Universiteit Leiden

Order-Regularized Fourier Features for Spectral Bias and Adversarial Robustness

Bram Westerbeek (s4104870)

Supervisors:
Anna Dawid-Łękowska & Juan Agustin Duque

Bachelor Thesis – Data Science and Artificial Intelligence

Leiden Institute of Advanced Computer Science (LIACS)
liacs.leidenuniv.nl

July 9, 2026

Abstract

Deep neural networks perform well on image classification tasks, but their training dynamics and their vulnerability to adversarial perturbations are still not fully understood. One possible explanation is the frequency structure of learned representations: neural networks often show a spectral bias, where low-frequency components are learned before high-frequency components. At the same time, adversarial examples are often connected to brittle or high-frequency input patterns. This thesis investigates whether explicitly controlling the order in which Fourier frequencies are introduced during training leads to different classification performance, training dynamics, and adversarial robustness. A simple order-regularized Fourier feature model is implemented by transforming images to the Fourier domain, applying an epoch-dependent radial low-frequency mask, and training a logistic regression classifier on the resulting magnitude features. The model is compared with pixel logistic regression, unrestricted Fourier logistic regression, a multilayer perceptron, and a small convolutional neural network on MNIST and CIFAR-10. The experiments show that the neural network baselines, especially the convolutional model, achieve the strongest classification accuracy. For robustness, the CNN is strongest on MNIST, while all models remain highly vulnerable on CIFAR-10 under the evaluated attacks. The order-regularized Fourier model does not outperform the neural baselines and does not provide a clear adversarial robustness advantage. However, on CIFAR-10 it improves over unrestricted Fourier logistic regression and pixel logistic regression, and the frequency-band analysis shows that the proposed training schedule changes the distribution of learned Fourier weights. The main contribution is therefore an experimental study of explicit spectral ordering as an interpretable tool for analysing frequency-dependent learning behaviour rather than a new state-of-the-art classifier.

Contents

1 Introduction

Deep neural networks have become a standard approach for image classification. Convolutional neural networks in particular are able to learn useful visual features directly from pixels and have achieved strong results on a wide range of recognition tasks [9, 4]. Still, their behaviour during training is not completely transparent. Modern neural networks can fit complicated mappings, including random labels, while still generalising well in many practical settings [20]. This makes it difficult to explain which properties of the data are learned first, which features are used for classification, and why trained models can be sensitive to small changes in the input.

A useful perspective on this issue is the frequency domain. Several studies have found that neural networks trained with gradient-based optimisation tend to learn low-frequency components before high-frequency components. This phenomenon is commonly called spectral bias or the frequency principle [14, 18, 1]. Low-frequency structure usually corresponds to smoother global patterns, while high-frequency structure corresponds to more local and rapidly changing patterns. In image data, this distinction is relevant because object shape, texture, edges, and noise can all be described partly in terms of frequency content.

Adversarial robustness is a second motivation for studying frequencies. Deep learning models can be fooled by small perturbations that are almost invisible to humans but lead to wrong predictions [15, 3]. Goodfellow et al. introduced the Fast Gradient Sign Method (FGSM) as a simple one-step way to construct such adversarial examples [3]. Madry et al. later framed adversarial training and Projected Gradient Descent (PGD) attacks in terms of robust optimisation [11]. Because adversarial perturbations are often small and visually subtle, it is natural to ask whether the frequency structure of a model's representation affects its robustness.

This thesis studies a controlled model that explicitly regulates the frequency order during training. Each image is transformed to the two-dimensional Fourier domain. A radial mask keeps only frequencies up to a radius r, and this radius is increased over training epochs. The resulting representation is used as input to a logistic regression classifier. The purpose is not to build a complex model, but to create a simple experimental setting where the frequency order is controlled directly. This model is compared with pixel logistic regression, unrestricted Fourier logistic regression, a multilayer perceptron (MLP), and a small convolutional neural network (CNN).

The main research question is:

> *How does an order-regularized Fourier feature model compare to standard neural networks in classification performance, training dynamics, and adversarial robustness?*

This question is split into three sub-questions:

RQ1. What classification performance does the order-regularized Fourier model achieve on MNIST and CIFAR-10?

RQ2. How do its training dynamics compare to logistic regression baselines, an MLP, and a CNN?

RQ3. Does explicit spectral ordering improve robustness against FGSM and PGD adversarial attacks?

The contribution of the thesis is experimental. The work implements a complete PyTorch pipeline for Fourier preprocessing, order-regularized training, baseline comparison, frequency-band analysis, and adversarial evaluation. The results show that the order-regularized Fourier model should not be presented as a replacement for neural networks. The CNN is clearly strongest on both datasets. Nevertheless, the Fourier model provides a transparent way to inspect how frequency restrictions affect performance and learned weights.

The remainder of this thesis is structured as follows. Section 2 introduces the relevant background on image classification, Fourier representations, spectral bias, and adversarial examples. Section 3 discusses related work. Section 4 describes the proposed model and evaluation method. Section 5 gives implementation and training details. Section 6 presents the results. Section 7 discusses the interpretation and limitations. Section 8 concludes and suggests directions for further research.

2 Background

2.1 Image classification

Image classification is the task of assigning an input image x to one of several classes. In supervised learning, a model is trained on labelled examples (x_i, y_i) and is evaluated on unseen test data. A simple baseline is multinomial logistic regression, where the input is flattened into a vector and mapped linearly to class scores. Although logistic regression is limited because it cannot learn nonlinear visual features, it is useful as a transparent baseline [2, 12].

Multilayer perceptrons extend linear models by introducing hidden layers and nonlinear activation functions. They can learn nonlinear decision boundaries, but when applied directly to flattened images they do not explicitly use the spatial structure of the input. CNNs are designed for images because convolutional filters are applied locally and shared over spatial positions. This allows a CNN to learn local visual patterns such as edges and textures efficiently [9]. For this reason, the CNN is expected to be the strongest model in this thesis.

2.2 Fourier representation of images

The two-dimensional discrete Fourier transform decomposes an image into sinusoidal components with different spatial frequencies. For an image x with height H and width W, its Fourier representation can be written as

$$\hat{x}(u, v) = \sum_{m=0}^{H-1} \sum_{n=0}^{W-1} x(m, n) e^{-2\pi i(um/H + vn/W)}. \tag{1}$$

Low frequencies describe slowly varying global structure, while high frequencies describe rapidly changing local structure. In images, low frequencies often correspond to broad shapes and smooth colour or intensity changes, whereas high frequencies often correspond to edges, texture, and noise. Fourier features are therefore a useful tool for studying whether a model depends mostly on global or local image structure.

In this thesis, the Fourier transform is used as a preprocessing step. The complex Fourier coefficients are converted to magnitudes and flattened into a feature vector. This means that the Fourier models classify images using frequency magnitude information rather than the original pixel values. This choice makes the model simple and interpretable, although it also removes phase information, which can be important for reconstructing spatial structure.

2.3 Spectral bias and the frequency principle

Spectral bias refers to the tendency of neural networks to learn low-frequency components before high-frequency components during gradient-based training. Rahaman et al. show that deep networks have a bias towards low-frequency functions, meaning that smoother components are often fitted earlier than more rapidly varying components [14]. Xu et al. describe a similar phenomenon as the frequency principle and observe that neural networks often capture dominant low-frequency components first [18]. Basri et al. further analyse how data density can affect the speed at which frequencies are learned [1].

This idea is relevant to generalisation. Low-frequency functions are usually smoother and may capture more stable patterns in the data. High-frequency functions can represent fine details, but they can also fit noise or brittle patterns. However, the relation is not absolute: high-frequency information can also be useful for classification, especially in natural images where texture and edges carry class information. Therefore, the purpose of this thesis is not to assume that low frequencies are always better, but to test what happens when frequency learning order is controlled explicitly.

Figure 1 gives a schematic illustration of this idea. It does not represent a measured result from the experiments, but introduces the learning pattern discussed in this section.

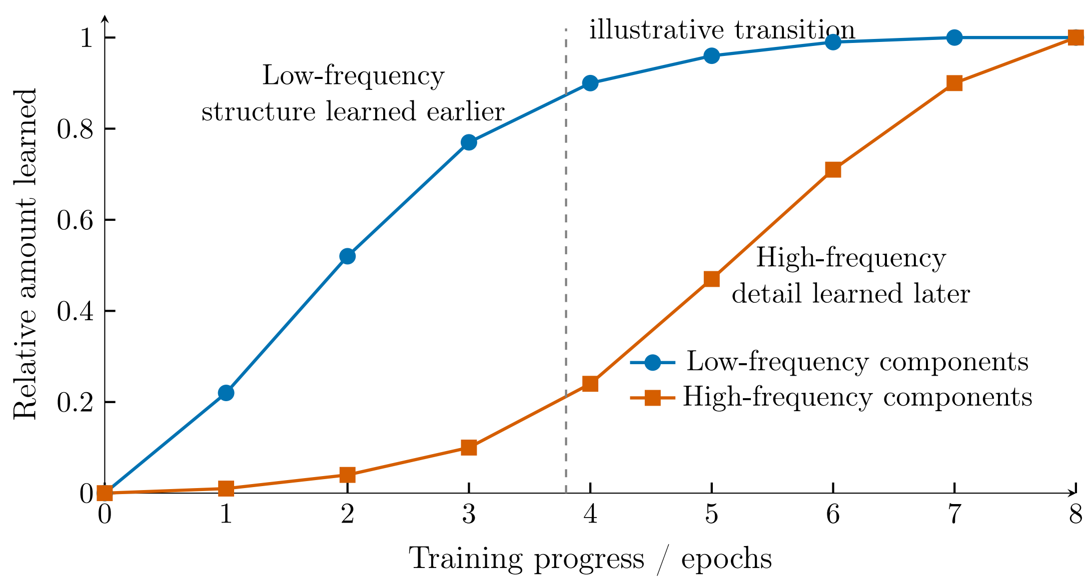


Figure 1: Conceptual illustration of spectral bias. Neural networks are commonly observed to fit lower-frequency structure earlier in training, while higher-frequency components are learned later. The figure is schematic and is included to illustrate the phenomenon discussed in Section 2.3.

2.4 Adversarial examples and robustness

Adversarial examples are inputs that are intentionally perturbed to cause a model to make an incorrect prediction. Szegedy et al. first showed that neural networks can be vulnerable to small perturbations that transfer across models [15]. Goodfellow et al. argued that this vulnerability can be partly explained by the locally linear behaviour of high-dimensional models and introduced

FGSM [3]. For an input image x, label y, model parameters θ, loss function J, and perturbation budget ϵ, FGSM constructs an adversarial image as

$$x_{adv} = x + \epsilon \cdot \text{sign}(\nabla_x J(\theta, x, y)). \tag{2}$$

PGD is a stronger multi-step attack. Starting from an image, the attack repeatedly updates the image in the gradient direction and projects the perturbation back into an allowed L_∞ ball around the original input [11]. In this thesis, FGSM and PGD are used only for evaluation, not for adversarial training.

The connection between robustness and frequencies is still an active research topic. Yin et al. use a Fourier perspective to study robustness trade-offs in computer vision and show that robustness improvements can differ between high-frequency and low-frequency corruptions [19]. Ilyas et al. argue that adversarial examples can arise from non-robust features that are predictive but brittle [5]. These studies motivate the question of whether explicitly restricting the frequency content during training can affect adversarial robustness.

Figure 2 illustrates the basic idea behind adversarial examples. A small perturbation is added to a clean input. Although the resulting image can look almost unchanged to a human observer, the model prediction may change.

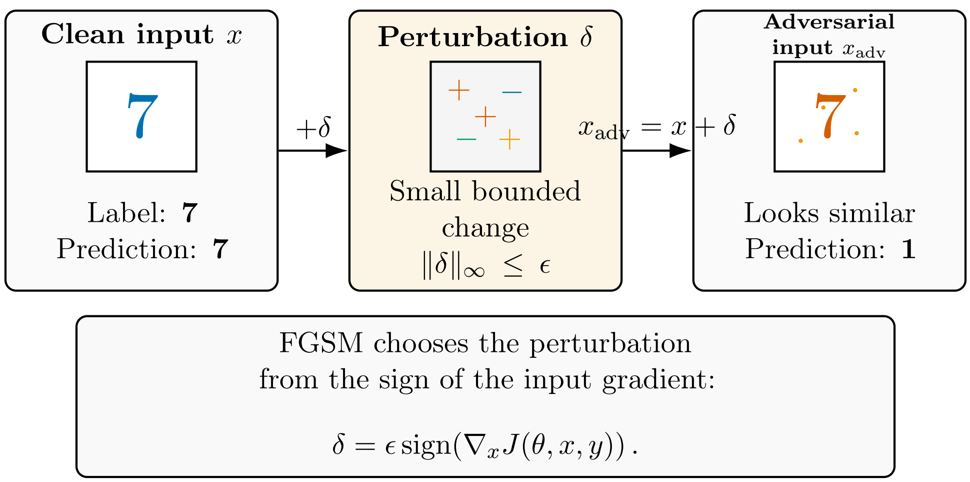


Figure 2: Conceptual illustration of an adversarial example. A clean input x is combined with a small bounded perturbation δ to form an adversarial input x_{adv}. Although the perturbed image can look visually similar to the original image, the model prediction may change.

3 Related Work

The first group of related work concerns spectral bias. Rahaman et al. provide one of the central references by showing that neural networks prefer low-frequency functions during training [14]. Xu et al. study training behaviour in the frequency domain and describe the low-to-high learning pattern as the frequency principle [18]. Basri et al. add that frequency bias can depend on input density, which is relevant because real image datasets are not uniformly distributed in input space [1]. These works mostly analyse neural networks directly. The present thesis differs by implementing a simple Fourier-domain classifier whose frequency access is controlled explicitly by a radial schedule.

The second group concerns Fourier features and Fourier-domain learning. Tancik et al. show that Fourier feature mappings can help MLPs learn high-frequency functions in low-dimensional domains [16]. Luo et al. formulate supervised learning in the Fourier domain and analyse well-posedness properties [10]. Jo and Bengio use Fourier filtering to study whether CNNs rely on surface statistical regularities in image datasets [6]. These works support the idea that Fourier analysis is useful not only for signal processing, but also for understanding what machine learning models learn from images.

The third group concerns adversarial examples. Szegedy et al. showed that neural networks can be fooled by small, carefully chosen perturbations [15]. Goodfellow et al. introduced FGSM and linked adversarial vulnerability to linear behaviour in high dimensions [3]. Madry et al. proposed a robust optimisation view of adversarial robustness and used PGD-style attacks as strong first-order adversaries for evaluating and training robust models [11]. Later work by Ilyas et al. and Tsipras et al. suggests that robustness is not just a technical problem, but may involve different feature representations and a possible trade-off with standard accuracy [5, 17].

The final group concerns datasets and implementation tools. MNIST is a classic handwritten digit dataset and is closely connected to early CNN work [9]. CIFAR-10 is a more complex natural image benchmark with ten classes and small colour images [8]. PyTorch is used for the implementation because it provides automatic differentiation and a flexible imperative programming style [13]. Adam is used as the optimiser because it is a common adaptive first-order optimisation method [7].

This thesis combines these lines of work in a small but controlled experimental study. The aim is not to introduce a new high-performance architecture, but to test whether explicit low-to-high Fourier ordering produces measurable differences in accuracy, training dynamics, learned frequency weights, and adversarial robustness.

4 Methodology

4.1 Datasets

Two datasets are used: MNIST and CIFAR-10. MNIST contains grayscale handwritten digit images of size 28×28 with ten classes. It is relatively simple and has strong low-frequency structure. CIFAR-10 contains colour images of size 32×32 with ten object classes. It is more difficult because the images have more variation, background clutter, colour information, and texture. Using both datasets makes it possible to test whether the Fourier model behaves differently on a simple digit dataset and a more complex natural image dataset.

For MNIST, the original training set of 60,000 images is split into 50,000 training images and 10,000 validation images. The standard 10,000-image test set is used for final evaluation. For CIFAR-10, the original training set of 50,000 images is split into 40,000 training images and 10,000 validation images. The standard 10,000-image test set is used for final evaluation. The same split is used for all models.

Figure 3 shows representative examples from both datasets. The MNIST examples are simple grayscale digit images, while the CIFAR-10 examples contain colour, texture, background variation, and more complex object structure.

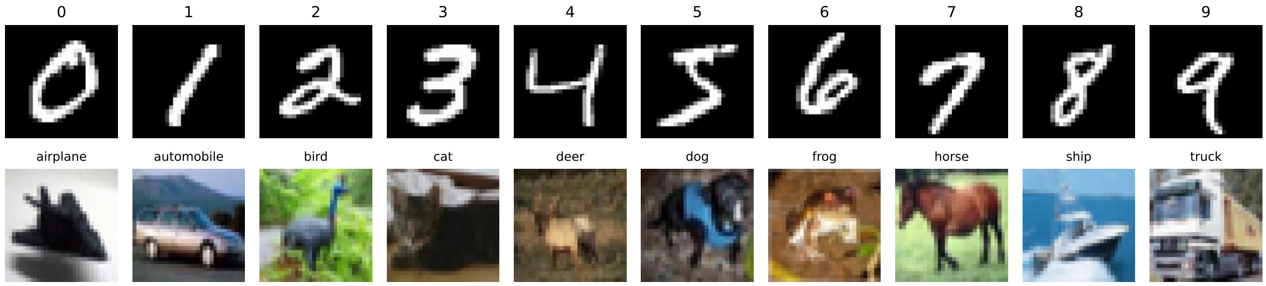


Figure 3: Example images from the MNIST and CIFAR-10 datasets [9, 8]. MNIST consists of 28×28 grayscale handwritten digit images, while CIFAR-10 consists of 32×32 colour natural images from ten object classes.

4.2 Fourier preprocessing

For the Fourier models, each input image is transformed using the two-dimensional Fourier transform. The implementation applies the transform over the spatial dimensions of each channel. The complex coefficients are then converted to magnitudes. If no frequency mask is applied, the full magnitude spectrum is flattened and passed to the classifier.

The radial low-frequency mask is defined as

$$M_r(u,v) = \begin{cases} 1 & \text{if } \sqrt{u^2+v^2} \le r, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Here, u and v denote centred frequency coordinates after shifting the zero-frequency component to the centre, and r is the radius of the included frequency region. Before applying the mask, the Fourier coefficients are shifted so that the zero-frequency component is at the centre. After masking, the coefficients are shifted back. The final Fourier feature vector is

$$\phi_r(x) = \text{flatten}\left(|M_r \odot \hat{x}|\right), \tag{4}$$

where $\odot$ denotes element-wise multiplication. If r is omitted, all frequencies are used.

Figure 4 illustrates the effect of this radial masking procedure on an MNIST image. Small radius values retain only coarse low-frequency structure, while larger radius values gradually restore sharper digit details.

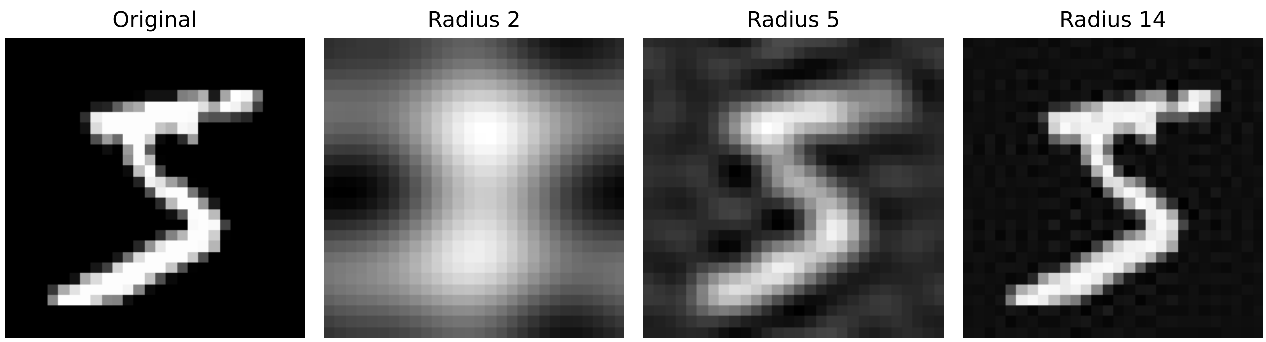


Figure 4: Example of radial low-frequency Fourier masking on an MNIST image. Smaller radii preserve only coarse low-frequency structure, while larger radii gradually restore sharper digit details.

4.3 Order-regularized Fourier model

The proposed model is a logistic regression classifier trained on masked Fourier magnitude features. Its main difference from standard Fourier logistic regression is that the mask radius changes during training. At epoch t, the model uses radius $r(t)$. The schedule is linear from a minimum radius r_{min} to a maximum radius r_{max}:

$$r(t) = \text{round}\left(r_{min} + \frac{t}{T-1}(r_{max} - r_{min})\right), \tag{5}$$

where $t = 0, \ldots, T-1$ and T is the total number of epochs. For MNIST, $r_{min} = 2$ and $r_{max} = 14$. For CIFAR-10, $r_{min} = 2$ and $r_{max} = 16$. After training, the model is evaluated using the maximum radius.

This method can be seen as an explicit version of low-to-high frequency learning. Instead of relying on the implicit spectral bias of gradient descent, the model is forced to start with low-frequency information and only later receives higher frequencies. Because the classifier is linear in the selected Fourier magnitude features, the setup remains simple and interpretable.

4.4 Baseline models

Four baselines are used.

1. **Pixel logistic regression:** a linear classifier trained directly on flattened pixel values.

2. **Fourier logistic regression:** a linear classifier trained on the full Fourier magnitude spectrum without order regularization.

3. **MLP:** a two-layer multilayer perceptron with one hidden layer of 256 units and ReLU activation.

4. **CNN:** a small convolutional network with two convolutional layers. The first layer has 32 filters and the second has 64 filters. Each convolution is followed by ReLU activation and max pooling. The resulting feature map is flattened and passed to a final linear classification layer.

The baselines are chosen to cover a range from simple linear models to a spatial neural network. This makes it possible to compare the order-regularized model both against models of similar complexity and against stronger neural baselines.

4.5 Rationale for the experimental design

The order-regularized Fourier model was intentionally kept simple. Instead of combining Fourier features with a deep neural network, the masked Fourier representation is passed to a logistic regression classifier. This design makes the experiment easier to interpret. If a deep network were used after the Fourier transformation, it would be harder to determine whether any change in performance came from the frequency ordering itself or from the additional nonlinear modelling capacity of the network.

The comparison between pixel logistic regression, unrestricted Fourier logistic regression, and order-regularized Fourier logistic regression therefore isolates three related questions. Pixel logistic regression tests how far a simple linear classifier can get using the original image representation. Fourier logistic regression tests whether a full frequency-domain magnitude representation changes the behaviour of the same type of classifier. The order-regularized Fourier model then tests whether controlling the order in which frequency bands become available affects training dynamics, learned frequency weights, or robustness.

The MLP and CNN serve a different purpose. They are not expected to have the same interpretability as the Fourier logistic models, but they represent stronger neural baselines. In particular, the CNN is included because it is specifically designed for image data and can learn local spatial features. This makes it possible to evaluate whether the proposed frequency-ordering mechanism is merely interesting in isolation, or whether it is competitive with standard neural-network approaches.

4.6 Evaluation metrics

The models are evaluated using final test accuracy, training loss over epochs, validation accuracy over epochs, Fourier frequency-band weight magnitudes, and adversarial robustness curves. For frequency-band analysis, the learned linear weights of the Fourier logistic models are reshaped to the image dimensions and averaged by radial frequency band. The bands used are 0–3, 3–6, 6–10, 10–14, and 14–20.

For adversarial robustness, each trained model is evaluated under FGSM and PGD attacks for $\epsilon \in \{0.00, 0.05, 0.10, 0.15, 0.20, 0.25, 0.30\}$. Accuracy is measured at each perturbation budget. A model is considered more robust within this experiment if its accuracy decreases more slowly as ϵ increases.

For the Fourier-based models, the adversarial attacks are applied in the Fourier feature space rather than directly to the original pixel values. More specifically, the attack perturbs the Fourier magnitude features that are provided to the classifier. The resulting robustness measurements therefore describe vulnerability at the level of the Fourier representation and should not be interpreted as a direct evaluation of pixel-level adversarial robustness for these models.

5 Experimental Setup

All experiments are implemented in PyTorch [13]. The same training procedure is used for all models as much as possible. Images are converted to tensors using `ToTensor()`, which scales pixel values to the interval $[0, 1]$. No additional dataset normalisation or data augmentation is applied. This keeps the setup simple and makes the comparison between pixel and Fourier representations easier to interpret.

Each model is trained for 20 epochs using cross-entropy loss and the Adam optimiser with learning rate 0.001 [7]. The batch size is 128. A fixed random seed of 42 is used for reproducibility. The experiments are single-run experiments, meaning that each model is trained once per dataset. Therefore, the reported results do not include confidence intervals or standard deviations across multiple seeds.

The adversarial evaluation uses the trained models without adversarial training. For the pixel-based and neural-network baselines, FGSM and PGD operate on the image input. For the Fourier-based models, the attacks operate on the Fourier magnitude feature representation supplied to the classifier. FGSM is implemented as a one-step gradient sign attack. PGD is implemented with 10 iterations and step size $\alpha = 0.01$. After each PGD step, the perturbation is clipped to the L_∞ interval $[-\epsilon, \epsilon]$ and the resulting image is clipped to $[0, 1]$. A limitation of this implementation is that for $\epsilon > 0.10$, the fixed setting of 10 steps with $\alpha = 0.01$ does not fully exploit the larger perturbation budget unless the gradient path changes enough across steps. Therefore, PGD results at larger epsilon values should be interpreted as a fixed 10-step evaluation rather than a fully tuned strongest possible PGD attack.

Setting	Value
Framework	PyTorch
Optimizer	Adam
Learning rate	0.001
Loss	Cross-entropy
Batch size	128
Epochs	20
Seed	42
Preprocessing	Tensor conversion to $[0, 1]$
MNIST split	50,000 train / 10,000 validation / 10,000 test
CIFAR-10 split	40,000 train / 10,000 validation / 10,000 test
FGSM epsilons	0.00 to 0.30 in steps of 0.05
PGD	10 steps, $\alpha = 0.01$

Table 1: Main experimental settings.

5.1 Reproducibility and scope of the implementation

The implementation was organised as a full experimental pipeline rather than as separate isolated tests. The pipeline first trains the models, stores the training histories, computes final test accuracy, evaluates adversarial robustness, and then generates the result tables and figures from the saved outputs. This separation between experiment execution and result visualisation is useful because the plots can be regenerated without changing or rerunning the trained models.

The same saved result files were used for the reported tables and figures. Training loss and validation accuracy were stored per epoch, while final accuracy was computed on the held-out test set after training. For the Fourier-based models, the learned classifier weights were additionally summarised over radial frequency bands. This makes the experiment reproducible at the level of the reported outputs: the numerical tables and visualisations are derived from the same stored experimental results.

The scope of the implementation is deliberately limited to standard training followed by post-hoc evaluation. No adversarial training, data augmentation, or extensive hyperparameter tuning was added. This keeps the comparison focused on the representation and model choices. As a result, the experiments should be read as a controlled comparison between model classes under the same basic training conditions, not as an attempt to obtain the highest possible benchmark performance on MNIST or CIFAR-10.

6 Results

6.1 Final test accuracy

Table 2 shows the final test accuracy for all models on MNIST and CIFAR-10. The CNN is the best model on both datasets. On MNIST, the CNN reaches 98.92% accuracy and the MLP reaches 98.06%. Pixel logistic regression also performs reasonably well at 92.81%. The Fourier logistic models are weaker, with 84.04% for unrestricted Fourier logistic regression and 83.26% for the order-regularized Fourier model.

On CIFAR-10, the task is more difficult for all models. The CNN again performs best with 69.12%. The MLP reaches 49.23%. The order-regularized Fourier model reaches 42.19%, which is higher than pixel logistic regression at 39.25% and unrestricted Fourier logistic regression at 35.51%. This suggests that explicit frequency ordering helps the Fourier model on CIFAR-10 compared with the unrestricted Fourier baseline, but it is still far below the neural network models.

Dataset	Model	Test accuracy
MNIST	CNN	98.92%
MNIST	MLP	98.06%
MNIST	Pixel logistic regression	92.81%
MNIST	Fourier logistic regression	84.04%
MNIST	Order-regularized Fourier	83.26%
CIFAR-10	CNN	69.12%
CIFAR-10	MLP	49.23%
CIFAR-10	Order-regularized Fourier	42.19%
CIFAR-10	Pixel logistic regression	39.25%
CIFAR-10	Fourier logistic regression	35.51%

Table 2: Final test accuracy on MNIST and CIFAR-10.

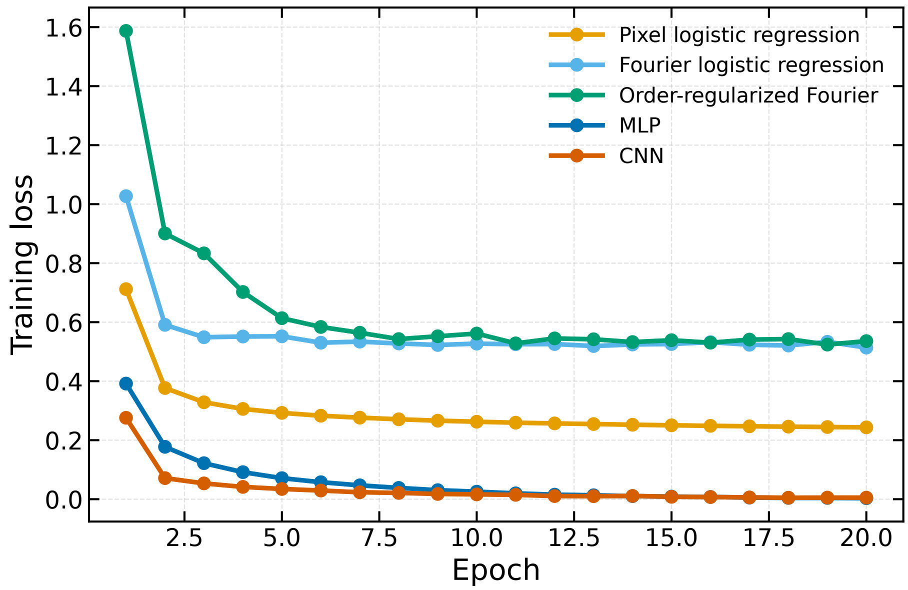
The first research sub-question can therefore be answered as follows. The order-regularized Fourier model achieves 83.26% accuracy on MNIST and 42.19% accuracy on CIFAR-10. Its performance is not competitive with the CNN or MLP on MNIST. On CIFAR-10, it performs better than the two linear baselines, but still not as well as the neural networks.

6.2 Training dynamics

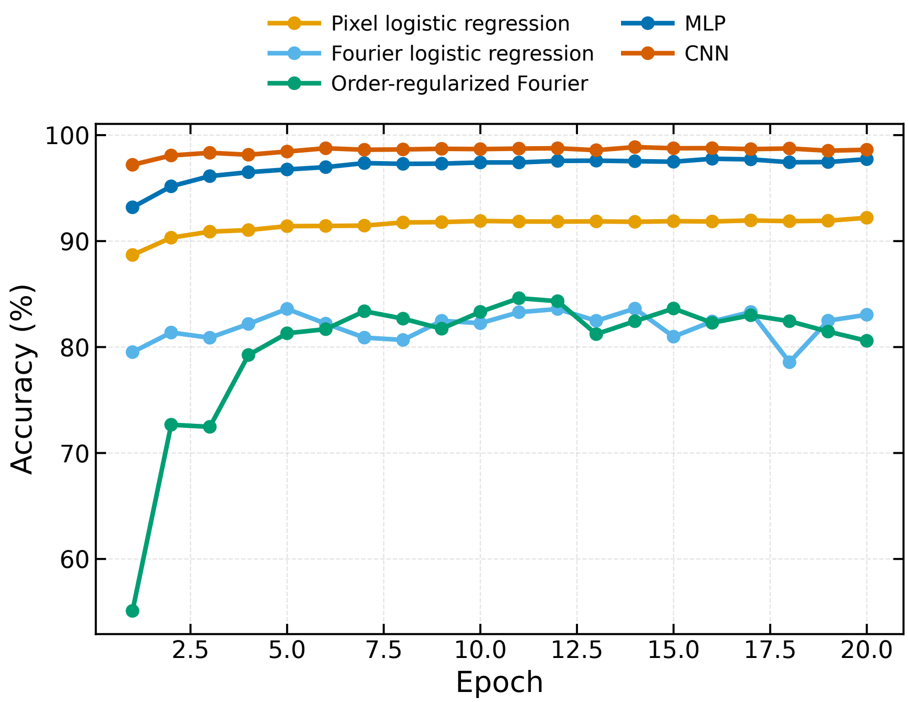
Figures 5 and 6 show the training loss and validation accuracy curves. On MNIST, the CNN and MLP converge quickly and reach high validation accuracy. Pixel logistic regression improves steadily but remains below the neural networks. The Fourier-based logistic models improve more slowly and reach lower validation accuracy. The order-regularized model is also affected by the changing radius schedule: at early epochs it receives only low-frequency features, so its input representation is deliberately incomplete.

On CIFAR-10, the CNN again separates clearly from the other models. Its validation accuracy rises above the MLP and all linear models. The MLP improves but remains limited compared with the CNN. The linear models have much lower validation accuracy, which is expected because CIFAR-10 contains complex colour images that require spatial and nonlinear feature extraction. The order-regularized Fourier model performs better than unrestricted Fourier logistic regression on validation and test accuracy, suggesting that the low-to-high schedule is more useful on CIFAR-10 than on MNIST.

These results answer the second research sub-question. The order-regularized Fourier model learns differently from the unrestricted Fourier baseline because its input feature set changes during training. However, the strongest learning dynamics are observed for the CNN and MLP, especially on MNIST. Explicit spectral ordering does not make the simple Fourier classifier competitive with neural networks.

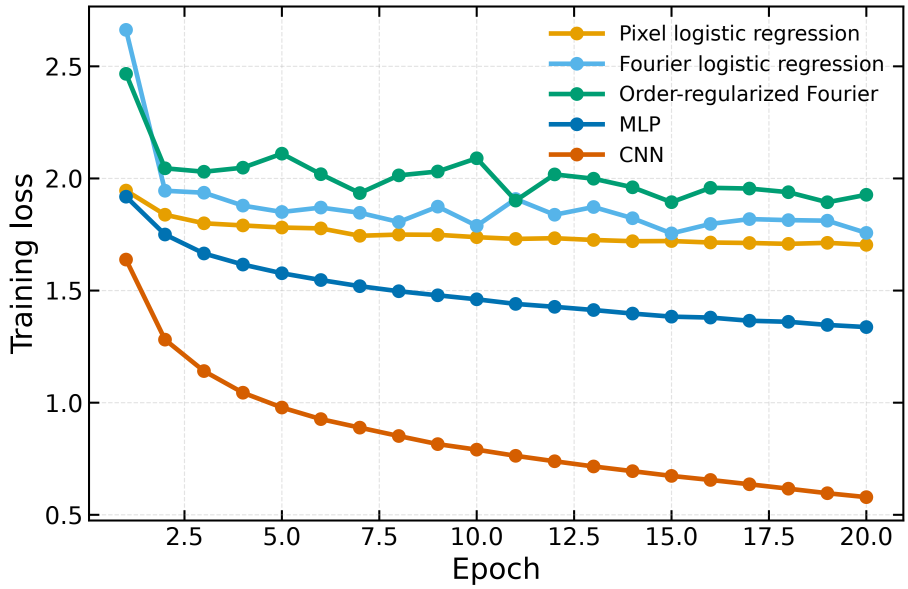


(a) Training loss.

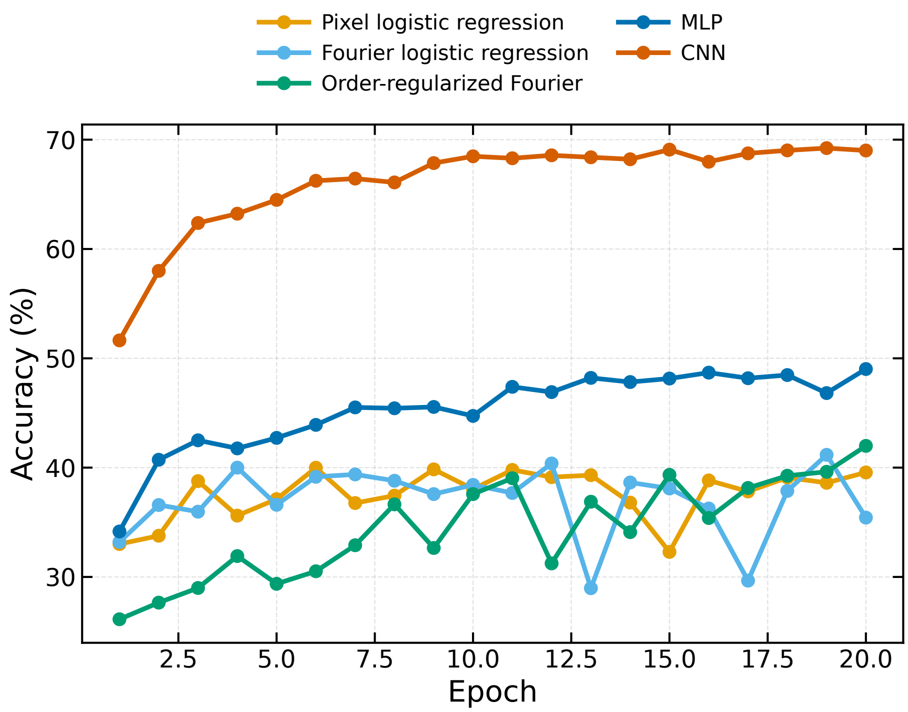


(b) Validation accuracy.

Figure 5: MNIST training dynamics.



(a) Training loss.



(b) Validation accuracy.

Figure 6: CIFAR-10 training dynamics.

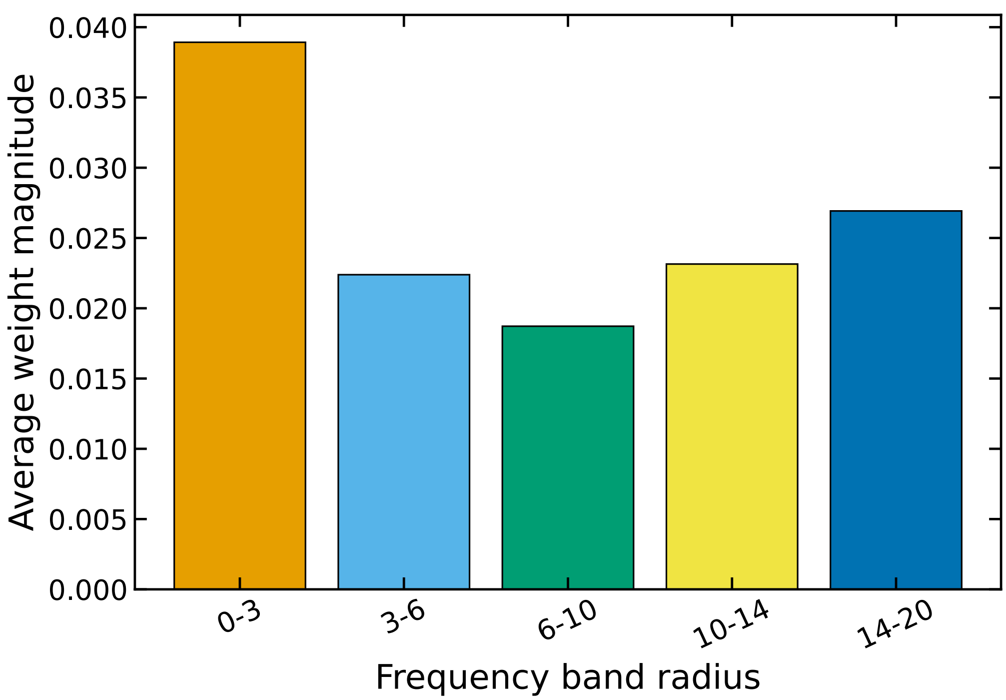
6.3 Frequency-band weight analysis

The frequency-band weight magnitudes provide a more direct view of how the Fourier logistic models use different frequency regions. Table 3 shows the average learned weight magnitude for each frequency band.

Dataset	Model	0–3	3–6	6–10	10–14	14–20
MNIST	Fourier logistic	0.0389	0.0224	0.0187	0.0231	0.0269
MNIST	Order-reg. Fourier	0.0395	0.0223	0.0181	0.0183	0.0180
CIFAR-10	Fourier logistic	0.0073	0.0090	0.0109	0.0136	0.0182
CIFAR-10	Order-reg. Fourier	0.0075	0.0088	0.0103	0.0108	0.0101

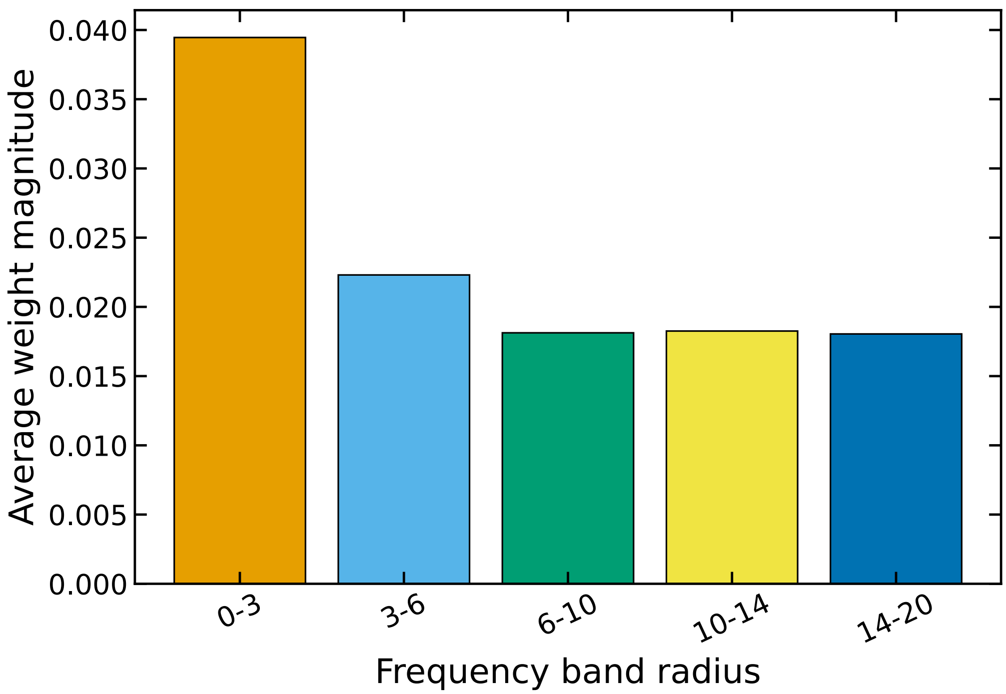
Table 3: Average classifier weight magnitude by radial frequency band.

On MNIST, both Fourier models assign the largest average magnitude to the lowest frequency band, as shown in Figures 7a and 7b. The unrestricted Fourier model also has increased magnitude in the highest band, while the order-regularized model has a flatter and lower magnitude in the high-frequency bands. This is consistent with the intended effect of the radius schedule: the model is encouraged to first use low-frequency information and does not rely as strongly on high-frequency coefficients.



(a) Fourier logistic regression.

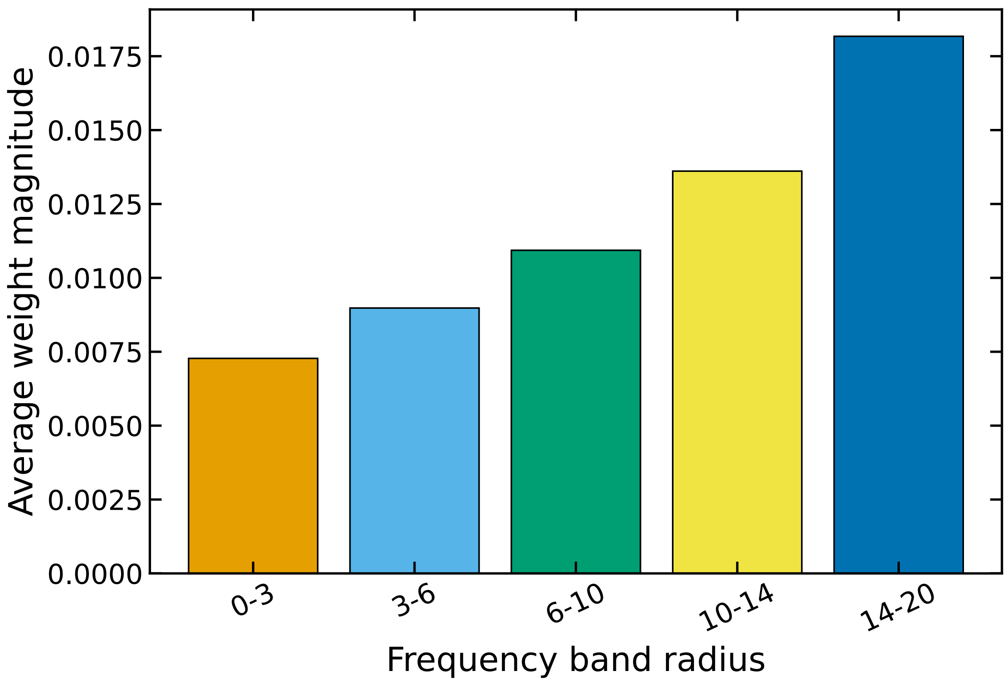
Figure 7: MNIST frequency-band weight magnitudes.



(b) Order-regularized Fourier.

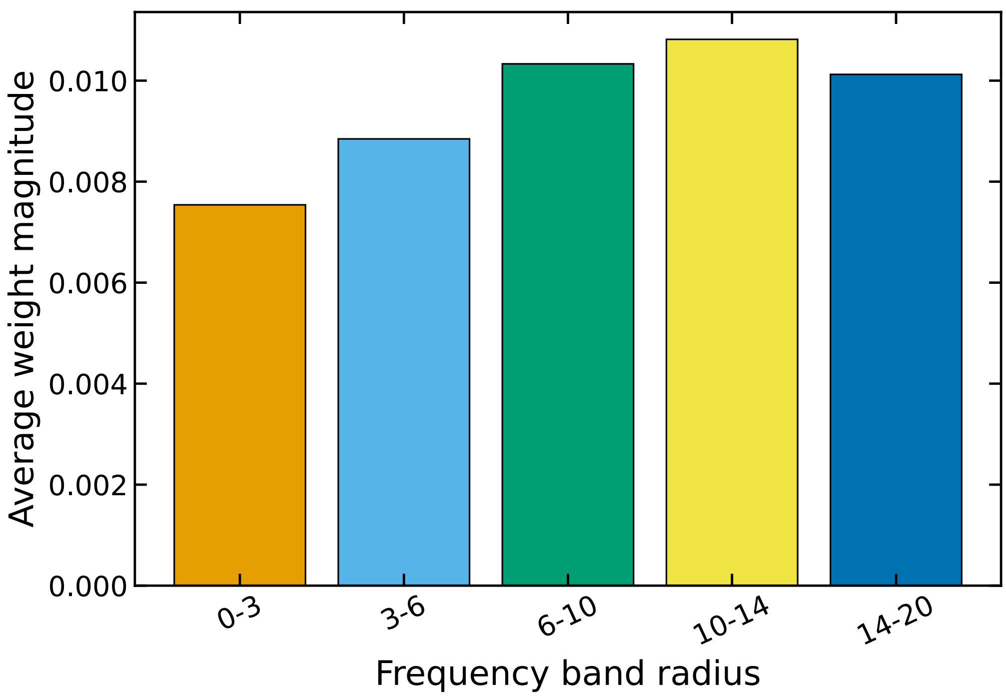
Figure 7: MNIST frequency-band weight magnitudes.

On CIFAR-10, unrestricted Fourier logistic regression shows increasing weight magnitude towards higher frequency bands, as shown in Figures 8a and 8b. This suggests that the classifier makes relatively strong use of high-frequency components. The order-regularized Fourier model has lower high-frequency magnitudes and a less steep increase across bands. This supports the idea that the schedule changes the learned spectral profile. However, this does not automatically translate into better overall accuracy than the neural networks.



(a) Fourier logistic regression.

Figure 8: CIFAR-10 frequency-band weight magnitudes.



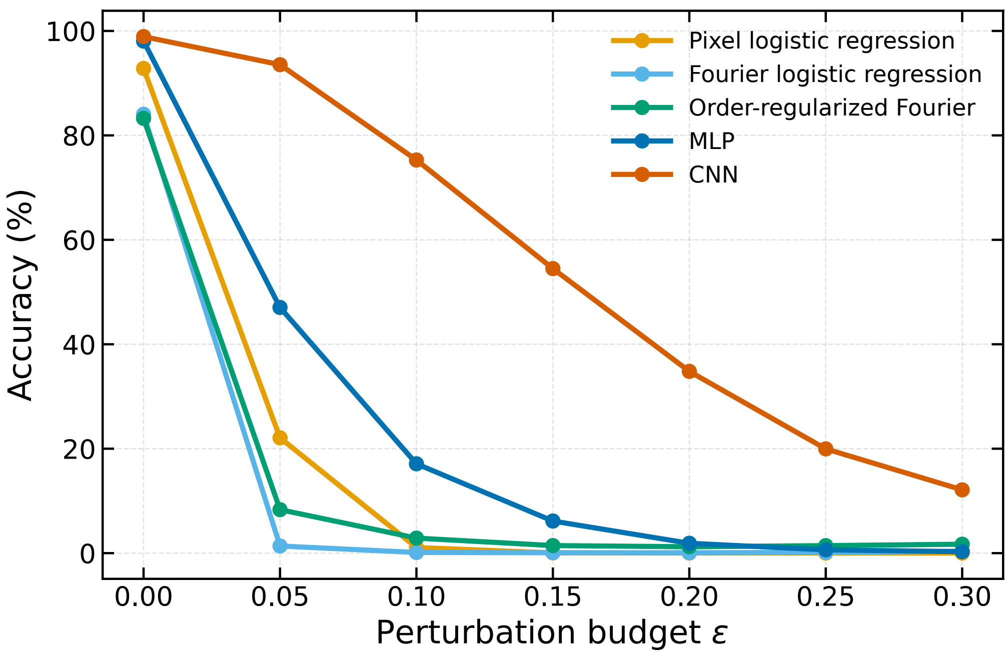
(b) Order-regularized Fourier.

Figure 8: CIFAR-10 frequency-band weight magnitudes.

6.4 FGSM robustness

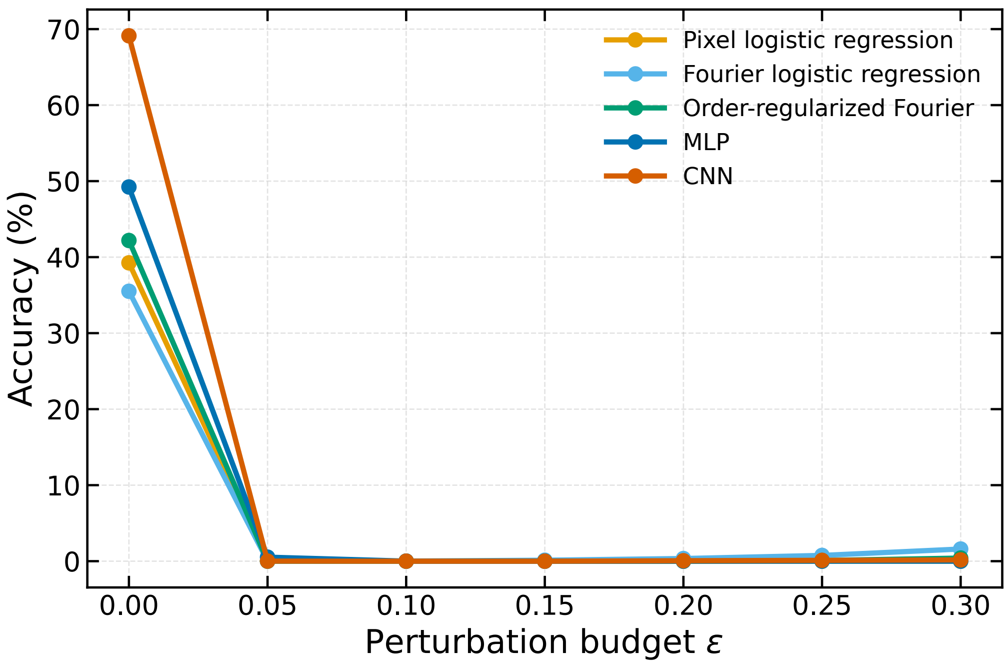
Figures 9a and 9b show the FGSM robustness curves. On MNIST, the CNN is clearly the most robust model under FGSM. At $\epsilon = 0.05$, it still achieves 93.54% accuracy, while the MLP drops to 47.05%, pixel logistic regression drops to 22.08%, and the Fourier models drop much further. At larger epsilon values, all non-CNN models are close to zero accuracy. The order-regularized Fourier model performs slightly better than unrestricted Fourier logistic regression for some epsilon values, but the difference is not enough to claim a strong robustness benefit.

On CIFAR-10, all models are highly vulnerable under FGSM. Even at $\epsilon = 0.05$, accuracy is almost zero for all models. This is partly because CIFAR-10 images are more complex and because no adversarial training or input normalisation strategy was used. The CNN has the highest clean accuracy but also collapses under the attack. Therefore, the CIFAR-10 FGSM results do not show a meaningful robustness advantage for the order-regularized Fourier model.



(a) MNIST.

Figure 9: FGSM robustness curves.



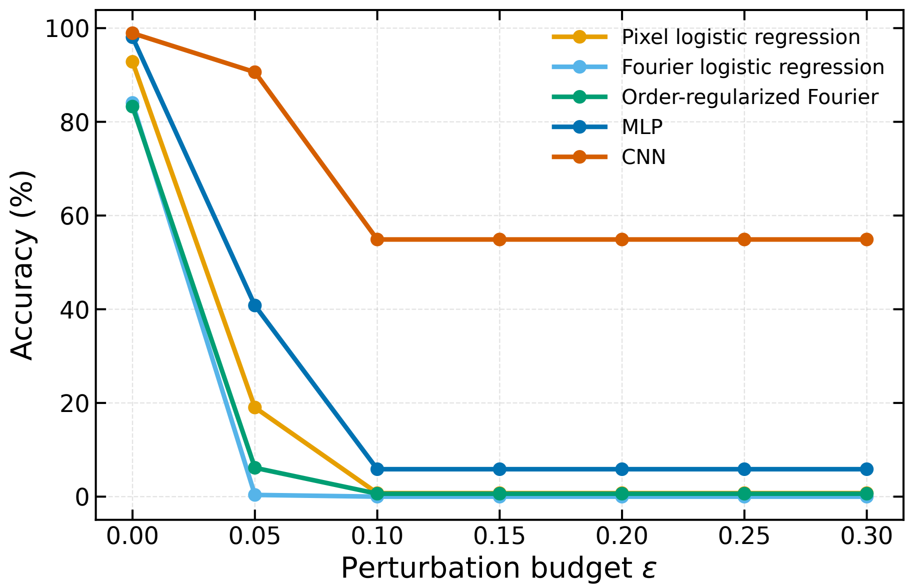
(b) CIFAR-10.

Figure 9: FGSM robustness curves.

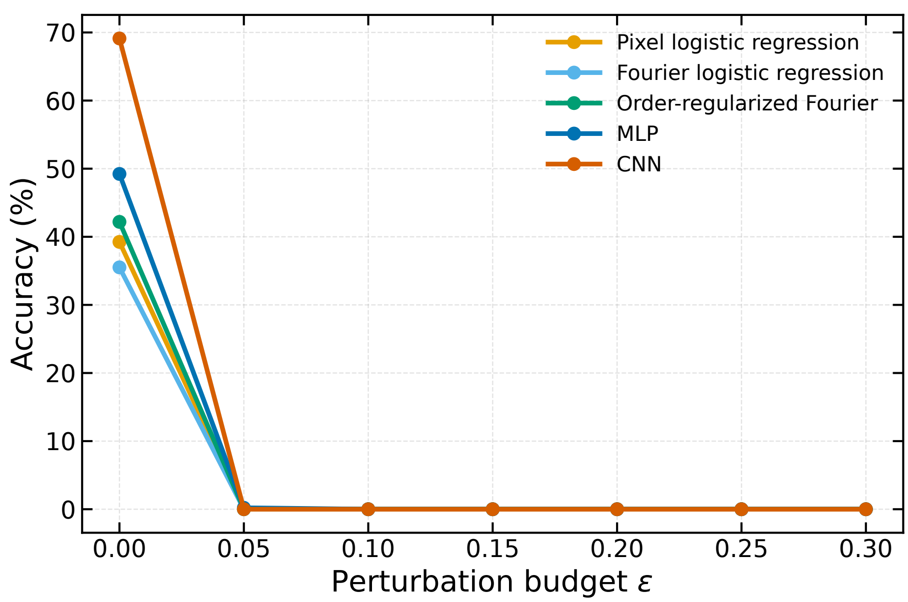
6.5 PGD robustness

Figure 10 shows the PGD robustness curves. On MNIST, the CNN again remains the strongest model. At $\epsilon = 0.05$, the CNN reaches 90.60%, while the MLP reaches 40.80% and the order-regularized Fourier model reaches 6.16%. The Fourier models are therefore not robust under the PGD setting.

On CIFAR-10, all models collapse almost completely under PGD at $\epsilon = 0.05$. This suggests that none of the trained models is robust to this attack in the current experimental setup. Since the PGD implementation uses a fixed number of steps and fixed step size, these results should be interpreted as evidence of strong vulnerability under the implemented attack, not as a complete robustness benchmark.



(a) MNIST.



(b) CIFAR-10.

Figure 10: PGD robustness curves.

The third research sub-question can therefore be answered negatively. Explicit spectral ordering does not clearly improve adversarial robustness. The order-regularized Fourier model is more robust than unrestricted Fourier logistic regression in a few MNIST FGSM points, but it is far less robust than the CNN and does not show a consistent advantage across attacks and datasets.

7 Discussion

7.1 Interpreting the contribution of the results

The results of this thesis are most useful when interpreted as an analysis of frequency-controlled learning rather than as a search for a new best-performing classifier. The order-regularized Fourier model does not outperform the neural network baselines, and the experiments do not provide evidence that explicit spectral ordering alone is sufficient for adversarial robustness. However, this negative result is still informative. It shows that controlling the availability of Fourier frequencies changes the learned spectral profile, but that this change is not enough to overcome the limitations of a linear classifier in Fourier magnitude space.

This distinction is important for the contribution of the thesis. If the proposed model had only been evaluated by final accuracy, the conclusion would simply be that the CNN performs best. By also analysing training dynamics, frequency-band weights, and robustness curves, the experiment gives a more detailed picture of what changes when frequency order is controlled. In particular, the CIFAR-10 results show that order regularization can improve over unrestricted Fourier logistic regression, while the robustness results show that this improvement does not automatically translate into adversarial robustness.

The main contribution is therefore methodological and analytical. The thesis implements a transparent framework for comparing pixel-space, Fourier-space, order-regularized Fourier, and neural-network models under the same experimental conditions. This framework makes it possible to connect classification performance with the spectral behaviour of the learned model. The findings are limited, but they help clarify where explicit frequency ordering is useful and where it is not sufficient.

7.2 Comparison with neural network baselines

The main result is that explicit frequency ordering changes the behaviour of the Fourier logistic model, but does not make it competitive with neural network baselines. This outcome is not surprising. The proposed model is still a linear classifier in Fourier magnitude space. It does not learn hierarchical spatial features, and it discards Fourier phase information. A CNN, by contrast, can learn local features and combine them spatially. This gives the CNN a strong advantage, especially on CIFAR-10.

7.3 Dataset-specific behaviour

For MNIST, the order-regularized Fourier model performs slightly worse than unrestricted Fourier logistic regression. One explanation is that MNIST is already simple enough that restricting the frequency content during early training does not help. Pixel logistic regression performs well because digit shapes are relatively aligned and simple. The Fourier magnitude representation may remove spatial phase information that is useful for distinguishing digits, which can explain why the Fourier logistic models underperform the pixel logistic baseline.

For CIFAR-10, the order-regularized Fourier model performs better than both unrestricted Fourier logistic regression and pixel logistic regression. This is one of the more interesting findings. It suggests that on a more complex dataset, directly exposing all Fourier frequencies from the start may lead the linear classifier to rely too much on less stable high-frequency patterns. The frequency-band analysis supports this interpretation: unrestricted Fourier logistic regression has its largest weight magnitudes in the high-frequency bands, while the order-regularized model has a more controlled spectral profile.

7.4 Robustness and frequency control

However, improved spectral profile is not the same as adversarial robustness. The robustness experiments show that the Fourier-based models remain highly vulnerable. The CNN is much stronger on MNIST, and all models are weak on CIFAR-10 under the implemented adversarial attacks. This suggests that simply limiting the order in which frequencies are introduced is not sufficient to produce robust classifiers. Robustness likely requires additional methods such as adversarial training, stronger regularisation, data augmentation, or architectures specifically designed for robust feature learning [11, 17].

The results also show that frequency information should be interpreted carefully. It is tempting to equate high frequencies with noise and low frequencies with useful signal, but this is too simple. High-frequency components can contain important edges and textures, especially in natural images. At the same time, high-frequency features can also be brittle or non-robust [5, 19]. The present results support a more balanced interpretation: controlling frequency access can change what the model learns, but whether this helps depends on the dataset, representation, and model class.

7.5 Limitations

Several limitations must be mentioned. First, each experiment was run only once with a fixed seed. This is acceptable for an initial bachelor thesis experiment, but it means that no statistical claims can be made about variance across random initialisations. Second, the models were trained for only 20 epochs and no extensive hyperparameter search was performed. Third, the Fourier representation uses magnitude only and discards phase. This makes the representation simpler but may lose important spatial information. Fourth, the PGD attack uses a fixed step size and number of iterations rather than a fully tuned attack for every epsilon value. Fifth, no data augmentation, normalisation beyond tensor scaling, or adversarial training was used.

A further limitation is that the thesis does not separately reproduce a classical spectral-bias demonstration on a synthetic function with known low- and high-frequency components. Such an experiment would make the frequency principle visible in a more controlled setting before moving to image classification. Instead, this thesis focuses on applying the frequency-ordering idea directly to MNIST and CIFAR-10. Similarly, the adversarial examples are introduced conceptually and evaluated quantitatively through FGSM and PGD robustness curves, but the thesis does not include visual examples of generated adversarial images from the test set. Adding such visualisations would make the attack mechanism more concrete for the reader.

Despite these limitations, the experiment answers the research question clearly. The order-regularized Fourier feature model is useful as an interpretable analysis model, but not as a high-performance or robust replacement for standard neural networks. The most defensible conclusion is that explicit spectral ordering affects learned frequency weights and can improve over unrestricted Fourier logistic regression on CIFAR-10, but it does not close the gap to neural networks and does not solve adversarial vulnerability.

8 Conclusion and Further Research

This thesis investigated how an order-regularized Fourier feature model compares to standard neural network baselines in classification performance, training dynamics, and adversarial robustness. The model transforms images into the Fourier domain, applies a radial low-frequency mask, and increases the mask radius over training epochs. It was evaluated on MNIST and CIFAR-10 against pixel logistic regression, unrestricted Fourier logistic regression, an MLP, and a CNN.

The answer to the main research question is that the order-regularized Fourier model does not outperform standard neural networks. The CNN achieves the highest test accuracy on both MNIST and CIFAR-10 and is also the most robust model on MNIST. The MLP is also stronger than the Fourier models, especially on MNIST. The order-regularized Fourier model achieves 83.26% accuracy on MNIST and 42.19% accuracy on CIFAR-10. Its CIFAR-10 result is better than unrestricted Fourier logistic regression and pixel logistic regression, but still below the MLP and CNN.

The training dynamics and frequency-band analysis show that the order-regularized model changes the learned spectral profile. In particular, it reduces reliance on higher frequency bands compared with unrestricted Fourier logistic regression. This supports the idea that explicit spectral ordering can be used as an analysis tool. However, the adversarial robustness experiments do not show a clear robustness advantage. Under FGSM and PGD, the Fourier-based models remain vulnerable, and the CNN is clearly stronger on MNIST.

Future work could improve the experiment in several ways. First, the results should be repeated across multiple random seeds to measure variance. Second, the Fourier representation could include both magnitude and phase instead of magnitude only. Third, the radius schedule could be tuned or learned rather than fixed linearly. Fourth, order-regularized Fourier features could be combined with neural networks instead of a linear classifier. Another thing is that adversarial training could be added to test whether spectral ordering and robust optimization complement each other. Finally, Future work should also evaluate the Fourier feature models against adversarial attacks applied directly in pixel space. In such an evaluation, the perturbed pixel image would first be transformed into Fourier features before classification. This would make it possible to determine whether Fourier-based representations provide any robustness advantage against realistic input-level perturbations and would allow a more direct comparison with the pixel-based neural-network models.

These extensions would make it possible to better understand when frequency control helps and when it is too limited by the model class.

References

[1] Ronen Basri, Meirav Galun, Amnon Geifman, David Jacobs, Yoni Kasten, and Shira Kritchman. Frequency bias in neural networks for input of non-uniform density. In *Proceedings of the 37th International Conference on Machine Learning*, pages 685–694. PMLR, 2020.

[2] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[5] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[6] Jason Jo and Yoshua Bengio. Measuring the tendency of CNNs to learn surface statistical regularities. *arXiv preprint arXiv:1711.11561*, 2017.

[7] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[8] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

[9] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[10] Tao Luo, Zheng Ma, Zhiwei Wang, Zhi-Qin John Xu, and Yaoyu Zhang. Fourier-domain variational formulation and its well-posedness for supervised learning. *arXiv preprint arXiv:2012.03238*, 2020.

[11] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[12] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[14] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred A. Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019.

[15] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

[16] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, volume 33, pages 7537–7547, 2020.

[17] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2019.

[18] Zhi-Qin John Xu, Yaoyu Zhang, and Yanyang Xiao. Training behavior of deep neural network in frequency domain. In *Neural Information Processing*, pages 264–274. Springer, 2019.

[19] Dong Yin, Raphael Gontijo Lopes, Jonathon Shlens, Ekin D. Cubuk, and Justin Gilmer. A Fourier perspective on model robustness in computer vision. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[20] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations*, 2017.