}\boldsymbol{\mu}^\top \mathbf{R}] &= \bar{\mathbf{R}}\boldsymbol{\mu}\boldsymbol{\mu}^\top \bar{\mathbf{R}} + \frac{1}{N}\frac{\operatorname{Tr}(\Sigma\bar{\mathbf{R}}\boldsymbol{\mu}\boldsymbol{\mu}^\top \bar{\mathbf{R}})}{(1+\delta_R)^2}\mathbb{E}[\mathbf{R}\Sigma\mathbf{R}] \\
&= \bar{\mathbf{R}}\boldsymbol{\mu}\boldsymbol{\mu}^\top \bar{\mathbf{R}} + \frac{1}{N}\frac{\boldsymbol{\mu}^\top \bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}\boldsymbol{\mu}}{(1+\delta_R)^2}\frac{1}{\tilde{h}}\bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}
\end{aligned}$$

And since we are in the regime of $N \to \infty$, then:

$$\frac{1}{N}\boldsymbol{\mu}^\top \bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}\boldsymbol{\mu} = \mathcal{O}(N^{-1})$$

Thus:

$$\mathbb{E}[\mathbf{R}\boldsymbol{\mu}\boldsymbol{\mu}^\top\mathbf{R}] = \bar{\mathbf{R}}\boldsymbol{\mu}\boldsymbol{\mu}^\top\bar{\mathbf{R}} \tag{33}$$

Hence, T_3 becomes:

$$T_3 = \frac{\gamma^2}{h(1+\delta_R)^2}\left[\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\Sigma_\beta\bar{\mathbf{Q}}\bar{\mathbf{R}}\boldsymbol{\mu} + \left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top\bar{\mathbf{R}}\boldsymbol{\mu}\right)\frac{1}{\tilde{h}}\frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}\bar{\mathbf{Q}}\Sigma_\beta\bar{\mathbf{Q}})\right]$$

And we also have that:

$$\begin{aligned}\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\Sigma_\beta\bar{\mathbf{Q}}\bar{\mathbf{R}}\boldsymbol{\mu} &= \boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta\boldsymbol{\mu}_\beta^\top\bar{\mathbf{Q}}\bar{\mathbf{R}}\boldsymbol{\mu} + \boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}^2\bar{\mathbf{R}}\boldsymbol{\mu} \\ &= \left(\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta\right)^2 + \boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}^2\bar{\mathbf{R}}\boldsymbol{\mu}\end{aligned}$$

And:

$$\frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}\Sigma\bar{\mathbf{R}}\bar{\mathbf{Q}}\Sigma_\beta\bar{\mathbf{Q}}) = \frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}^2\bar{\mathbf{Q}}^2)$$

Therefore:

$$T_3 = \frac{\gamma^2}{h(1+\delta_R)^2}\left[\left(\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta\right)^2 + \boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}^2\bar{\mathbf{R}}\boldsymbol{\mu} + \left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top\bar{\mathbf{R}}\boldsymbol{\mu}\right)\frac{1}{\tilde{h}}\frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}^2\bar{\mathbf{Q}}^2)\right] \tag{34}$$

Then using lemmas A.4 and A.5:

$$\begin{aligned}T_3 &= \frac{\gamma^2}{h(1+\delta_R)^2}\left[\left(\boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}\boldsymbol{\mu}_\beta\right)^2 + \boldsymbol{\mu}^\top\bar{\mathbf{R}}\bar{\mathbf{Q}}^2\bar{\mathbf{R}}\boldsymbol{\mu}\right] + \frac{\gamma^2}{h(1+\delta_R)^2}\left(1 - \frac{2}{(1+\delta_R)}\boldsymbol{\mu}^\top\bar{\mathbf{R}}\boldsymbol{\mu}\right)\frac{1}{\tilde{h}}\frac{1}{N}\operatorname{Tr}(\bar{\mathbf{R}}^2\bar{\mathbf{Q}}^2) \\ &= \frac{\gamma^2(1+\delta_Q)^2}{h}\Bigg[\frac{\|\boldsymbol{\mu}\|^2}{\lambda_R^2}\left(\frac{\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q^2} + \frac{1}{(1+\gamma(1+\delta_Q))^2}\left(1 + \frac{\beta^2\|\boldsymbol{\mu}\|^2\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q^2} - \frac{2\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q}\right)\right) + \\ &\frac{\tilde{\eta}}{(1+\gamma(1+\delta_Q))^2(1+\tilde{\gamma}(1+\delta_R))^2}\left(1 - \frac{2\|\boldsymbol{\mu}\|^2}{\lambda_R}\right)\Bigg] \\ &= \frac{\gamma^2(1+\delta_Q)^2}{h}\left[\frac{\|\boldsymbol{\mu}\|^2}{\lambda_R^2}\left(\frac{\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q^2} + \frac{1-h}{\eta}\left(1 + \frac{\beta^2\|\boldsymbol{\mu}\|^2\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q^2} - \frac{2\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q} + (1-\tilde{h})\left(1 - \frac{2\|\boldsymbol{\mu}\|^2}{\lambda_R}\right)\right)\right)\right]\end{aligned}$$

Finally:

$$T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\lambda_Q} - 2(1-h)\right) + \frac{1-h}{h} \tag{35}$$

$$T_2 = \frac{2\gamma\beta(1+\delta_Q)\|\boldsymbol{\mu}\|^2}{\lambda_R\lambda_Q}\left(1 - \frac{\gamma(1+\delta_Q)}{h\lambda_Q}\right) \tag{36}$$

$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\left[\frac{\|\boldsymbol{\mu}\|^2}{\lambda_R^2}\left(\frac{\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q^2} + \frac{1-h}{\eta}\left(1 + \frac{\beta^2\|\boldsymbol{\mu}\|^2\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q^2} - \frac{2\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q} + (1-\tilde{h})\left(1 - \frac{2\|\boldsymbol{\mu}\|^2}{\lambda_R}\right)\right)\right)\right] \tag{37}$$

And the expression of the second order expectation reads:

$$\mathbb{E}[(\boldsymbol{w}_\alpha^\top\boldsymbol{x})^2] = T_1 + \alpha T_2 + \alpha^2 T_3 \tag{38}$$

And finally, Theorem 4.2 follows:

Theorem B.1 (Gaussianity of the fine-tuned Ridge model)**.** *Let $\boldsymbol{w}_\alpha$ be the fine-tuned classifier as defined in (α-FTC) and suppose that Assumption 4.1 holds. The decision function $\boldsymbol{w}_\alpha^\top \boldsymbol{x}$, on some test sample $\boldsymbol{x} \in \mathscr{C}_a$ independent of $\mathbf{X}$, satisfies:*

$$\boldsymbol{w}_\alpha^\top \boldsymbol{x} \xrightarrow{\mathscr{D}} \mathcal{N}\left((-1)^a m_\alpha,\, \nu_\alpha - m_\alpha^2\right),$$

where:

$$m_\alpha = \frac{1}{\lambda_Q}\left(\|\boldsymbol{\mu}_\beta\|^2 + \frac{\alpha\beta\gamma(1+\delta_Q)}{\lambda_R}\|\boldsymbol{\mu}\|^2\right),$$
$$\nu_\alpha = T_1 + \alpha T_2 + \alpha^2 T_3.$$

With:

$$T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q} - 2(1-h)\right) + \frac{1-h}{h},$$
$$T_2 = \frac{2\gamma\beta(1+\delta_Q)\|\boldsymbol{\mu}\|^2}{\lambda_R\lambda_Q}\left(1 - \frac{\gamma(1+\delta_Q)}{h\lambda_Q}\right),$$
$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\times$$
$$\left[\frac{\|\boldsymbol{\mu}\|^2}{\lambda_R^2}\left(\frac{\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q^2} + \frac{1-h}{\eta}\left(1 + \frac{\beta^2\|\boldsymbol{\mu}\|^2\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q^2} - \frac{2\beta^2\|\boldsymbol{\mu}\|^2}{\lambda_Q}\right)\right) + \frac{(1-h)(1-\tilde{h})}{\eta}\left(1 - \frac{2\|\boldsymbol{\mu}\|^2}{\lambda_R}\right)\right].$$

B.3 Finding optimal scaling parameter

Since the test accuracy is given by $\mathscr{A}_{\text{test}} = 1 - \varphi\left((\nu_\alpha - m_\alpha^2)^{-\frac{1}{2}} m_\alpha\right)$ as in Proposition 4.3, and that $\phi(x)$ is a non-increasing function, then finding the optimal α^* that maximizes the test accuracy boils down to maximizing the term inside ϕ. Thus, by computing the derivative with respect to α of $(\nu_\alpha - m_\alpha^2)^{-\frac{1}{2}} m_\alpha$ and finding the zero of the gradient gives us the final form of the best scaling parameter α^*:

$$\alpha^* = \frac{\lambda_R T_2\|\boldsymbol{\mu}_\beta\|^2 - 2\beta\gamma T_1(1+\delta_Q)\|\boldsymbol{\mu}\|^2}{\beta\gamma T_2(1+\delta_Q)\|\boldsymbol{\mu}\|^2 - 2\lambda_R T_3\|\boldsymbol{\mu}_\beta\|^2}$$

And since the worst test accuracy is 50% (random classification), which is obtained for $m_\alpha = 0$, then solving the previous equation gives the worst scaling $\bar{\alpha}$ to use:

$$\bar{\alpha} = -\frac{\lambda_R\|\boldsymbol{\mu}_\beta\|^2}{\beta\gamma(1+\delta_Q)\|\boldsymbol{\mu}\|^2}$$

C RMT analysis for arbitrary source classifier

Let $\boldsymbol{x} \sim \mathcal{N}((-1)^a\boldsymbol{\mu}_\beta, \mathbf{I}_p)$ be an independent test sample. Let $\tilde{\boldsymbol{w}}$ be the source classifier (obtained through some optimization algorithm). We recall that:

$$\boldsymbol{w}_\alpha = \boldsymbol{w} + \alpha\tilde{\boldsymbol{w}} - \frac{\alpha}{n}\mathbf{Q}(\gamma)\mathbf{X}\mathbf{X}^\top\tilde{\boldsymbol{w}}, \quad \boldsymbol{w} = \frac{1}{n}\mathbf{Q}(\gamma)\mathbf{X}\boldsymbol{y}$$

C.1 Test Expectation

We have that:

$$\mathbb{E}[\boldsymbol{w}_\alpha^\top \boldsymbol{x}] = \mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}] + \alpha\, \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] - \frac{\alpha}{n}\, \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}] \tag{39}$$

Let us compute each term of this previous sum.
First, using lemma A.6, we have that, since $\boldsymbol{x}$ is independent of $\mathbf{X}$:

$$\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}] = \mathbb{E}[\boldsymbol{w}]^\top\, \mathbb{E}[\boldsymbol{x}] = \frac{(-1)^a}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta$$

And we have that:

$$\mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] = (-1)^a \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta$$

And:

$$\begin{aligned}
\frac{\alpha}{n}\, \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}] &= \frac{\alpha}{n} \sum_{i=1}^n \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}\boldsymbol{x}] \\
&= \frac{\alpha}{n(1+\delta_Q)} \sum_{i=1}^n \mathbb{E}[\tilde{\boldsymbol{w}}^\top \boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i}\boldsymbol{x}] \\
&= \frac{\alpha}{n(1+\delta_Q)} \sum_{i=1}^n \mathbb{E}[\tilde{\boldsymbol{w}}^\top \Sigma_\beta \mathbf{Q}_{-i}\boldsymbol{x}] \\
&= \frac{(-1)^a \alpha}{1+\delta_Q} \tilde{\boldsymbol{w}}^\top \Sigma_\beta \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta
\end{aligned}$$

Thus:

$$\begin{aligned}
\mathbb{E}[\boldsymbol{w}_\alpha^\top \boldsymbol{x}] &= (-1)^a \left(\frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta - \frac{\alpha}{1+\delta_Q} \tilde{\boldsymbol{w}}^\top \Sigma_\beta \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right) \\
&= (-1)^a \left(\frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta - \alpha \tilde{\boldsymbol{w}}^\top (\bar{\mathbf{Q}}^{-1} - \gamma \mathbf{I}_p) \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right) \\
&= (-1)^a \left(\frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta + \alpha\gamma \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right)
\end{aligned}$$

Using the forumlas in lemma A.5:

$$\mathbb{E}[\boldsymbol{w}_\alpha^\top \boldsymbol{x}] = \frac{(-1)^a}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)} \left(\|\boldsymbol{\mu}_\beta\|^2 + \alpha\gamma(1+\delta_Q) \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta \right) \tag{40}$$

C.2 Test variance

To compute the variance of $\boldsymbol{w}_\alpha^\top \boldsymbol{x}$, it suffices to compute the second moment: $\mathbb{E}[(\boldsymbol{w}_\alpha^\top \boldsymbol{x})^2]$.

$$\mathbb{E}[(\boldsymbol{w}_\alpha^\top \boldsymbol{x})^2] = \mathbb{E}[(\boldsymbol{w}^\top \boldsymbol{x} + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2 + \frac{\alpha^2}{n^2} (\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2 - \frac{2\alpha}{n} \tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x} (\boldsymbol{w}^\top \boldsymbol{x} + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{x})] \tag{41}$$

First term: We start by computing

$$\mathbb{E}[(\boldsymbol{w}^\top \boldsymbol{x} + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2] = \mathbb{E}[(\boldsymbol{w}^\top \boldsymbol{x})^2] + \alpha^2 \mathbb{E}[(\tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2] + 2\alpha \mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x} \tilde{\boldsymbol{w}}^\top \boldsymbol{x}]$$

We have that, as proved in Firdoussi & Seddik (2024):

$$\begin{aligned}
\mathbb{E}[(\boldsymbol{w}^\top \boldsymbol{x})^2] &= \frac{1}{h(1+\delta_Q)} \left(\frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \Sigma_\beta \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - 2(1-h) \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right) + \frac{1-h}{h} \\
&= \frac{1}{h(1+\delta_Q)} \left(\frac{1}{1+\delta_Q} \left((\boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta)^2 + \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}}^2 \boldsymbol{\mu}_\beta \right) - 2(1-h) \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right) + \frac{1-h}{h} \\
&= \frac{\|\boldsymbol{\mu}_\beta\|^2}{h(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))} \left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)} - 2(1-h) \right) + \frac{1-h}{h}
\end{aligned}$$

And we have that:

$$\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2] = \tilde{\boldsymbol{w}}^\top \Sigma_\beta \tilde{\boldsymbol{w}}$$

And:

$$\begin{aligned}
\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x} \tilde{\boldsymbol{w}}^\top \boldsymbol{x}] &= \mathbb{E}[\boldsymbol{w}]^\top \Sigma_\beta \tilde{\boldsymbol{w}} \\
&= \frac{1}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \Sigma_\beta \tilde{\boldsymbol{w}}
\end{aligned}$$

Thus we have the first sum.

Second term: Now let us compute the expectation of the second term:

$$\begin{aligned}
\frac{1}{n^2} \mathbb{E}[(\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q} \boldsymbol{x})^2] &= \frac{1}{n^2} \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q} \boldsymbol{x}\boldsymbol{x}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q} \tilde{\boldsymbol{w}}] \\
&= \tilde{\boldsymbol{w}}^\top \mathbb{E}[\frac{1}{n}\mathbf{X}\mathbf{X}^\top \mathbf{Q} \Sigma_\beta \frac{1}{n}\mathbf{X}\mathbf{X}^\top \mathbf{Q}] \tilde{\boldsymbol{w}} \\
&= \tilde{\boldsymbol{w}}^\top \mathbb{E}[(\mathbf{Q}^{-1} - \gamma \mathbf{I}_p) \mathbf{Q} \Sigma_\beta (\mathbf{Q}^{-1} - \gamma \mathbf{I}_p) \mathbf{Q}] \tilde{\boldsymbol{w}} \\
&= \tilde{\boldsymbol{w}}^\top \mathbb{E}[(\mathbf{I}_p - \gamma \mathbf{Q}) \Sigma_\beta (\mathbf{I}_p - \gamma \mathbf{Q})] \tilde{\boldsymbol{w}} \\
&= \tilde{\boldsymbol{w}}^\top \mathbb{E}\left[\Sigma_\beta - \gamma \Sigma_\beta \mathbf{Q} - \gamma \mathbf{Q} \Sigma_\beta + \gamma^2 \mathbf{Q} \Sigma_\beta \mathbf{Q} \right] \tilde{\boldsymbol{w}} \\
&= \tilde{\boldsymbol{w}}^\top \left(\Sigma_\beta - \gamma \Sigma_\beta \bar{\mathbf{Q}} - \gamma \bar{\mathbf{Q}} \Sigma_\beta + \gamma^2 \right) \tilde{\boldsymbol{w}} \\
&= \tilde{\boldsymbol{w}}^\top \Sigma_\beta \boldsymbol{w} - 2\gamma \tilde{\boldsymbol{w}}^\top \Sigma_\beta \bar{\mathbf{Q}} \tilde{\boldsymbol{w}} + \gamma^2 \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q} \Sigma_\beta \mathbf{Q}] \tilde{\boldsymbol{w}}
\end{aligned}$$

Third term: Now we will compute the last term: $\frac{2\alpha}{n} \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q} \boldsymbol{x} (\boldsymbol{w}^\top \boldsymbol{x} + \alpha \tilde{\boldsymbol{w}}^\top \boldsymbol{x})]$.
We have that:

$$\begin{aligned}
&\frac{1}{n} \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q} \boldsymbol{x}\boldsymbol{x}^\top \boldsymbol{w}] = \tilde{\boldsymbol{w}}^\top \mathbb{E}[(\mathbf{Q}^{-1} - \gamma \mathbf{I}_p) \mathbf{Q} \Sigma_\beta \boldsymbol{w}] \\
&= \tilde{\boldsymbol{w}}^\top \mathbb{E}[(\mathbf{I}_p - \gamma \mathbf{Q}) \Sigma_\beta \boldsymbol{w}] \\
&= \tilde{\boldsymbol{w}}^\top \Sigma_\beta \mathbb{E}[\boldsymbol{w}] - \gamma \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q} \Sigma_\beta \boldsymbol{w}] \\
&= \tilde{\boldsymbol{w}}^\top \frac{\Sigma_\beta}{1+\delta_Q} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - \gamma \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q} \Sigma_\beta \boldsymbol{w}]
\end{aligned}$$

$$
\begin{aligned}
&= \tilde{\boldsymbol{w}}^\top \frac{\Sigma_\beta}{1+\delta_Q} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - \gamma \frac{1}{n} \sum_{i=1}^{n} \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q} \Sigma_\beta \mathbf{Q} y_i \boldsymbol{x}_i] \\
&= \tilde{\boldsymbol{w}}^\top \frac{\Sigma_\beta}{1+\delta_Q} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - \gamma \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q} \Sigma_\beta \mathbf{Q} y_i \boldsymbol{x}_i] \\
&= \tilde{\boldsymbol{w}}^\top \frac{\Sigma_\beta}{1+\delta_Q} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - \frac{\gamma}{1+\delta_Q} \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q} \Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i] \\
&= \tilde{\boldsymbol{w}}^\top \frac{\Sigma_\beta}{1+\delta_Q} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - \frac{\gamma}{1+\delta_Q} \tilde{\boldsymbol{w}}^\top \mathbb{E}\left[\left(\mathbf{Q}_{-i} - \frac{\frac{1}{n} \mathbf{Q}_{-i} \boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i}}{1+\delta_Q} \right) \Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i \right] \\
&= \tilde{\boldsymbol{w}}^\top \frac{\Sigma_\beta}{1+\delta_Q} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - \frac{\gamma}{1+\delta_Q} \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q}_{-i} \Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i] + \frac{\gamma}{n(1+\delta_Q)^2} \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q}_{-i} \boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i} \Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i] \\
&= \tilde{\boldsymbol{w}}^\top \frac{\Sigma_\beta}{1+\delta_Q} \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - \frac{\gamma}{1+\delta_Q} \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q} \Sigma_\beta \mathbf{Q}] \boldsymbol{\mu}_\beta + \frac{\gamma}{n(1+\delta_Q)^2} \operatorname{Tr}(\Sigma_\beta \mathbb{E}[\mathbf{Q} \Sigma_\beta \mathbf{Q}]) \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta
\end{aligned}
$$

And:

$$
\begin{aligned}
&\frac{1}{n} \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X} \mathbf{X}^\top \mathbf{Q} \boldsymbol{x} \boldsymbol{x}^\top \tilde{\boldsymbol{w}}] = \tilde{\boldsymbol{w}}^\top \mathbb{E}[(\mathbf{Q}^{-1} - \gamma \mathbf{I}_p) \mathbf{Q} \Sigma_\beta] \tilde{\boldsymbol{w}} \\
&= \tilde{\boldsymbol{w}}^\top \mathbb{E}[(\mathbf{I}_p - \gamma \mathbf{Q}) \Sigma_\beta] \tilde{\boldsymbol{w}} \\
&= \tilde{\boldsymbol{w}}^\top \Sigma_\beta \tilde{\boldsymbol{w}} - \gamma \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \Sigma_\beta \tilde{\boldsymbol{w}}
\end{aligned}
$$

Grouping all the terms: Thus, we now that we have the expression of all the term, we will group them in the following way:

$$
\mathbb{E}[(\boldsymbol{w}_\alpha^\top \boldsymbol{x})^2] = T_1 + \alpha T_2 + \alpha^2 T_3
$$

Terms without α:

$$
T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))} \left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)} - 2(1-h) \right) + \frac{1-h}{h} \tag{42}
$$

Terms in α: There are two : $2\,\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x} \tilde{\boldsymbol{w}}^\top \boldsymbol{x}]$ and $\frac{2}{n} \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X} \mathbf{X}^\top \mathbf{Q} \boldsymbol{x} \boldsymbol{w}^\top \boldsymbol{x}]$:

$$
\begin{aligned}
T_2 &= 2\,\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x} \tilde{\boldsymbol{w}}^\top \boldsymbol{x}] - \frac{2}{n} \mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X} \mathbf{X}^\top \mathbf{Q} \boldsymbol{x} \boldsymbol{w}^\top \boldsymbol{x}] \\
&= \frac{2\gamma}{h(1+\delta_Q)} \left(\tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \Sigma_\beta \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta - (1-h)(1+\delta_Q) \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right) \\
&= \frac{2\gamma}{h(1+\delta_Q)} \left(\tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta + \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}}^2 \boldsymbol{\mu}_\beta - (1-h)(1+\delta_Q) \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \right)
\end{aligned}
$$

And we have that:

$$
\tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta = \frac{(1+\delta_Q)^2 \|\boldsymbol{\mu}_\beta\|^2 \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta}{(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))^2}, \quad \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}}^2 \boldsymbol{\mu}_\beta = \frac{(1+\delta_Q)^2 \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta}{(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))^2}.
$$

Thus:

$$
T_2 = \frac{2\gamma(1+\delta_Q) \tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta}{h(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))} \left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)} - (1-h) \right)
$$

Terms in α^2: we have three terms: $\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2]$, $\frac{1}{n^2}\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2]$ and $\frac{-2}{n}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\tilde{\boldsymbol{w}}^\top \boldsymbol{x}]$:

$$
\begin{aligned}
T_3 &= \mathbb{E}[(\tilde{\boldsymbol{w}}^\top \boldsymbol{x})^2] + \frac{1}{n^2}\mathbb{E}[(\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x})^2] - \frac{2}{n}\mathbb{E}[\tilde{\boldsymbol{w}}^\top \mathbf{X}\mathbf{X}^\top \mathbf{Q}\boldsymbol{x}\tilde{\boldsymbol{w}}^\top \boldsymbol{x}] \\
&= \tilde{\boldsymbol{w}}^\top \Sigma_\beta \tilde{\boldsymbol{w}} + \tilde{\boldsymbol{w}}^\top \Sigma_\beta \tilde{\boldsymbol{w}} - 2\gamma \tilde{\boldsymbol{w}}^\top \Sigma_\beta \bar{\mathbf{Q}} \tilde{\boldsymbol{w}} + \gamma^2 \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}]\tilde{\boldsymbol{w}} - 2\tilde{\boldsymbol{w}}^\top \Sigma_\beta \tilde{\boldsymbol{w}} + 2\gamma \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}}\Sigma_\beta \tilde{\boldsymbol{w}} \\
&= \gamma^2 \tilde{\boldsymbol{w}}^\top \mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}]\tilde{\boldsymbol{w}} \\
&= \frac{\gamma^2}{h}\tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}}\tilde{\boldsymbol{w}} \\
&= \frac{\gamma^2}{h}\left((\tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}}\boldsymbol{\mu}_\beta)^2 + \tilde{\boldsymbol{w}}^\top \bar{\mathbf{Q}}^2 \tilde{\boldsymbol{w}}\right) \\
&= \frac{\gamma^2(1+\delta_Q)^2}{h}\left(\frac{(\tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta)^2}{(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))^2} + \frac{1-h}{\eta}\left(\|\tilde{\boldsymbol{w}}\|^2 + \frac{\|\boldsymbol{\mu}_\beta\|^2(\tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta)^2}{(\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q))^2} - \frac{2(\tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta)^2}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)}\right)\right) \\
&= \frac{\gamma^2(1+\delta_Q)^2}{h}\left(\frac{(\tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta)^2}{\lambda_Q^2} + \frac{1-h}{\eta}\|\tilde{\boldsymbol{w}}\|^2 + \frac{(1-h)(\tilde{\boldsymbol{w}}^\top \boldsymbol{\mu}_\beta)^2}{\eta\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q} - 2\right)\right)
\end{aligned}
$$

D Extension to Multi-Source Transfer Learning

Given T source classifiers $\{\boldsymbol{w}_t\}_{t=1}^T$ and a single target task, the goal is to fine-tune a mixture of these classifiers on the target task. Specifically, we want to find the optimal fine-tuned classifier $\boldsymbol{w}_\Omega$ that is written as:

$$
\boldsymbol{w}_\Omega = \sum_{t=1}^T \alpha_t \boldsymbol{w}_t + \boldsymbol{a}
$$

where $\alpha_t \in \mathbb{R}$ and $\boldsymbol{a}$ is an adapter trained on the target dataset as follows:

$$
\boldsymbol{a} = \arg\min_{\boldsymbol{v}} \frac{1}{n}\|\mathbf{X}^\top(\sum_{t=1}^T \alpha_t \boldsymbol{w}_t + \boldsymbol{v}) - \boldsymbol{y}\|^2 + \gamma\|\boldsymbol{v}\|^2
$$

Then, $\boldsymbol{a}$ expresses as:

$$
\boldsymbol{a} = \frac{1}{n}\left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + \gamma \mathbf{I}_p\right)^{-1}\left(\mathbf{X}\boldsymbol{y} - \mathbf{X}\mathbf{X}^\top \sum_{t=1}^T \alpha_t \boldsymbol{w}_t\right)
$$

Thus, our new fine-tuned classifier writes as:

$$
\boldsymbol{w}_\Omega = \sum_{t=1}^T \alpha_t \boldsymbol{w}_t + \boldsymbol{a} = \frac{1}{n}\mathbf{Q}\mathbf{X}\boldsymbol{y} + \gamma \sum_{t=1}^T \alpha_t \mathbf{Q}\boldsymbol{w}_t
$$

To compute the theoretical test accuracy of this classifier, we will take a test sample $\boldsymbol{x} \sim \mathcal{N}((-1)^a \boldsymbol{\mu}_\beta, \mathbf{I}_p)$, independent from the training data $(\boldsymbol{x}_i)_{i=1}^n$, and we compute the statistics of the decision function $\boldsymbol{w}_\Omega^\top \boldsymbol{x}$.

D.1 Test Expectation

We have that:

$$
\mathbb{E}[\boldsymbol{w}_\Omega^\top \boldsymbol{x}] = \mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}] + \gamma \sum_{t=1}^T \alpha_t \mathbb{E}[\boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x}]
$$

$$= \mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}] + (-1)^a \gamma \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta$$

From the previous section, we have that:

$$\mathbb{E}[\boldsymbol{w}^\top \boldsymbol{x}] = \frac{(-1)^a}{1+\delta_Q} \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta = \frac{(-1)^a \|\boldsymbol{\mu}_\beta\|^2}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)}$$

And from lemma A.5, we have that:

$$\boldsymbol{w}_t^\top \bar{\mathbf{Q}} \boldsymbol{\mu}_\beta = \frac{(1+\delta_Q)\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta \rangle}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)}$$

Finally, we get that:

$$\boxed{\mathbb{E}[\boldsymbol{w}_\Omega^\top \boldsymbol{x}] = \frac{(-1)^a}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)} \left(\|\boldsymbol{\mu}_\beta\|^2 + \gamma(1+\delta_Q) \sum_{t=1}^{T} \alpha_t \langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta \rangle \right)}$$

In a vectorized form, denote by $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_T)^\top$ the vector of coefficients and by $\mathbf{W} = (\boldsymbol{w}_1, \dots, \boldsymbol{w}_T) \in \mathbb{R}^{p \times T}$, then we have that:

$$\boxed{\mathbb{E}[\boldsymbol{w}_\Omega^\top \boldsymbol{x}] = (-1)^a \frac{\left(\|\boldsymbol{\mu}_\beta\|^2 + \gamma(1+\delta_Q) \boldsymbol{\alpha}^\top \mathbf{W}^\top \boldsymbol{\mu}_\beta \right)}{\|\boldsymbol{\mu}_\beta\|^2 + 1 + \gamma(1+\delta_Q)}}$$

D.2 Test variance

Now we will compute the expectation of the second order moment of $\boldsymbol{w}_\Omega^\top \boldsymbol{x}$:

$$\mathbb{E}[(\boldsymbol{w}_\Omega^\top \boldsymbol{x})^2] = \mathbb{E}\left[(\boldsymbol{w}^\top \boldsymbol{x})^2 + \gamma^2 \left(\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q} \boldsymbol{x} \right)^2 + 2\gamma \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q} \boldsymbol{x} \boldsymbol{w}^\top \boldsymbol{x} \right]$$

Let us compute each term of this sum and then aggregate the results at the end.

First term. We have that:

$$\mathbb{E}[(\boldsymbol{w}^\top \boldsymbol{x})^2] = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q} \left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\lambda_Q} - 2(1-h) \right) + \frac{1-h}{h}$$

Second term. Now let us compute the second term of the sum:

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q} \boldsymbol{x} \boldsymbol{w}^\top \boldsymbol{x} \right] &= \sum_{t=1}^{T} \alpha_t \, \mathbb{E}[\boldsymbol{w}_t^\top \mathbf{Q} \boldsymbol{x} \boldsymbol{x}^\top \boldsymbol{w}] \\
&= \sum_{t=1}^{T} \alpha_t \, \mathbb{E}[\boldsymbol{w}_t^\top \mathbf{Q} \Sigma_\beta \boldsymbol{w}] \\
&= \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \, \mathbb{E}[\mathbf{Q} \Sigma_\beta \frac{1}{n} \sum_{i=1}^{n} y_i \mathbf{Q} \boldsymbol{x}_i]
\end{aligned}$$

$$= \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q} y_i \boldsymbol{x}_i] \qquad (\boldsymbol{x}_i \text{ i.i.d})$$

$$= \frac{1}{1+\delta_Q} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i]$$

And since we have that:

$$\mathbf{Q} = \mathbf{Q}_{-i} - \frac{\mathbf{Q}_{-i}\boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i}}{n(1+\delta_Q)}$$

Then:

$$\mathbb{E}\left[\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{w}^\top \boldsymbol{x}\right] = \frac{1}{1+\delta_Q} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}\left[\left(\mathbf{Q}_{-i} - \frac{\mathbf{Q}_{-i}\boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i}}{n(1+\delta_Q)}\right)\Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i\right]$$

$$= \frac{1}{1+\delta_Q} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}_{-i}\Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i] - \frac{1}{n(1+\delta_Q)^2} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i}\Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i]$$

We have that:

$$\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}_{-i}\Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i] = \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}]\boldsymbol{\mu}_\beta$$

$$= \frac{1}{h} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}}\boldsymbol{\mu}_\beta$$

$$= \frac{1}{h} \sum_{t=1}^{T} \alpha_t \frac{(1+\delta_Q)^2}{\lambda_Q^2} \langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta \rangle \left(\|\boldsymbol{\mu}_\beta\|^2 + 1\right)$$

And we have that:

$$\frac{1}{n(1+\delta_Q)^2} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i}\Sigma_\beta \mathbf{Q}_{-i} y_i \boldsymbol{x}_i] = \frac{1}{n(1+\delta_Q)^2} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}_{-i} y_i \boldsymbol{x}_i \operatorname{Tr}(\boldsymbol{x}_i \boldsymbol{x}_i^\top \mathbf{Q}_{-i}\Sigma_\beta \mathbf{Q}_{-i})]$$

$$= \frac{1}{n(1+\delta_Q)^2} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}_{-i} y_i \boldsymbol{x}_i \operatorname{Tr}(\Sigma_\beta \mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}])]$$

$$= \frac{1}{n(1+\delta_Q)^2} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbb{E}[\mathbf{Q}_{-i} y_i \boldsymbol{x}_i] \frac{1}{h} \operatorname{Tr}((\Sigma_\beta \bar{\mathbf{Q}})^2)$$

$$= \frac{1-h}{h} \sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \bar{\mathbf{Q}}\boldsymbol{\mu}_\beta$$

$$= \frac{1-h}{h} \sum_{t=1}^{T} \alpha_t \frac{(1+\delta_Q)\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta \rangle}{\lambda_Q}$$

Thus the second term is given by:

$$\mathbb{E}\left[\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{w}^\top \boldsymbol{x}\right] = \frac{(1+\delta_Q)}{h\lambda_Q} \sum_{t=1}^{T} \alpha_t \left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\lambda_Q} - (1-h)\right) \langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta \rangle$$

$$= \frac{(1+\delta_Q)}{h\lambda_Q} \left(\frac{\|\boldsymbol{\mu}_\beta\|^2 + 1}{\lambda_Q} - (1-h)\right) \boldsymbol{\alpha}^\top \mathbf{W}^\top \boldsymbol{\mu}_\beta$$

Third term. We have that:

$$\begin{aligned}
\gamma^2 \mathbb{E}\left[\left(\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x}\right)^2\right] &= \gamma^2 \mathbb{E}\left[\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x} \sum_{k=1}^{T} \alpha_k \boldsymbol{w}_k^\top \mathbf{Q}\boldsymbol{x}\right] \\
&= \gamma^2 \sum_{t,k=1}^{T} \mathbb{E}[\alpha_t \alpha_k \boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x}\boldsymbol{x}^\top \mathbf{Q}\boldsymbol{w}_k] \\
&= \gamma^2 \sum_{t,k=1}^{T} \mathbb{E}[\boldsymbol{w}_t^\top \mathbf{Q}\Sigma_\beta \mathbf{Q}\boldsymbol{w}_k] \\
&= \gamma^2 \sum_{t,k=1}^{T} \boldsymbol{w}_t^\top \, \mathbb{E}[\mathbf{Q}\Sigma_\beta \mathbf{Q}]\boldsymbol{w}_k \\
&= \frac{\gamma^2}{h} \sum_{t,k=1}^{T} \alpha_t \alpha_k \boldsymbol{w}_t^\top \bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}}\boldsymbol{w}_k
\end{aligned}$$

And we have that:

$$\begin{aligned}
\bar{\mathbf{Q}}\Sigma_\beta \bar{\mathbf{Q}} &= \bar{\mathbf{Q}}\left(\boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top + \mathbf{I}_p\right)\bar{\mathbf{Q}} \\
&= \bar{\mathbf{Q}}\boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top \bar{\mathbf{Q}} + \bar{\mathbf{Q}}^2 \\
&= \frac{(1+\delta_Q)^2}{\lambda_Q^2}\boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top + \frac{(1+\delta_Q)^2}{(1+\gamma(1+\delta_Q))^2}\left(\mathbf{I}_p + \frac{(\boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top)^2}{\lambda_Q^2} - \frac{2\boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top}{\lambda_Q}\right)
\end{aligned}$$

Thus the last term is given by:

$$\gamma^2 \mathbb{E}\left[\left(\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x}\right)^2\right] = \frac{\gamma^2(1+\delta_Q)^2}{h} \times$$
$$\sum_{t,k=1}^{T} \alpha_t \alpha_k \left[\frac{\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k, \boldsymbol{\mu}_\beta\rangle}{\lambda_Q^2} + \frac{1}{(1+\gamma(1+\delta_Q))^2}\left(\langle \boldsymbol{w}_t, \boldsymbol{w}_k\rangle + \frac{\|\boldsymbol{\mu}_\beta\|^2\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k, \boldsymbol{\mu}_\beta\rangle}{\lambda_Q^2} - \frac{2\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k, \boldsymbol{\mu}_\beta\rangle}{\lambda_Q}\right)\right]$$

In a vectorized form, we have that:

$$\begin{aligned}
&\gamma^2 \mathbb{E}\left[\left(\sum_{t=1}^{T} \alpha_t \boldsymbol{w}_t^\top \mathbf{Q}\boldsymbol{x}\right)^2\right] = \frac{\gamma^2(1+\delta_Q)^2}{h} \times \\
&\left[\frac{(\boldsymbol{\alpha}^\top \mathbf{W}^\top \boldsymbol{\mu}_\beta)^2}{\lambda_Q^2} + \frac{1}{(1+\gamma(1+\delta_Q))^2}\left(\boldsymbol{\alpha}^\top \mathbf{W}^\top \mathbf{W}\boldsymbol{\alpha} + \frac{\|\boldsymbol{\mu}_\beta\|^2(\boldsymbol{\alpha}^\top \mathbf{W}^\top \boldsymbol{\mu}_\beta)^2}{\lambda_Q^2} - \frac{2(\boldsymbol{\alpha}^\top \mathbf{W}^\top \boldsymbol{\mu}_\beta)^2}{\lambda_Q}\right)\right] \\
&= \frac{\gamma^2(1+\delta_Q)^2}{h}\boldsymbol{\alpha}^\top \mathbf{M}\boldsymbol{\alpha}
\end{aligned}$$

where:

$$\mathbf{M} = \frac{(1-h)}{\eta}\mathbf{W}^\top \mathbf{W} + \left(\frac{1}{\lambda_Q^2} + \frac{(1-h)}{\eta\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2}{\lambda_Q} - 2\right)\right)\mathbf{W}^\top \boldsymbol{\mu}_\beta \boldsymbol{\mu}_\beta^\top \mathbf{W}^\top$$

Finally gives us the expression of the second order moment of $\boldsymbol{w}_\Omega^\top \boldsymbol{x}$ as follows:

$$\mathbb{E}[(\boldsymbol{w}_\Omega^\top \boldsymbol{x})^2] = T_1 + T_2 + T_3$$

where:

$$T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q} - 2(1-h)\right) + \frac{1-h}{h}$$

$$T_2 = \frac{2\gamma(1+\delta_Q)}{h\lambda_Q}\sum_{t=1}^{T}\alpha_t\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q} - (1-h)\right)\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle$$

$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\times$$

$$\sum_{t,k=1}^{T}\alpha_t\alpha_k\left[\frac{\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k, \boldsymbol{\mu}_\beta\rangle}{\lambda_Q^2} + \frac{1}{(1+\gamma(1+\delta_Q))^2}\left(\langle \boldsymbol{w}_t, \boldsymbol{w}_k\rangle + \frac{\|\boldsymbol{\mu}_\beta\|^2\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k, \boldsymbol{\mu}_\beta\rangle}{\lambda_Q^2} - \frac{2\langle \boldsymbol{w}_t, \boldsymbol{\mu}_\beta\rangle\langle \boldsymbol{w}_k, \boldsymbol{\mu}_\beta\rangle}{\lambda_Q}\right)\right]$$

Which also writes in a vectorized form:

$$T_1 = \frac{\|\boldsymbol{\mu}_\beta\|^2}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q} - 2(1-h)\right) + \frac{1-h}{h}$$

$$T_2 = \frac{2\gamma(1+\delta_Q)}{h\lambda_Q}\left(\frac{\|\boldsymbol{\mu}_\beta\|^2+1}{\lambda_Q} - (1-h)\right)\boldsymbol{\alpha}^\top\mathbf{W}^\top\boldsymbol{\mu}_\beta$$

$$T_3 = \frac{\gamma^2(1+\delta_Q)^2}{h}\boldsymbol{\alpha}^\top\mathbf{M}\boldsymbol{\alpha}$$

E Extension to Linear Regression Transfer

We start by stating the lemmas which will be necessary to derive our results.

E.1 Preliminary results

We have that the resolvent matrix we're working with in this section is given by:

$$\mathbf{Q}(\gamma) = \left(\frac{1}{n}\mathbf{X}\mathbf{X}^\top + \gamma\mathbf{I}_p\right)^{-1}$$

A deterministic equivalent of this latter is:

$$\bar{\mathbf{Q}}(\gamma) = \frac{1+\delta}{1+\gamma(1+\delta)}\mathbf{I}_p, \quad \delta = \frac{\eta-\gamma-1+\sqrt{(\eta-\gamma-1)^2+4\eta\gamma}}{2\gamma} \tag{43}$$

Lemma E.1 (Deterministic equivalent of **QAQ**)**.** *For any deterministic positive semi-definite matrix* $\mathbf{A}\in\mathbb{R}^{p\times p}$*, we have that:*

$$\mathbf{QAQ} \leftrightarrow \frac{(1+\delta)^2}{(1+\gamma(1+\delta))^2}\mathbf{A} + \frac{1}{n}\frac{\mathrm{Tr}(\mathbf{A})}{(1+\gamma(1+\delta))^2}\,\mathbb{E}[\mathbf{Q}^2]$$

Furthermore, we can use the above d.e to find that:

$$\mathbb{E}[\mathbf{Q}^2] = \frac{(1+\delta)^2}{(1+\gamma(1+\delta))^2-\eta}\mathbf{I}_p \tag{44}$$

Which then gives a simplified expression of the above deterministic equivalent:

$$\mathbf{QAQ} \leftrightarrow \frac{(1+\delta)^2}{(1+\gamma(1+\delta))^2}\left(\mathbf{A} + \frac{\frac{1}{n}\mathrm{Tr}(\mathbf{A})}{(1+\gamma(1+\delta))^2-\eta}\mathbf{I}_p\right)$$

Proof. Similar to the proof of lemma A.7. □

E.2 Test error

We recall that we want to study the transfer learning between two linear regression tasks. In fact, given a pre-trained (source) linear regressor $\mathbf{W}_s$, we want to adapt it to fit a target dataset comprised of n features $\mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{p \times n}$ and their corresponding labels $\mathbf{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n] \in \mathbb{R}^{d \times n}$ where:

$$\boldsymbol{x}_i \sim \mathscr{N}(0, \mathbf{I}_p), \quad \boldsymbol{y}_i = \mathbf{W}_t \boldsymbol{x}_i + \boldsymbol{z}_i, \quad \boldsymbol{z}_i \sim \mathscr{N}(0, \sigma^2 \mathbf{I}_d)$$

This gave us a closed-form solution of the fine-tuned classifier which expresses as follows:

$$\mathbf{W}_\alpha = \frac{1}{n} \mathbf{Y}\mathbf{X}^\top \mathbf{Q} + \alpha\gamma \mathbf{W}_s \mathbf{Q}$$

To evaluate the efficiency of this fine-tuning process, we can compute the theoretical test error of this regressor defined as follows:

$$E_{\text{test}} = \mathbb{E}\left[\|\mathbf{W}_\alpha \boldsymbol{x} - \boldsymbol{y}\|^2\right] \tag{45}$$

for test samples $(\boldsymbol{x}, \boldsymbol{y})$ independent of the training set. This error decomposes as follows:

$$E_{\text{test}} = \mathbb{E}[\|\mathbf{W}_\alpha \boldsymbol{x}\|^2 + \|\boldsymbol{y}\|^2 - 2\boldsymbol{x}^\top \mathbf{W}_\alpha^\top \boldsymbol{y}]$$

Therefore, we will compute the expectation of each term and then aggregate the results to get the theoretical test risk of the fine-tuned regressor.
First, denote by:

$$\lambda = (1 + \gamma(1 + \delta)) \tag{46}$$

First term. We have that:

$$\begin{aligned}
\mathbb{E}[\|\boldsymbol{y}\|^2] &= \mathbb{E}[\boldsymbol{y}^\top \boldsymbol{y}] \\
&= \mathbb{E}[(\mathbf{W}_t \boldsymbol{x} + \boldsymbol{z})^\top (\mathbf{W}_t \boldsymbol{x} + \boldsymbol{z})] \\
&= \mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_t^\top \mathbf{W}_t \boldsymbol{x}] + 2\,\mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_t^\top \boldsymbol{z}] + \mathbb{E}[\boldsymbol{z}^\top \boldsymbol{z}]
\end{aligned}$$

We have that:

$$\mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_t^\top \mathbf{W}_t \boldsymbol{x}] = \operatorname{Tr}(\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top \mathbf{W}_t^\top \mathbf{W}_t]) = \operatorname{Tr}(\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top] \mathbf{W}_t^\top \mathbf{W}_t) = \operatorname{Tr}(\mathbf{W}_t^\top \mathbf{W}_t)$$

And:

$$\mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_t^\top \boldsymbol{z}] = \mathbb{E}[\boldsymbol{x}]^\top \mathbf{W}_t^\top\, \mathbb{E}[\boldsymbol{z}] = 0$$

and finally:

$$\mathbb{E}[\boldsymbol{z}^\top \boldsymbol{z}] = \sigma^2 . d$$

Thus:

$$\boxed{\mathbb{E}[\|\boldsymbol{y}\|^2] = \operatorname{Tr}(\mathbf{W}_t^\top \mathbf{W}_t) + \sigma^2 . d}$$

Second term. We will now compute: $\mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_\alpha \boldsymbol{y}]$. We have that:

$$\begin{aligned}\mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_\alpha^\top \boldsymbol{y}] &= \operatorname{Tr}(\mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_\alpha^\top \boldsymbol{y}]) \\ &= \operatorname{Tr}(\mathbb{E}[\boldsymbol{y}\boldsymbol{x}^\top \mathbf{W}_\alpha^\top]) \\ &= \operatorname{Tr}(\mathbb{E}[\boldsymbol{y}\boldsymbol{x}^\top]\,\mathbb{E}[\mathbf{W}_\alpha^\top])\end{aligned}$$

We have that:

$$\begin{aligned}\mathbb{E}[\boldsymbol{y}\boldsymbol{x}^\top] &= \mathbb{E}[(\mathbf{W}_t\boldsymbol{x} + \boldsymbol{z})\boldsymbol{x}^\top] \\ &= \mathbb{E}[\mathbf{W}_t\boldsymbol{x}\boldsymbol{x}^\top + \boldsymbol{z}\boldsymbol{x}^\top] \\ &= \mathbf{W}_t\end{aligned}$$

And :

$$\begin{aligned}\mathbb{E}[\mathbf{W}_\alpha^\top] &= \frac{1}{n}\,\mathbb{E}[\mathbf{Q}\mathbf{X}\mathbf{Y}^\top] + \alpha\gamma\,\mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top] \\ &= \frac{1}{n}\sum_{i=1}^{\top}\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top] + \alpha\gamma\bar{\mathbf{Q}}\mathbf{W}_s^\top \\ &= \frac{1}{n(1+\delta)}\sum_{i=1}^{n}\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{y}_i^\top] + \alpha\gamma\bar{\mathbf{Q}}\mathbf{W}_s^\top \\ &= \frac{1}{n(1+\delta)}\sum_{i=1}^{n}\mathbb{E}[\mathbf{Q}_{-i}\,\mathbb{E}_i[\boldsymbol{x}_i\boldsymbol{y}_i^\top]] + \alpha\gamma\bar{\mathbf{Q}}\mathbf{W}_s^\top \\ &= \frac{1}{n(1+\delta)}\sum_{i=1}^{n}\bar{\mathbf{Q}}\mathbf{W}_t^\top + \alpha\gamma\bar{\mathbf{Q}}\mathbf{W}_s^\top \\ &= \bar{\mathbf{Q}}\left(\frac{\mathbf{W}_t^\top}{1+\delta} + \alpha\gamma\mathbf{W}_s^\top\right) \\ &= \frac{(1+\delta)}{\lambda}\left(\frac{\mathbf{W}_t^\top}{1+\delta} + \alpha\gamma\mathbf{W}_s^\top\right)\end{aligned}$$

Thus:

$$\boxed{\mathbb{E}[\boldsymbol{x}^\top \mathbf{W}_\alpha \boldsymbol{y}] = \frac{1}{\lambda}\left(\operatorname{Tr}(\mathbf{W}_t^\top\mathbf{W}_t) + \alpha\gamma(1+\delta)\operatorname{Tr}(\mathbf{W}_s^\top\mathbf{W}_t)\right)}$$

Third term. Now we will compute the last term of our sum: $\mathbb{E}[\|\mathbf{W}_\alpha\boldsymbol{x}\|^2]$. We have that:

$$\begin{aligned}\mathbb{E}[\|\mathbf{W}_\alpha\boldsymbol{x}\|^2] &= \mathbb{E}[\boldsymbol{x}^\top\mathbf{W}_\alpha^\top\mathbf{W}_\alpha\boldsymbol{x}] \\ &= \operatorname{Tr}(\mathbb{E}[\boldsymbol{x}\boldsymbol{x}^\top]\,\mathbb{E}[\mathbf{W}_\alpha^\top\mathbf{W}_\alpha]) \\ &= \operatorname{Tr}(\mathbb{E}[\mathbf{W}_\alpha^\top\mathbf{W}_\alpha])\end{aligned}$$

And we have that:

$$\begin{aligned}\mathbb{E}[\mathbf{W}_\alpha^\top\mathbf{W}_\alpha] &= \mathbb{E}\left[(\frac{1}{n}\mathbf{Q}\mathbf{X}\mathbf{Y}^\top + \alpha\gamma\mathbf{Q}\mathbf{W}_s^\top)(\frac{1}{n}\mathbf{Y}\mathbf{X}^\top\mathbf{Q} + \alpha\gamma\mathbf{W}_s\mathbf{Q})\right] \\ &= \frac{1}{n^2}\,\mathbb{E}[\mathbf{Q}\mathbf{X}\mathbf{Y}^\top\mathbf{Y}\mathbf{X}^\top\mathbf{Q}] + \frac{2\alpha\gamma}{n}\,\mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top\mathbf{Y}\mathbf{X}^\top\mathbf{Q}] + (\alpha\gamma)^2\,\mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top\mathbf{W}_s\mathbf{Q}]\end{aligned}$$

$$= A_1 + A_2 + A_3$$

We have that:

$$\begin{aligned}
A_1 &= \frac{1}{n^2}\,\mathbb{E}[\mathbf{Q}\mathbf{X}\mathbf{Y}^\top\mathbf{Y}\mathbf{X}^\top\mathbf{Q}] = \frac{1}{n^2}\sum_{i,j=1}^{n}\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}] \\
&= \frac{1}{n^2}\sum_{i=1}^{n}\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}] + \frac{1}{n^2}\sum_{i\neq j}\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}] \\
&= \frac{1}{n}\,\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}] + (1-\frac{1}{n})\,\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}] \\
&= T_1 + T_2
\end{aligned}$$

We compute each term of this decomposition then aggregate the results. We have that:

$$T_1 = \frac{1}{n}\,\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}] = \frac{1}{n(1+\delta)^2}\,\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] = \frac{1}{n(1+\delta)^2}\,\mathbb{E}[\mathbf{Q}_{-i}\,\mathbb{E}_i[\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_i\boldsymbol{x}_i^\top]\mathbf{Q}_{-i}]$$

And:

$$\begin{aligned}
\mathbb{E}[\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_i\boldsymbol{x}_i^\top] &= \mathbb{E}[\boldsymbol{x}_i(\boldsymbol{x}_i^\top\mathbf{W}_t^\top + \boldsymbol{z}_i^\top)(\mathbf{W}_t\boldsymbol{x}_i + \boldsymbol{z}_i)\boldsymbol{x}_i^\top] \\
&= \mathbb{E}[\boldsymbol{x}_i(\boldsymbol{x}_i^\top\mathbf{W}_t^\top\mathbf{W}_t\boldsymbol{x}_i + 2\boldsymbol{x}_i^\top\mathbf{W}_t^\top\boldsymbol{z}_i + \boldsymbol{z}_i^\top\boldsymbol{z}_i)\boldsymbol{x}_i^\top] \\
&= \mathbb{E}[\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{W}_t^\top\mathbf{W}_t\boldsymbol{x}_i\boldsymbol{x}_i^\top] + \mathbb{E}[\boldsymbol{x}_i\boldsymbol{z}_i^\top\boldsymbol{z}_i\boldsymbol{x}_i^\top] \qquad \text{(because } \mathbb{E}[\boldsymbol{z}_i] = 0\text{)} \\
&= \mathbb{E}[\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{W}_t^\top\mathbf{W}_t\boldsymbol{x}_i\boldsymbol{x}_i^\top] + \sigma^2.d.\,\mathbb{E}[\boldsymbol{x}_i\boldsymbol{x}_i^\top] \\
&= \mathbb{E}[\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{W}_t^\top\mathbf{W}_t\boldsymbol{x}_i\boldsymbol{x}_i^\top] + \sigma^2.d.\mathbf{I}_p
\end{aligned}$$

And by concentration in random matrices, we have that the term: $\frac{1}{n}\boldsymbol{x}_i^\top\mathbf{W}_t^\top\mathbf{W}_t\boldsymbol{x}_i$ concentrates to its expectation rapidly, and: $\frac{1}{n}\,\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{W}_t^\top\mathbf{W}_t\boldsymbol{x}_i] = \frac{1}{n}\,\mathrm{Tr}(\mathbf{W}_t^\top\mathbf{W}_t)$. Thus:

$$\mathbb{E}[\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_i\boldsymbol{x}_i^\top] = \Big(\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top) + \sigma^2.d\Big)\,\mathbf{I}_p$$

Thus:

$$\begin{aligned}
T_1 &= \frac{1}{n(1+\delta)^2}\Big(\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top) + \sigma^2.d\Big)\,\mathbb{E}[\mathbf{Q}^2] \\
&= \frac{\big(\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top) + \sigma^2.d\big)}{n(\lambda^2-\eta)}\mathbf{I}_p
\end{aligned}$$

And then we have that:

$$\begin{aligned}
T_2 &= (1-\frac{1}{n})\,\mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}] \\
&= \mathbb{E}[\mathbf{Q}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}] + \mathscr{O}(n^{-1}) \\
&= \frac{1}{(1+\delta)^2}\,\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}_{-j}] \\
&= \frac{1}{(1+\delta)^2}\,\mathbb{E}\left[\left(\mathbf{Q}_{-ij} - \frac{\mathbf{Q}_{-ij}\boldsymbol{x}_j\boldsymbol{x}_j^\top\mathbf{Q}_{-ij}}{n(1+\delta)}\right)\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\left(\mathbf{Q}_{-ij} - \frac{\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}}{n(1+\delta)}\right)\right]
\end{aligned}$$

$$= \frac{1}{(1+\delta)^2}(B_1 - B_2 - B_3 + B_4)$$

In fact:

$$\begin{aligned}
B_1 &= \mathbb{E}[\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}_{-ij}] \\
&= \mathbb{E}[\mathbf{Q}_{-ij}\,\mathbb{E}_i[\boldsymbol{x}_i\boldsymbol{y}_i^\top]\,\mathbb{E}_j[\boldsymbol{y}_j\boldsymbol{x}_j^\top]\mathbf{Q}_{-ij}] \\
&= \mathbb{E}[\mathbf{Q}_{-ij}\mathbf{W}_t^\top\mathbf{W}_t\mathbf{Q}_{-ij}] \\
&= \mathbb{E}[\mathbf{Q}\mathbf{W}_t^\top\mathbf{W}_t\mathbf{Q}] \\
&= \frac{(1+\delta)^2}{\lambda^2}\left(\mathbf{W}_t^\top\mathbf{W}_t + \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n(\lambda^2-\eta)}\mathbf{I}_p\right)
\end{aligned}$$

And:

$$\begin{aligned}
B_2 &= \frac{1}{n(1+\delta)}\,\mathbb{E}[\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}] \\
&= \frac{1}{n(1+\delta)}\,\mathbb{E}[\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}]\operatorname{Tr}(\mathbb{E}[\boldsymbol{y}_i^\top\boldsymbol{y}_j\boldsymbol{x}_j^\top\mathbf{Q}_{-ij}\boldsymbol{x}_i]) \\
&= \frac{1}{n(1+\delta)}\,\mathbb{E}[\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}]\operatorname{Tr}(\mathbf{W}_t^\top\mathbf{W}_t\bar{\mathbf{Q}}) \\
&= \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n\lambda}\,\mathbb{E}[\mathbf{Q}_{-ij}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-ij}] \\
&= \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n\lambda}\,\mathbb{E}[\mathbf{Q}^2]
\end{aligned}$$

Thus:

$$B_2 = \frac{(1+\delta)^2}{n\lambda(\lambda^2-\eta)}\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)\mathbf{I}_p$$

And because the training data is i.i.d:

$$B_3 = B_2, \quad B_4 = \mathscr{O}(n^{-1})$$

Hence:

$$\begin{aligned}
T_2 &= \frac{1}{(1+\delta)^2}(B_1 - 2b_2) \\
&= \frac{\mathbf{W}_t^\top\mathbf{W}_t}{\lambda^2} + \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n\lambda(\lambda^2-\eta)}\left(\frac{1}{\lambda} - 2\right)\mathbf{I}_p
\end{aligned}$$

Thus:

$$\begin{aligned}
A_1 &= T_1 + T_2 \\
&= \frac{\mathbf{W}_t^\top\mathbf{W}_t}{\lambda^2} + \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n\lambda(\lambda^2-\eta)}\left(\frac{1}{\lambda} - 2\right)\mathbf{I}_p + \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n(\lambda^2-\eta)}\mathbf{I}_p + \frac{\sigma^2.d}{n(\lambda^2-\eta)}\mathbf{I}_p \\
&= \frac{\mathbf{W}_t^\top\mathbf{W}_t}{\lambda^2} + \frac{\sigma^2.d}{n(\lambda^2-\eta)}\mathbf{I}_p + \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n(\lambda^2-\eta)}\left(1 + \frac{1}{\lambda^2} - \frac{2}{\lambda}\right)\mathbf{I}_p \\
&= \frac{\mathbf{W}_t^\top\mathbf{W}_t}{\lambda^2} + \frac{\sigma^2.d}{n(\lambda^2-\eta)}\mathbf{I}_p + \frac{\operatorname{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n(\lambda^2-\eta)}\left(1 - \frac{1}{\lambda}\right)^2\mathbf{I}_p
\end{aligned}$$

$$= \frac{\mathbf{W}_t^\top \mathbf{W}_t}{\lambda^2} + \frac{\sigma^2.d}{n(\lambda^2-\eta)}\mathbf{I}_p + \frac{(\lambda-1)^2}{\lambda^2}\frac{\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{n(\lambda^2-\eta)}\mathbf{I}_p$$

Thus:

$$\begin{aligned}\mathrm{Tr}(A_1) &= \frac{\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{\lambda^2} + \frac{\sigma^2.d\eta}{\lambda^2-\eta} + \frac{\eta(\lambda-1)^2}{\lambda^2}\frac{\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top)}{\lambda^2-\eta} \\ &= \frac{(\lambda(1+\eta)-2\eta)}{\lambda(\lambda^2-\eta)}\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top) + \frac{\sigma^2.d.\eta}{\lambda^2-\eta}\end{aligned}$$

Hence:

$$\boxed{\mathrm{Tr}(A_1) = \frac{(\lambda(1+\eta)-2\eta)}{\lambda(\lambda^2-\eta)}\mathrm{Tr}(\mathbf{W}_t\mathbf{W}_t^\top) + \frac{\sigma^2.d.\eta}{\lambda^2-\eta}}$$

Now let us compute the term $A_2 = \frac{2\alpha\gamma}{n}\mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top\mathbf{Y}\mathbf{X}^\top\mathbf{Q}]$. We have that:

$$\begin{aligned}A_2 &= \frac{2\alpha\gamma}{n}\mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top\mathbf{Y}\mathbf{X}^\top\mathbf{Q}] \\ &= \frac{2\alpha\gamma}{n}\sum_{i=1}^n \mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}] \\ &= \frac{2\alpha\gamma}{n(1+\delta)}\sum_{i=1}^n \mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] \\ &= \frac{2\alpha\gamma}{n(1+\delta)}\sum_{i=1}^n \mathbb{E}\left[\left(\mathbf{Q}_{-i} - \frac{\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}}{n(1+\delta)}\right)\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\right] \\ &= \frac{2\alpha\gamma}{n(1+\delta)}\sum_{i=1}^n \mathbb{E}\left[\mathbf{Q}_{-i}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\right] - \frac{2\alpha\gamma}{n^2(1+\delta)^2}\sum_{i=1}^n \mathbb{E}\left[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\right] \\ &= \frac{2\alpha\gamma}{(1+\delta)}\mathbb{E}[\mathbf{Q}_{-i}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] - \frac{2\alpha\gamma}{n(1+\delta)^2}\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] \\ &= T_1 - T_2\end{aligned}$$

We have that:

$$\begin{aligned}T_1 &= \frac{2\alpha\gamma}{(1+\delta)}\mathbb{E}[\mathbf{Q}_{-i}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}] \\ &= \frac{2\alpha\gamma}{(1+\delta)}\mathbb{E}[\mathbf{Q}_{-i}\mathbf{W}_s^\top \mathbf{W}_t\mathbf{Q}_{-i}] \\ &= \frac{2\alpha\gamma}{(1+\delta)}\frac{(1+\delta)^2}{\lambda^2}\left(\mathbf{W}_s^\top\mathbf{W}_t + \frac{\mathrm{Tr}(\mathbf{W}_s^\top\mathbf{W}_t)}{n(\lambda^2-\eta)}\mathbf{I}_p\right) \\ &= \frac{2\alpha\gamma(1+\delta)}{\lambda^2}\left(\mathbf{W}_s^\top\mathbf{W}_t + \frac{\mathrm{Tr}(\mathbf{W}_s^\top\mathbf{W}_t)}{n(\lambda^2-\eta)}\mathbf{I}_p\right)\end{aligned}$$

And:

$$\begin{aligned}T_2 &= \frac{2\alpha\gamma}{n(1+\delta)^2}\mathbb{E}\left[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\mathbf{W}_s^\top \boldsymbol{y}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\right] \\ &= \frac{2\alpha\gamma}{n(1+\delta)^2}\mathbb{E}[\mathbf{Q}_{-i}\boldsymbol{x}_i\boldsymbol{x}_i^\top\mathbf{Q}_{-i}]\,\mathrm{Tr}(\mathbb{E}[\boldsymbol{x}_i^\top\mathbf{Q}_{-i}\mathbf{W}_s^\top \boldsymbol{y}_i]) \\ &= \frac{2\alpha\gamma}{n(1+\delta)^2}\mathbb{E}[\mathbf{Q}^2]\,\mathrm{Tr}(\mathbf{W}_t\bar{\mathbf{Q}}\mathbf{W}_s^\top)\end{aligned}$$

$$
\begin{aligned}
&= \frac{2\alpha\gamma}{n(1+\delta)^2} \frac{(1+\delta)}{\lambda} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top) \, \mathbb{E}[\mathbf{Q}^2] \\
&= \frac{2\alpha\gamma}{n\lambda(1+\delta)} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top) \, \mathbb{E}[\mathbf{Q}^2] \\
&= \frac{2\alpha\gamma}{n\lambda(1+\delta)} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top) \frac{(1+\delta)^2}{\lambda^2 - \eta} \mathbf{I}_p \\
&= \frac{2\alpha\gamma(1+\delta)}{n\lambda(\lambda^2 - \eta)} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top) \mathbf{I}_p
\end{aligned}
$$

Then:

$$
\operatorname{Tr}(A_2) = \operatorname{Tr}(T_1 - T_2) = \frac{2\alpha\gamma(1+\delta)(\lambda - \eta)}{\lambda(\lambda^2 - \eta)} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top)
$$

Finally, we need to compute the last term: $A_3 = (\alpha\gamma)^2 \, \mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top \mathbf{W}_s \mathbf{Q}]$. We have that:

$$
\begin{aligned}
A_3 &= (\alpha\gamma)^2 \, \mathbb{E}[\mathbf{Q}\mathbf{W}_s^\top \mathbf{W}_s \mathbf{Q}] \\
&= (\alpha\gamma)^2 \frac{(1+\delta)^2}{\lambda^2} \left(\mathbf{W}_s^\top \mathbf{W}_s + \frac{\operatorname{Tr}(\mathbf{W}_s \mathbf{W}_s^\top)}{n(\lambda^2 - \eta)} \mathbf{I}_p \right)
\end{aligned}
$$

Thus:

$$
\begin{aligned}
\operatorname{Tr}(A_3) &= \frac{(\alpha\gamma(1+\delta))^2}{\lambda^2} \left(1 + \frac{\eta}{\lambda^2 - \eta} \right) \operatorname{Tr}(\mathbf{W}_s \mathbf{W}_s^\top) \\
&= \frac{(\alpha\gamma(1+\delta))^2}{\lambda^2 - \eta} \operatorname{Tr}(\mathbf{W}_s \mathbf{W}_s^\top)
\end{aligned}
$$

Now let us write the test error E_{test} in the following form:

$$
E_{\text{test}} = T_1 + \alpha T_2 + \alpha^2 T_3 \tag{47}
$$

Constant term T_1. We have that:

$$
\begin{aligned}
T_1 &= \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_t^\top) + \sigma^2.d - \frac{2}{\lambda} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_t^\top) + \frac{\lambda(1+\eta) - 2\eta}{\lambda(\lambda^2 - \eta)} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_t^\top) + \frac{\sigma^2.d.\eta}{\lambda^2 - \eta} \\
&= \left(1 - \frac{2}{\lambda} + \frac{\lambda(1+\eta) - 2\eta}{\lambda(\lambda^2 - \eta)} \right) \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_t^\top) + \sigma^2.d \left(1 + \frac{\eta}{\lambda^2 - \eta} \right) \\
&= \frac{(\lambda - 1)^2}{\lambda^2 - \eta} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_t^\top) + \frac{\sigma^2.d.\lambda^2}{\lambda^2 - \eta}
\end{aligned}
$$

Linear term T_2. We have that:

$$
\begin{aligned}
T_2 &= \frac{2\gamma(1+\delta)(\lambda - \eta)}{\lambda(\lambda^2 - \eta)} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top) - \frac{2\gamma(1+\delta)}{\lambda} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top) \\
&= \frac{2\gamma(1+\delta)(1 - \lambda)}{\lambda^2 - \eta} \operatorname{Tr}(\mathbf{W}_t \mathbf{W}_s^\top)
\end{aligned}
$$

Quadratic term T_3. We have that:

$$
T_3 = \frac{(\gamma(1+\delta))^2}{\lambda^2 - \eta} \operatorname{Tr}(\mathbf{W}_s \mathbf{W}_s^\top)
$$

E.3 Optimal scaling parameter

The goal is to find a parameter α that minimizes the test error $E_{\text{test}} = T_1 + \alpha T_2 + \alpha^2 T_3$. The objective function has a unique extremum point α given by:

$$\alpha^* = -\frac{T_2}{2T_3}$$

By replacing T_2 and T_3 by their corresponding values, we get that:

$$\boxed{\alpha^* = \frac{\text{Tr}(\mathbf{W}_t \mathbf{W}_s^\top)}{\text{Tr}(\mathbf{W}_s \mathbf{W}_s^\top)}} \tag{48}$$

This result is counter-intuitive, as the optimal α^* does **not** depend on the number of finetuning samples n.

F LLMs experimental details

F.1 Hyperparameters

In this section, we summarize all the details about our experiments on Fine-tuning `roberta-base` model on GLUE tasks. Let us define some notations first then give their corresponding values in each experiment: `lora_r` denotes the rank of LoRA modules, `lora_alpha` denotes the LoRA scaling parameter, `lr_adapter` means the learning rate used to train LoRA modules, `batch_size` and `batch_alpha` is the training batch size for LoRA modules and the vectors $\boldsymbol{\alpha}$ respectively, `lr_alpha` is the learning rate used to update $\boldsymbol{\alpha}$, `optim_alpha` is the optimizer used to train the vectors $\boldsymbol{\alpha}$, `val_split` is the percentage of the training set used to train $\boldsymbol{\alpha}$.

Common to all experiments. We optimize the LoRA modules using `AdamW` for all the benchmarks and with a linear scheduler for the learning rate. We initialize the vectors $\boldsymbol{\alpha}$ to the vector $\mathbf{1}$. The target modules are: the final classifier layer `classifier` (full training) and the attention modules `query` and `value` (Low Rank Adaptation).

F.2 Values of scaling parameters

We report in the following plots some metrics (mean, standard deviation, percentiles) describing the obtained values of the vectors $\boldsymbol{\alpha}$ for each module after the training phase.

Parameter	Value
optimizer	AdamW
LoRA Arguments	
lora_r	8
lora_alpha	8
lr_adapter	10^{-4}
Trainer Arguments	
n_epochs	10
batch_size	64
optim_alpha	AdamW
batch_alpha	64
lr_alpha	10^{-2}
T	1
val_split	1
seeds	1, 5, 123

Table 3: Implementation Details for the fine-tuning experiment on MNLI.

Parameter	Value
optimizer	AdamW
LoRA Arguments	
lora_r	8
lora_alpha	8
lr_adapter	10^{-4} for LoRA and 2.10^{-4} for α-LoRA
Trainer Arguments	
n_epochs	10
batch_size	64
optim_alpha	Adam
batch_alpha	64
lr_alpha	5.10^{-3}
T	20
val_split	0.2
seeds	1, 3, 123

Table 4: Implementation Details for the fine-tuning experiment on QNLI.

Parameter	Value
optimizer	AdamW
LoRA Arguments	
lora_r	8
lora_alpha	8
lr_adapter	10^{-4} for LoRA and 2.10^{-4} for α-LoRA
Trainer Arguments	
n_epochs	40
batch_size	64
optim_alpha	Adam
batch_alpha	64
lr_alpha	5.10^{-3}
T	20
val_split	0.2
seeds	3, 5, 123

Table 5: Implementation Details for the fine-tuning experiment on MRPC.

Parameter	Value
optimizer	AdamW
LoRA Arguments	
lora_r	8
lora_alpha	8
lr_adapter	10^{-4}
Trainer Arguments	
n_epochs	40
batch_size	64
optim_alpha	AdamW
batch_alpha	64
lr_alpha	5.10^{-3}
T	20
val_split	0.8 (and 0.2 for seed 123)
seeds	3, 5, 123

Table 6: Implementation Details for the fine-tuning experiment on RTE.

Parameter	Value
optimizer	AdamW
LoRA Arguments	
lora_r	8
lora_alpha	8
lr_adapter	10^{-4} for LoRA and 2.10^{-4} for α-LoRA
Trainer Arguments	
n_epochs	10
batch_size	128
optim_alpha	AdamW
batch_alpha	128
lr_alpha	5.10^{-3}
T	10 (and 20 for seed 5)
val_split	0.5 (and 0.9 for seed 5)
seeds	1, 3, 5

Table 7: Implementation Details for the fine-tuning experiment on SST2.

Parameter	Value
optimizer	AdamW
LoRA Arguments	
lora_r	8
lora_alpha	8
lr_adapter	5.10^{-4}
Trainer Arguments	
n_epochs	5
batch_size	256
optim_alpha	Adam, AdamW (seed 123)
batch_alpha	64
lr_alpha	5.10^{-3}
T	1 (seed 3), 10 (seed 5) and 20 (seed 123)
val_split	0.8
seeds	3, 5, 123

Table 8: Implementation Details for the fine-tuning experiment on QQP.

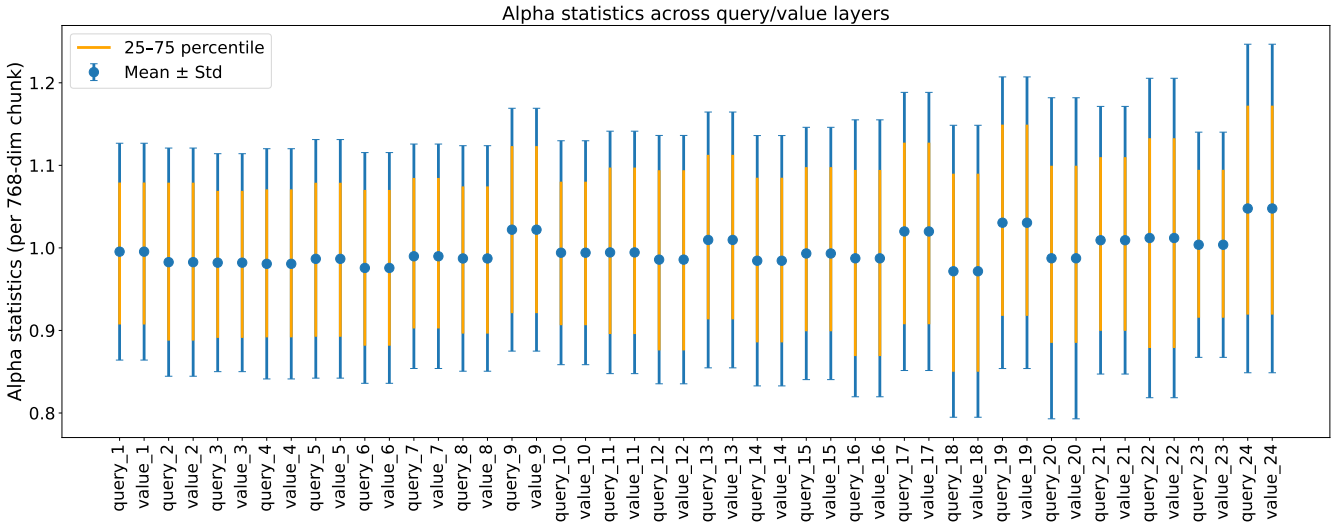

Figure 10: Statistics of the vectors $\boldsymbol{\alpha}$ for the QNLI benchmark

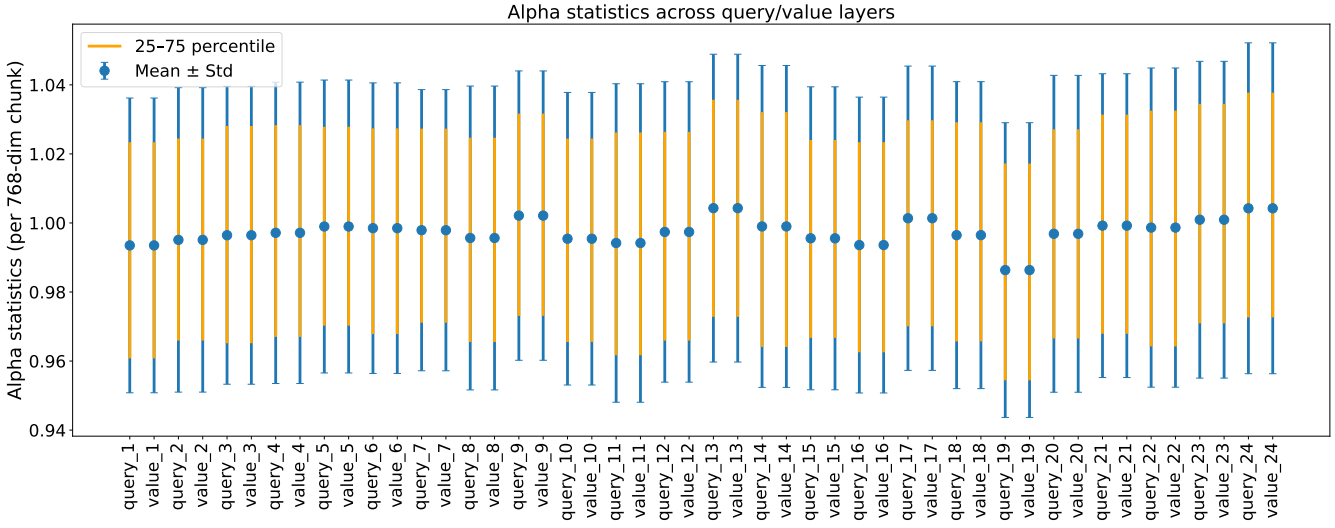

Figure 11: Statistics of the vectors $\boldsymbol{\alpha}$ for the RTE benchmark