Appendix for "Residual Alignment: Uncovering the Mechanisms of Residual Networks"

Anonymous Author(s)
Affiliation
Address
`email`

1 (RA2+3+4) Imply (RA1)

Theorem 3.1. *In a pre-activation ResNet, assuming the Jacobian linearizations are exact and satisfy (RA2+3+4), then (RA1) holds for the intermediate representations.*

Proof. In a pre-activation ResNet,

$$h_{i+1} = h_i + \mathcal{F}(h_i; \mathcal{W}_i).$$

Since Jacobian linearizations are exact, we have:

$$h_{i+1} = (I + J_i)h_i.$$

Recall, the singular value decomposition of J_i is given by

$$J_i = U_i S_i V_i^\top,$$

where U_i and V_i are the respective left and right singular vectors, and S_i is the singular value matrix. Invoking (RA2) ,

$$J_i = U S_i U^\top,$$

and therefore

$$h_{i+1} = (I + U S_i U^\top)h_i.$$

Applying recursively the above equality leads to

$$h_k = \left(\prod_{i=1}^{k-1}(I + U S_i U^\top)\right) h_1 = U \left(\prod_{i=1}^{k-1}(I + S_i)\right) U^\top h_1. \tag{1}$$

For binary classification, (RA3) implies the Jacobians are rank 1 and therefore

$$h_k = U_{k,1} \left(\prod_{i=1}^{k-1}(1 + S_{i,1})\right) U_{k,1}^\top h_1.$$

According to (RA4) ,

$$S_{i,1} = \frac{1}{i}$$

and

$$\prod_{i=1}^{k-1}(1 + S_{i,1}) = \prod_{i=1}^{k-1}(1 + 1/i) = k.$$

Substituting the above into Equation (1), we obtain

$$h_k = k U_{i,1}\, U_{i,1}^\top h_1.$$

This proves that the intermediate representations of a given input are *equispaced* on a *line* embedded in high dimensional space, i.e., (RA1) .

□

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

2 Unconstrained Jacobians Model Leads to RA

We start by providing motivation for the unconstrained Jacobians problem introduced in the main text. Assume a training example, x, is situated next to a point on the classification boundary, denoted by x_{mid}, satisfying $f(x_{\text{mid}}; \mathcal{W}) = 0$. Performing a Taylor expansion of the logits of x around x_{mid} yields

$$\begin{aligned} f(x; \mathcal{W}) &= f(x_{\text{mid}}; \mathcal{W}) + \left.\frac{\partial f(x; \mathcal{W})}{\partial x}\right|_{x_{\text{mid}}} \Delta_x + h(x, x_{\text{mid}}) \\ &= \left.\frac{\partial f(x; \mathcal{W})}{\partial x}\right|_{x_{\text{mid}}} \Delta_x + h(x, x_{\text{mid}}), \end{aligned} \tag{2}$$

where $\Delta_x = x - x_{\text{mid}}$ and $h(x, x_{\text{mid}})$ accounts for the approximation error, which is $\mathcal{O}(\|\Delta_x\|_2^2)$ and assumed to be negligible in our analysis.

Recall the loss associated with training a ResNet:

$$\underset{\{\mathcal{W}_i\}_{i=1}^{L+1}}{\text{minimize}} \ \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(f(x_n; \mathcal{W}), y_n) + \frac{\lambda}{2} \|\mathcal{W}\|_2^2. \tag{3}$$

Substituting Equation (2) into the above, neglecting the approximation error, and considering only the objective associated with the training sample x and its label y, we get

$$\mathcal{L}\left(\left.\frac{\partial f(x; \mathcal{W})}{\partial x}\right|_{x_{\text{mid}}} \Delta_x, y \right) + \frac{\lambda}{2} \|\mathcal{W}\|_2^2.$$

By using the chain rule, we can then obtain the following:[1]

$$\mathcal{L}\left(\frac{\partial \mathcal{C}(h_{L+1}; \mathcal{W}_{L+1})}{\partial h_{L+1}} \prod_{i=1}^{L} \left(I + \frac{\partial \mathcal{F}(h_i; \mathcal{W}_i)}{\partial h_i} \right) \Delta_x, y \right) + \frac{\lambda}{2} \sum_{i=1}^{L} \|\mathcal{W}_i\|_F^2 + \frac{\lambda}{2} \|\mathcal{W}_{L+1}\|_F^2, \tag{4}$$

where all the Jacobians are evaluated at the point x_{mid}. This naturally gives rise to the following definition, as introduced in the main text.

Definition 3.2 (Unconstrained Jacobians Model). *Given a fixed input $\Delta_x \in \mathbb{R}^D$ and its label $y \in \{+1, -1\}$, find matrices $J_i \in \mathbb{R}^{D \times D}$, $1 \le i \le L$, and vector $w \in \mathbb{R}^D$ that*

$$\underset{w, \{J_i\}_{i=1}^{L}}{\text{minimize}} \quad \mathcal{L}(w^\top \prod_{i=1}^{L} (I + J_i) \Delta_x, y) + \frac{\lambda}{2} \sum_{i=1}^{L} \|J_i\|_F^2 + \frac{\lambda}{2} \|w\|_2^2.$$

In the main text, we stated the following theorem.

Theorem 3.3. *There exists a global optimum of the Unconstrained Jacobians Model where the top Jacobian singular vectors are aligned (`RA2`), all Jacobians are rank one, analogous to (`RA3`), and the top Jacobian singular values are equal, analogous to (`RA4`).*

Proof. Throughout the proof, we assume, without loss of generality, that the label is $y = 1$. Using the cyclic property of the trace, the logit, $w^\top \prod_{i=1}^{L} (I + J_i) \Delta_x$, equals

$$\text{tr}\left\{ \Delta_x w^\top \prod_{i=1}^{L} (I + J_i) \right\}.$$

Denoting the power set of all natural numbers between 1 and L by $\mathcal{P}(L)$, the above can be expressed as follows:

$$\sum_{s \in \mathcal{P}(L)} \text{tr}\left\{ \Delta_x w^\top \prod_{i \in s} J_i \right\}.$$

Each element in the above summation can be upper bounded through the following theorem (a generalization of Von Neumann's trace inequality [Mirsky, 1975] to the product of more than two real matrices).

[1] For simplicity, we ignore the input transformation that maps the input Δ_x to the first representation $h_1 \in \mathbb{R}^D$ by simply assuming $\Delta_x \in \mathbb{R}^D$.

Theorem 2 ([Miranda and Thompson, 1993]). *Let $A_1, \dots, A_m$ be matrices with real entries. Take the singular values of A_j to be $s_1(A_j) \geq \cdots \geq s_n(A_j)$, for $j = 1, \dots, m$, and denote $S_j = \operatorname{diag}(s_1(A_j), \dots, s_n(A_j))$. Then, as the matrices $P_1, \dots, P_m$ range over all possible rotations, i.e., the special orthogonal group $\mathrm{SO}(n)$,*

$$\begin{aligned}&\sup_{P_1,\dots,P_m \in \mathrm{SO}(n)} \operatorname{tr}(A_1 P_1 \dots A_m P_m) \\ &= \sum_{i=1}^{n-1} \prod_{j=1}^{m} s_i(A_j) + [\operatorname{sign}\det(A_1 \dots A_m)] \prod_{j=1}^{m} s_n(A_j).\end{aligned}$$

Moreover, assuming $\operatorname{sign}\det(A_1 \dots A_m) = 1$,

$$\sup_{P_1,\dots,P_m \in \mathrm{SO}(n)} \operatorname{tr}(A_1 P_1 \dots A_m P_m) = \operatorname{tr}\left\{\prod_{i=1}^{m} S_i\right\}.$$

We will continue our proof using contradiction. Suppose all existing global optima of the unconstrained Jacobians problem consist of Jacobians that do not have aligning singular vectors, or do not have equal singular values, or are not rank 1. Then take any solution $\{J_i\}_{i=1}^{L}$ and w. Using the singular value decomposition, we have

$$J_i = U_i S_i V_i^\top, \quad \text{for } i = 1, \dots, L,$$

and

$$\Delta_x w^\top = U_{L+1} S_{L+1} V_{L+1}^\top.$$

Then Theorem 2 implies

$$\begin{aligned}\sum_{s \in \mathcal{P}(L)} \operatorname{tr}\left\{\Delta_x w^\top \prod_{i \in s} J_i\right\} &\leq \sum_{s \in \mathcal{P}(L)} \operatorname{tr}\left\{S_{L+1} \prod_{i \in s} S_i\right\} \\ &= \operatorname{tr}\left\{S_{L+1} \prod_{i=1}^{L} (I + S_i)\right\}.\end{aligned}$$

For all $s \in \mathcal{P}(L)$, the inequality becomes equality once the singular vectors of all the Jacobians align with those of $\Delta_x w^\top$ and once the vector w is chosen to be proportional to Δ_x (so that the matrix $\Delta_x w^\top$ is symmetric). The implication of the steps thus far is that one can increase, or at least keep constant the logit, and consequently reduce, or at least keep constant the loss by simply aligning the singular vectors of the Jacobians. In addition, since the regularization term $\|J_i\|_F^2 = \operatorname{tr}\{S_i^2\}$, this change of Jacobians does not affect the regularization terms.

Notice that $\Delta_x w^\top$ is a rank one matrix and so S_{L+1} has a single non-zero diagonal entry. Furthermore, the matrices S_i, for $1 \leq i \leq L$, are all diagonal. As such, we can zero out all their other diagonal entries and leave a single non-zero entry at the location that S_{L+1} has one, which does not affect the logits but reduces the regularization terms.

Using the inequality of arithmetic and geometric means on this only non-zero entry s_i of every diagonal matrix S_i gives

$$s_{L+1} \prod_{i=1}^{L} (1 + s_i) \leq s_{L+1} \left(1 + \frac{1}{L} \sum_{i=1}^{L} s_i\right)^L.$$

The implication of the above inequality is that, once the singular vectors of the Jacobians are aligned, one can further increase the logits and reduce the loss by averaging all the top singular values, s_i for $1 \leq i \leq L+1$, and forcing them to be equal. Furthermore, since $\|J_i\|_F^2 = \operatorname{tr}\{S_i^2\} = s_i^2$ is convex, by Jensen's inequality, averaging the singular values only decreases the value of the Jacobian regularization.

All in all, we obtain higher, or at least no lower logit, and lower, or at least no higher loss when all singular vectors are aligned, all top singular values are equal and all other singular values are zero, which contradicts the statement that no global optima of the unconstrained Jacobians problem satisfies all of these conditions. □

3 Additional Empirical Evidence

3.1 (RA1)

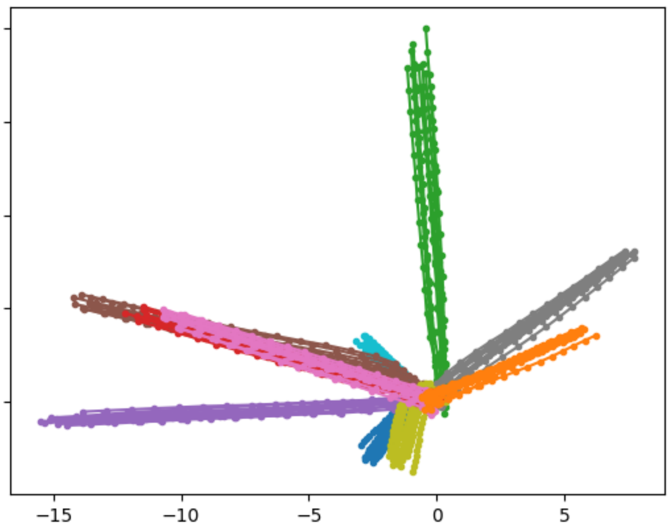


Figure 1: Fully-connected ResNet34 (Type 1 model) trained on MNIST.

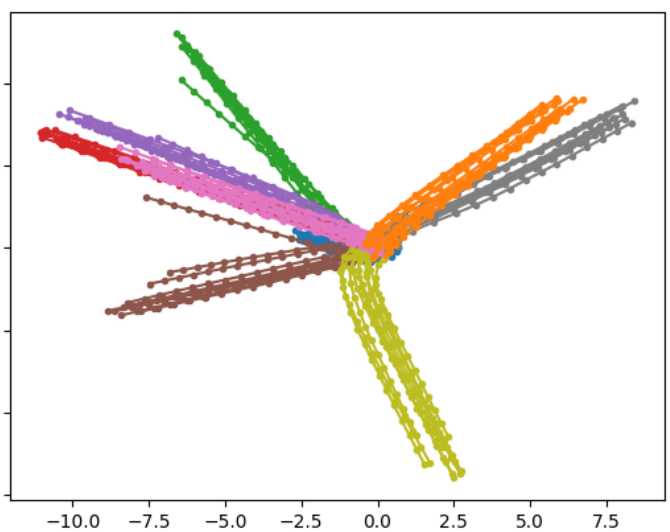


Figure 2: Fully-connected ResNet34 (Type 1 model) trained on FashionMNIST.

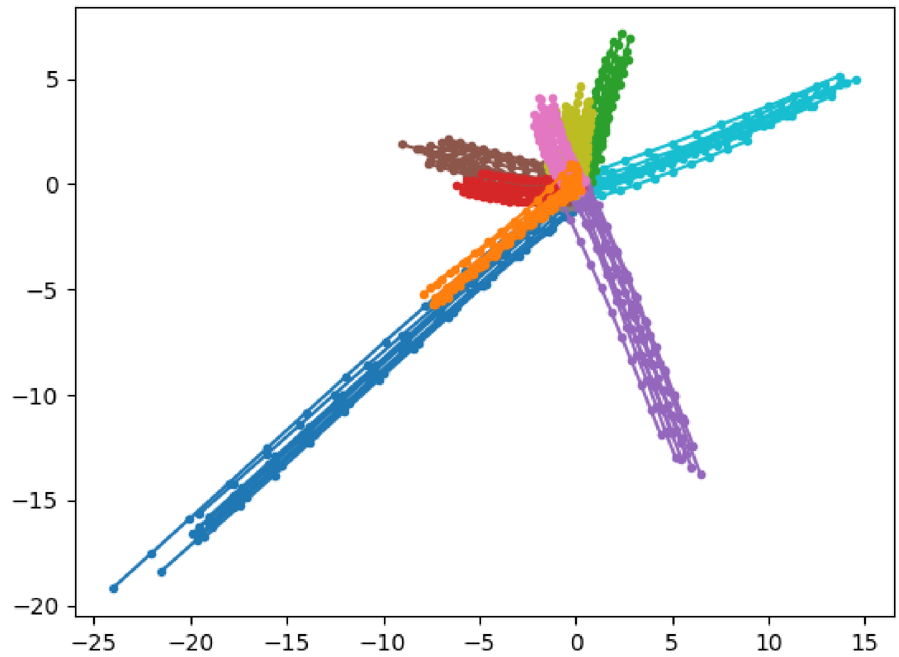


Figure 3: Fully-connected ResNet34 (Type 1 model) trained on CIFAR10.

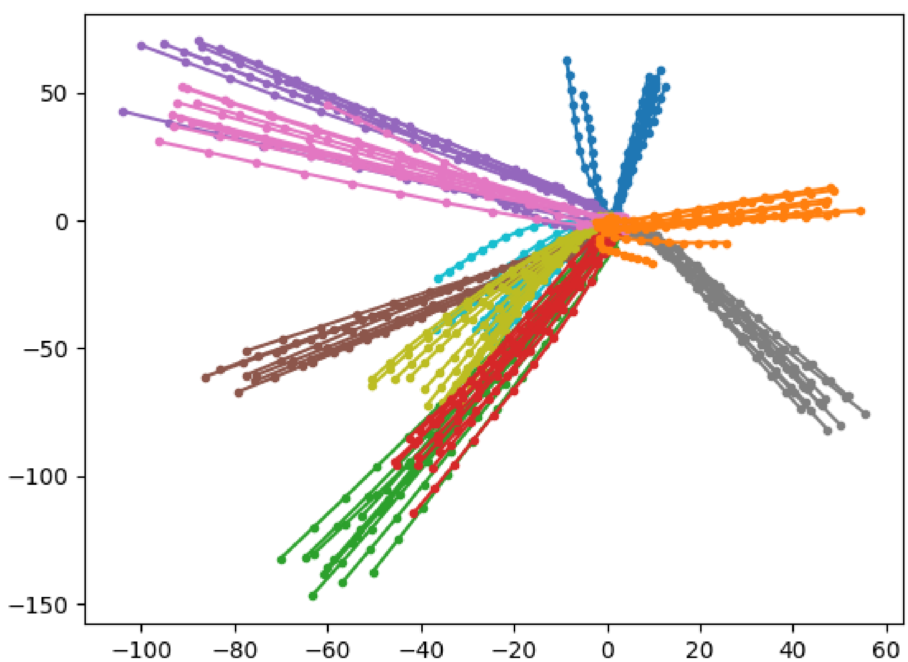


Figure 4: Convolutional ResNet34 (Type 2 model) trained on MNIST.

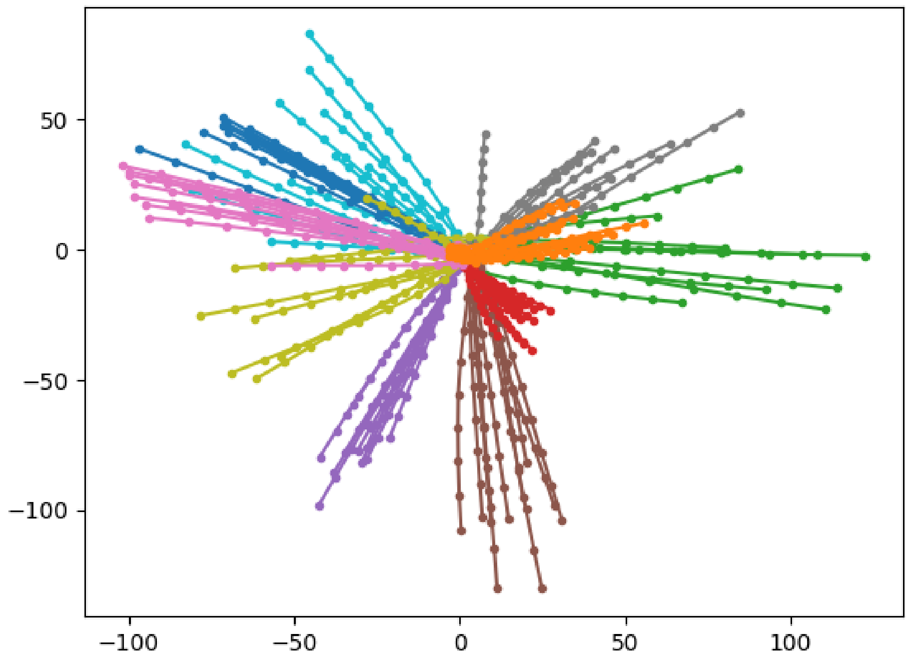


Figure 5: Convolutional ResNet34 (Type 2 model) trained on FashionMNIST.

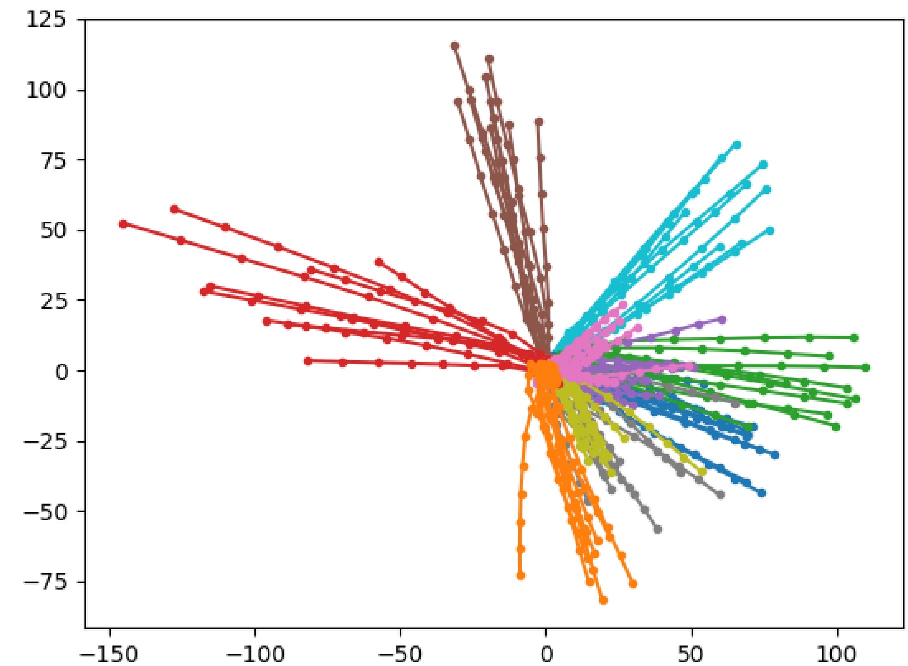


Figure 6: Convolutional ResNet34 (Type 2 model) trained on CIFAR10.

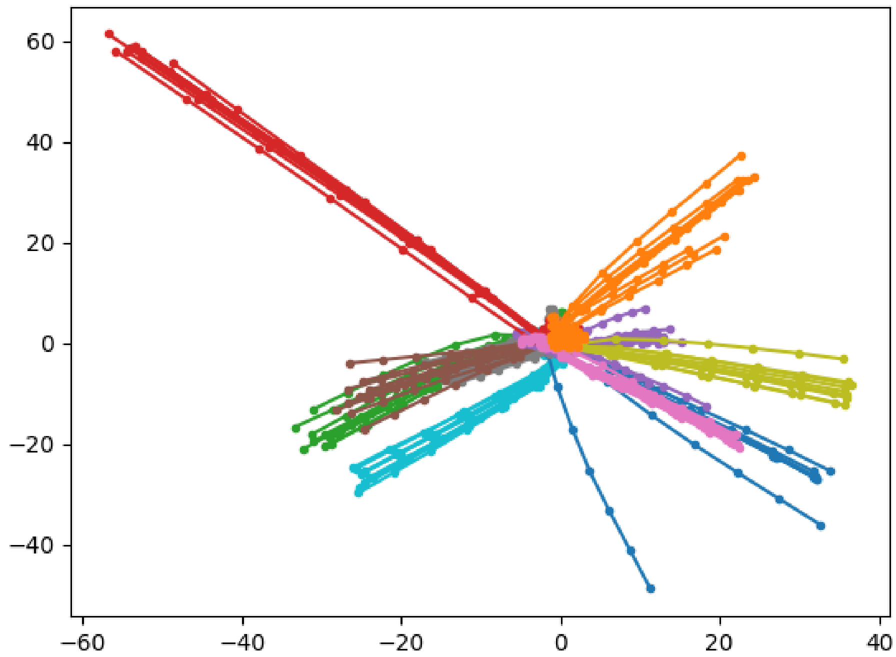


Figure 7: Convolutional ResNet34 with downsampling (Type 3 model) trained on MNIST.

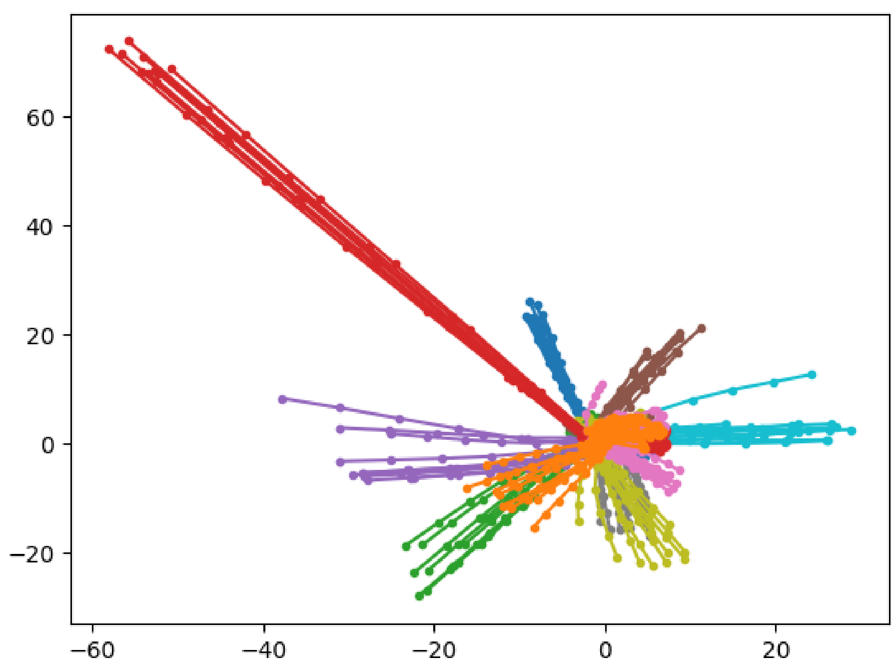


Figure 8: Convolutional ResNet34 with downsampling (Type 3 model) trained on FashionMNIST.

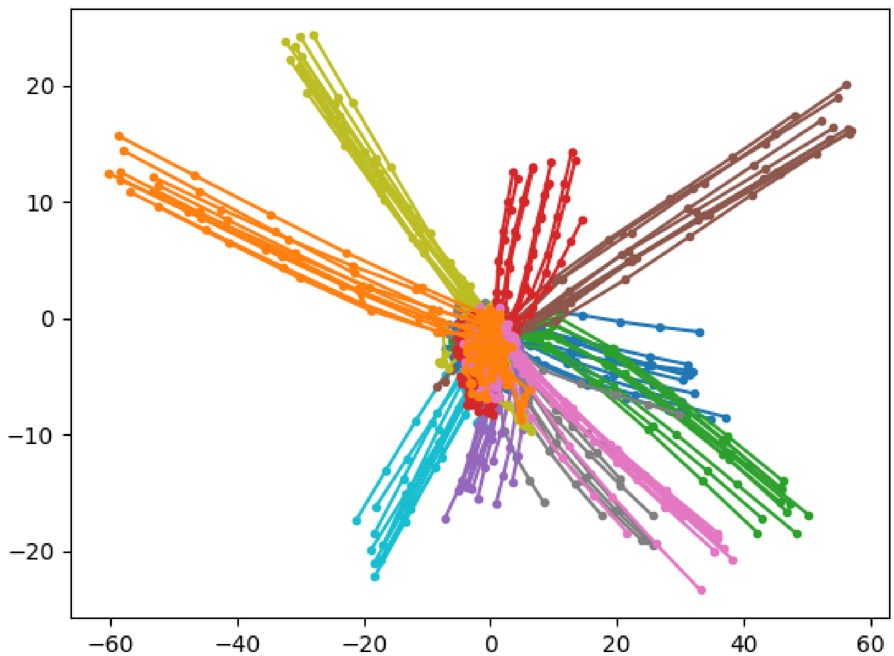


Figure 9: Convolutional ResNet34 with downsampling (Type 3 model) trained on CIFAR10.

3.2 (RA2) : $U_{j,K}^\top J_i V_{j,K}$

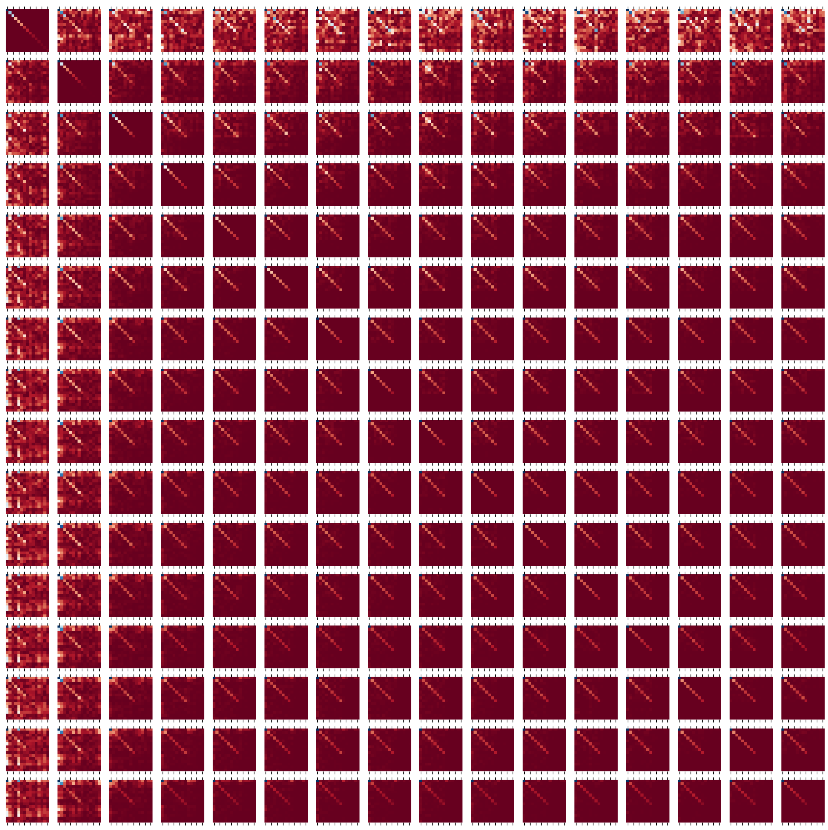


Figure 10: Fully-connected ResNet34 (Type 1 model) trained on MNIST.

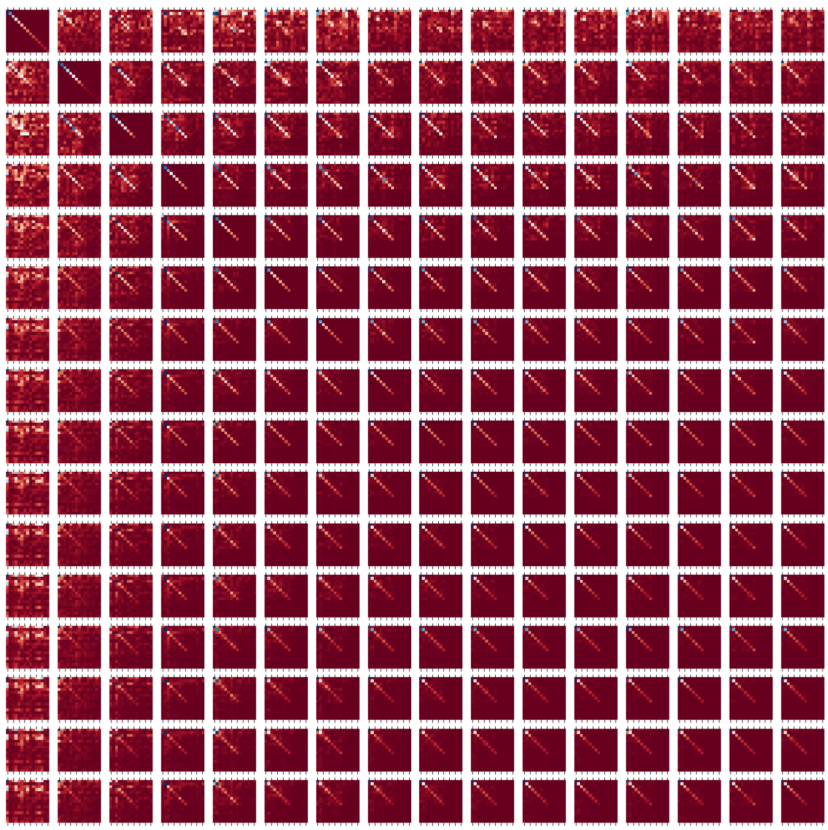


Figure 11: Fully-connected ResNet34 (Type 1 model) trained on FashionMNIST.

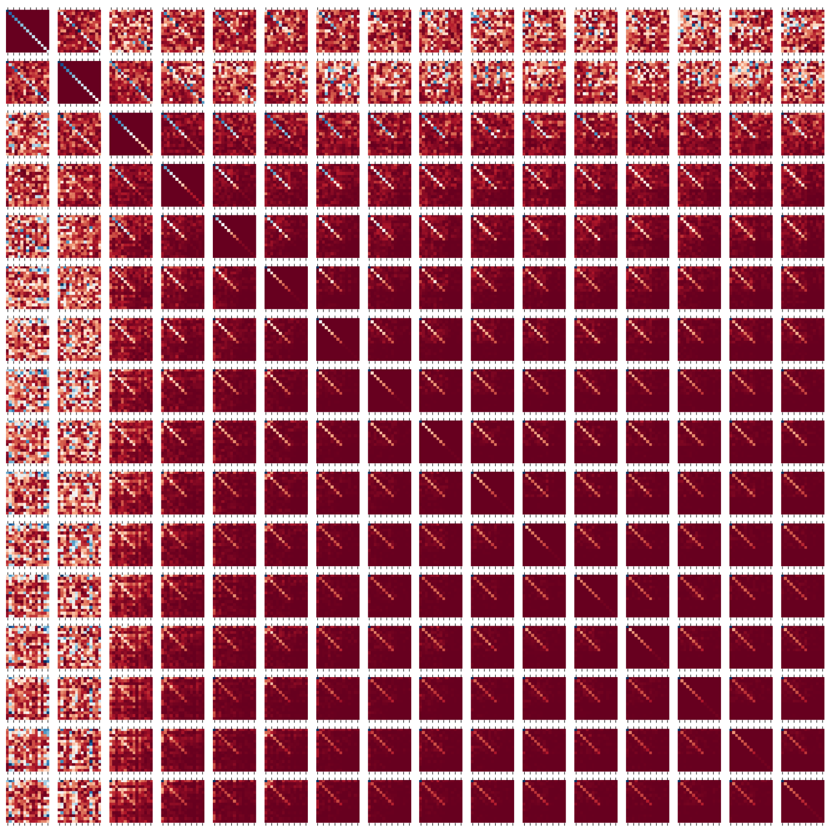


Figure 12: Fully-connected ResNet34 (Type 1 model) trained on CIFAR10.

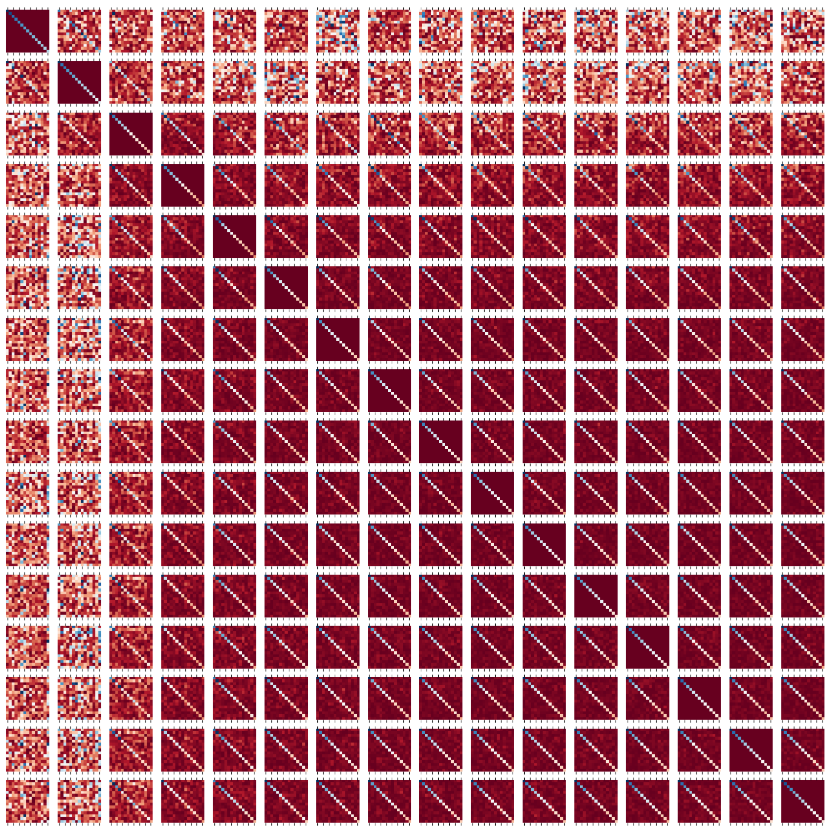


Figure 13: Fully-connected ResNet34 (Type 1 model) trained on CIFAR100.

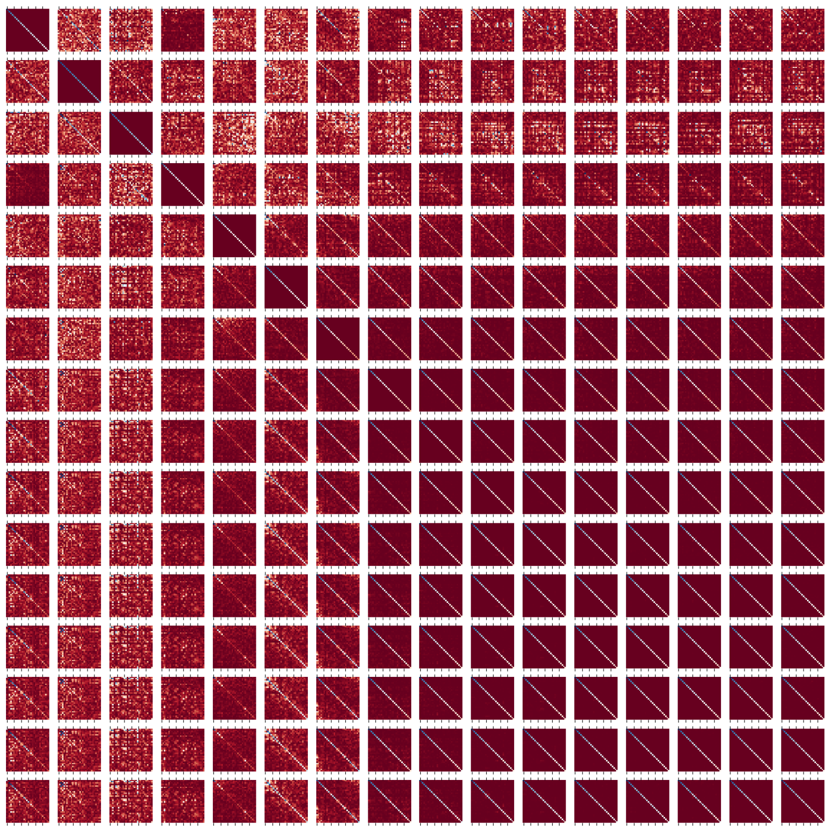


Figure 14: Convolutional ResNet34 (Type 2 model) trained on MNIST.

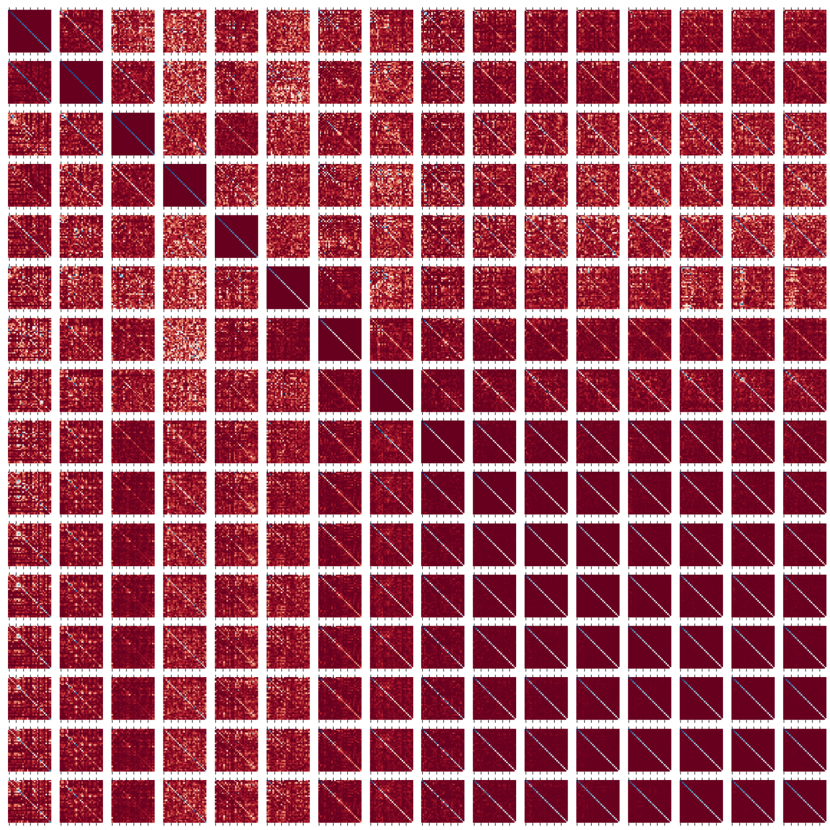


Figure 15: Convolutional ResNet34 (Type 2 model) trained on FashionMNIST.

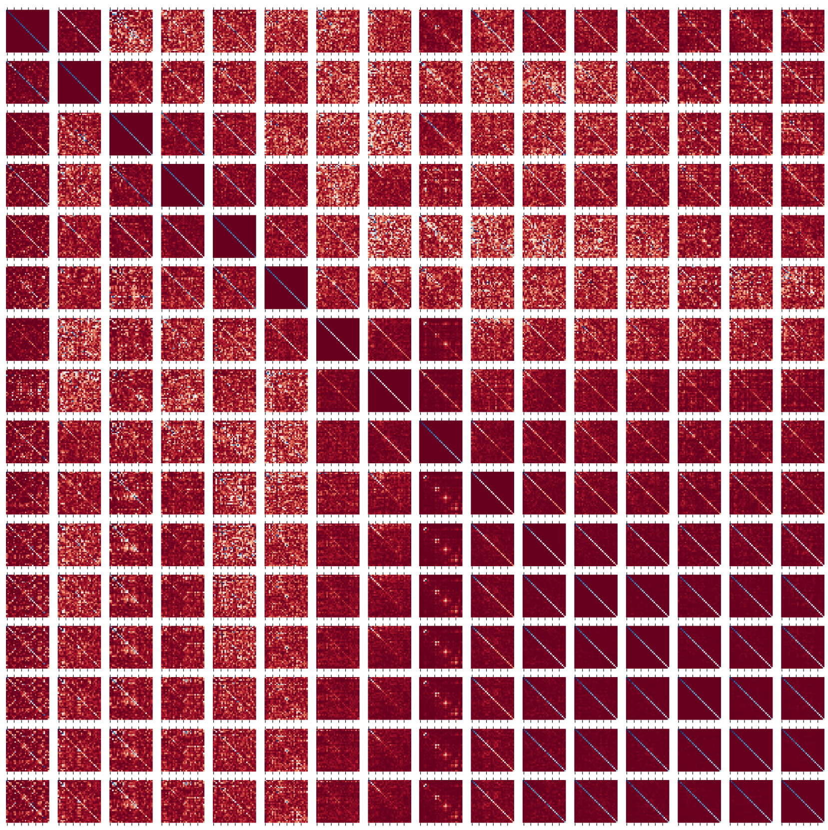


Figure 16: Convolutional ResNet34 (Type 2 model) trained on CIFAR10.

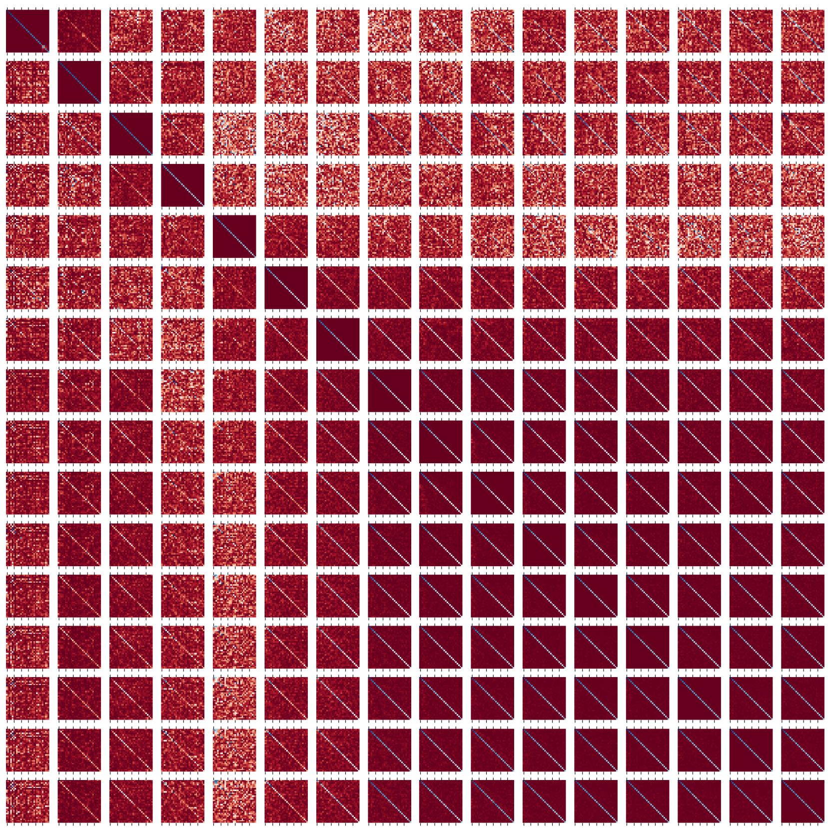


Figure 17: Convolutional ResNet34 (Type 2 model) trained on CIFAR100.

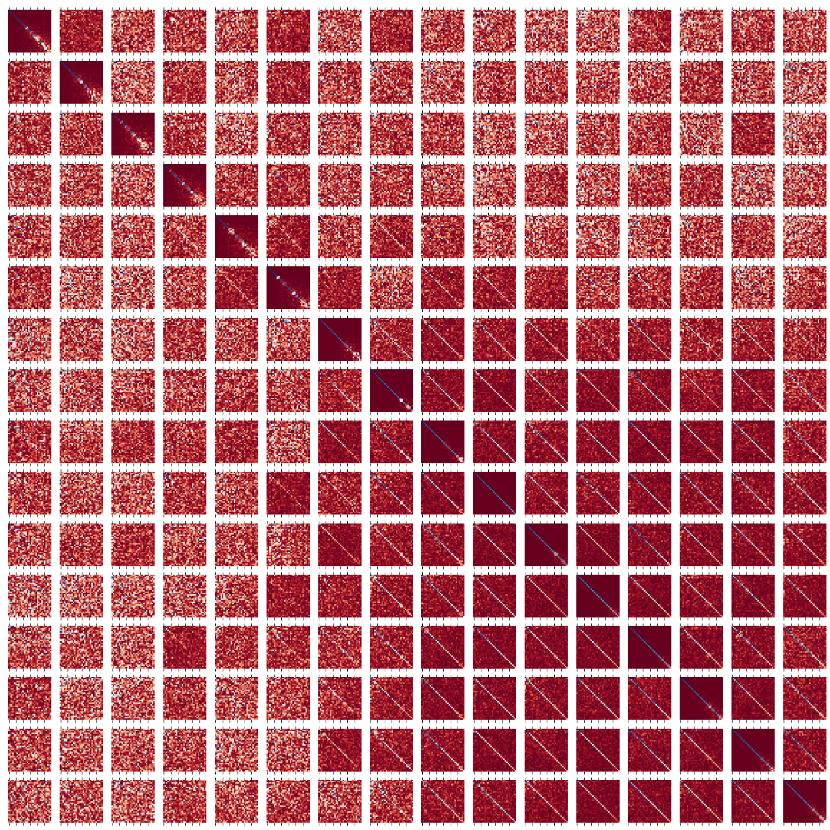


Figure 18: Convolutional ResNet34 (Type 2 model) trained on ImageNette.

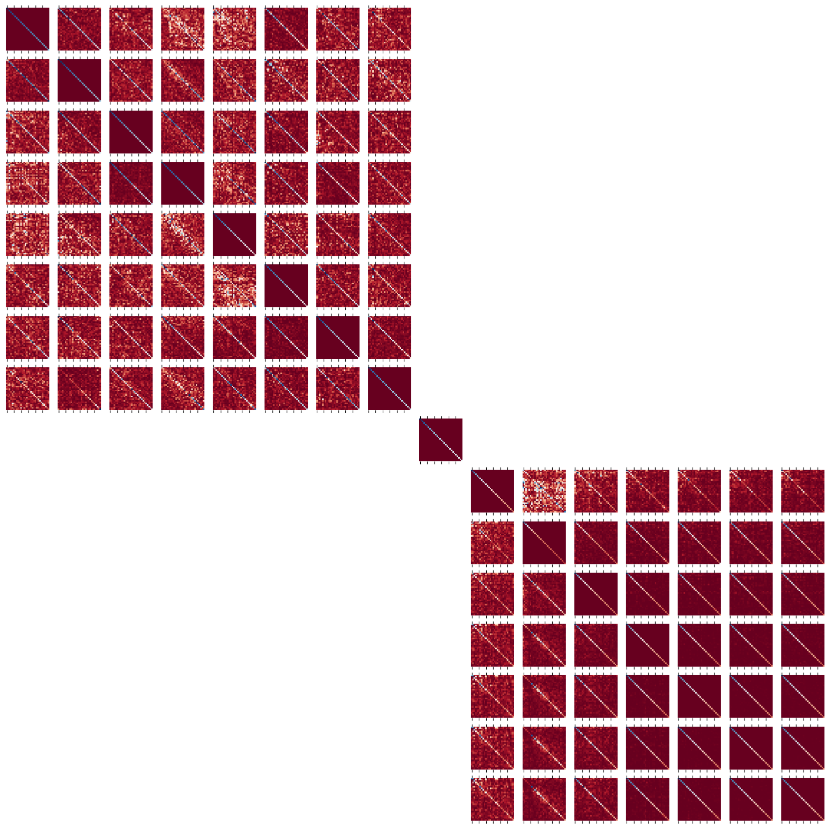


Figure 19: Convolutional ResNet34 with downsampling (Type 3 model) trained on MNIST.

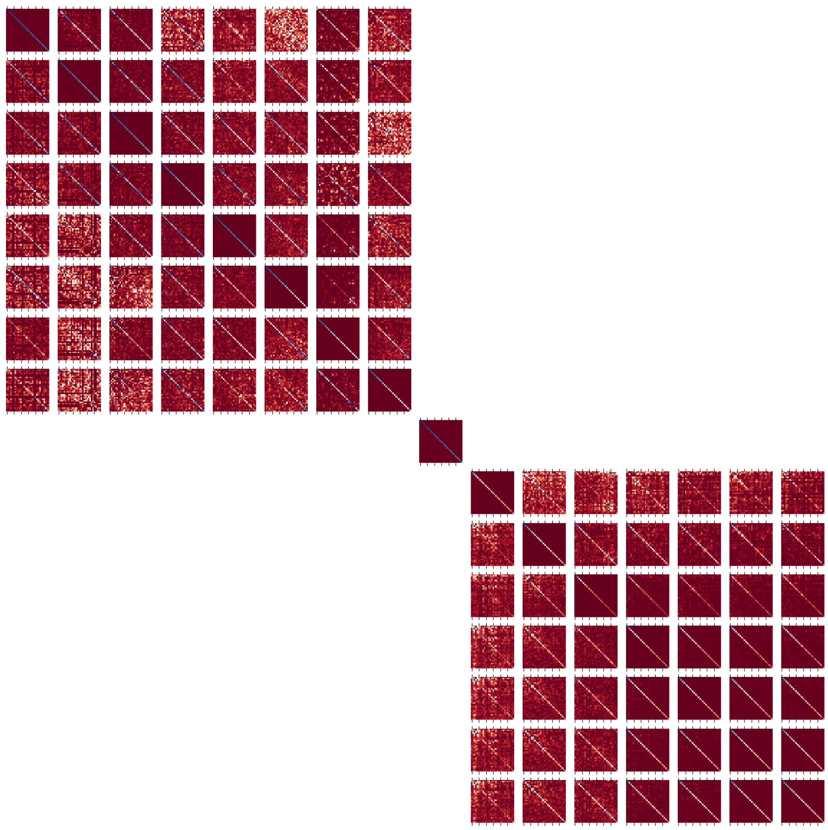


Figure 20: Convolutional ResNet34 with downsampling (Type 3 model) trained on FashionMNIST.

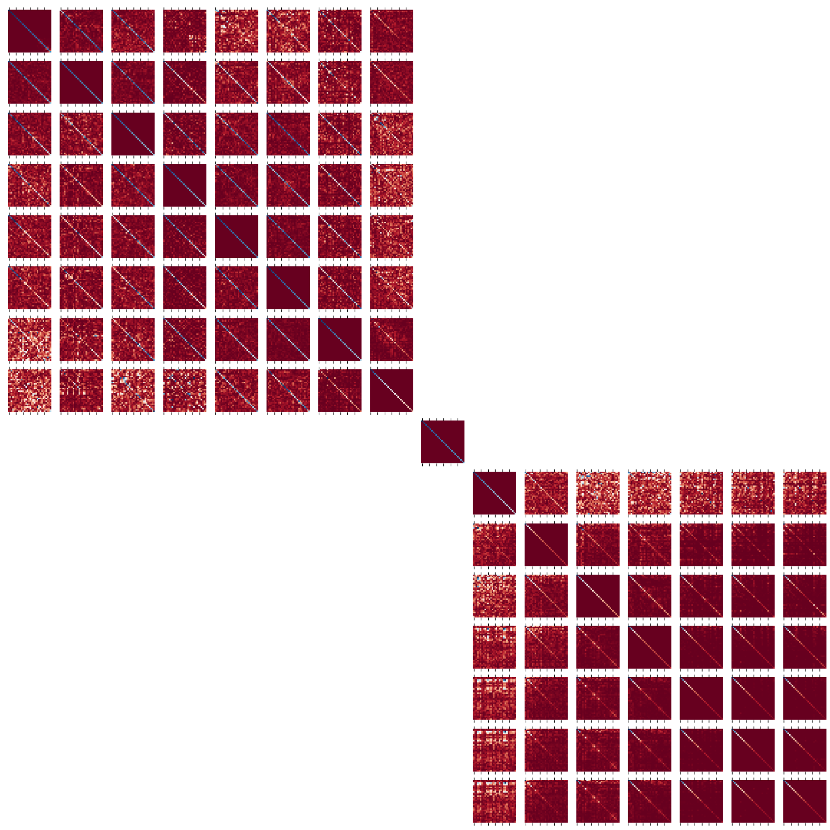


Figure 21: Convolutional ResNet34 with downsampling (Type 3 model) trained on CIFAR10.

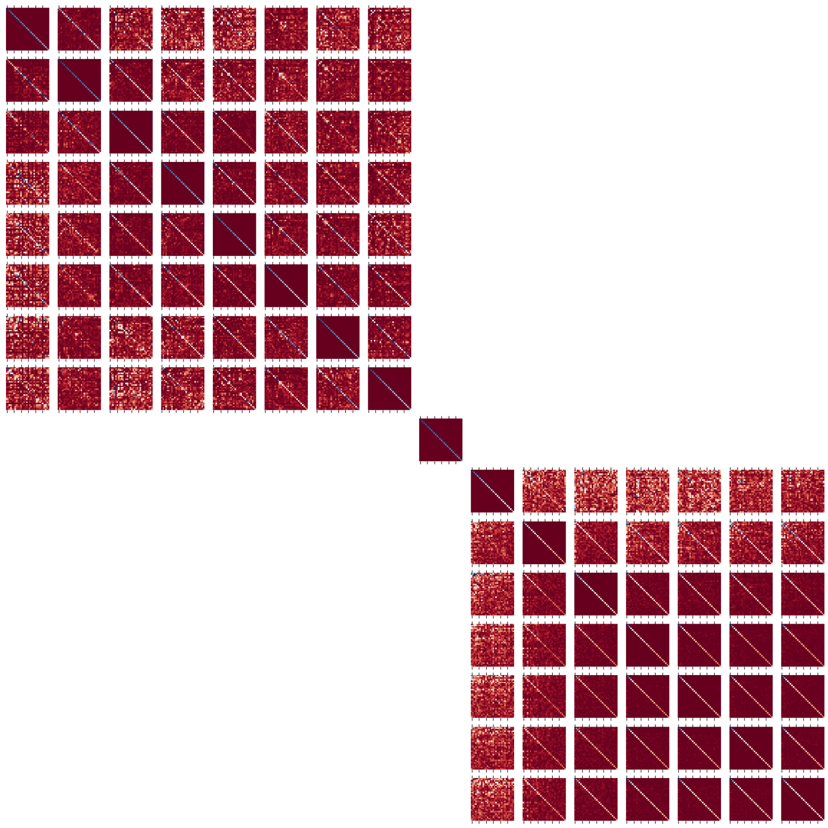


Figure 22: Convolutional ResNet34 with downsampling (Type 3 model) trained on CIFAR100.

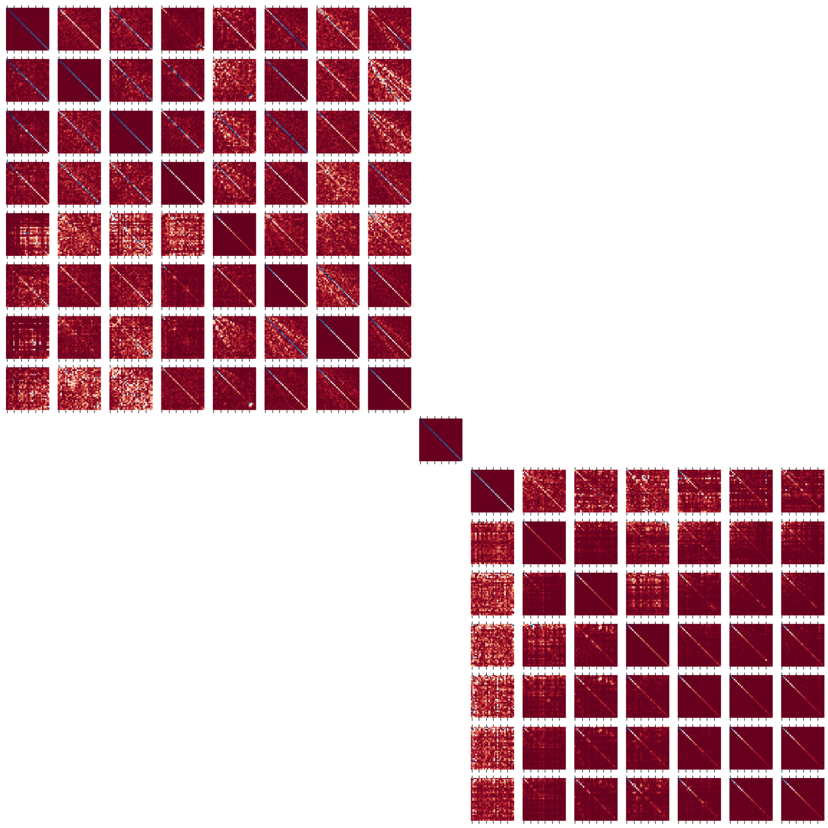


Figure 23: Convolutional ResNet34 with downsampling (Type 3 model) trained on ImageNette.

3.3 (RA2) : $V_{j,K}^{\top} J_i U_{j,K}$

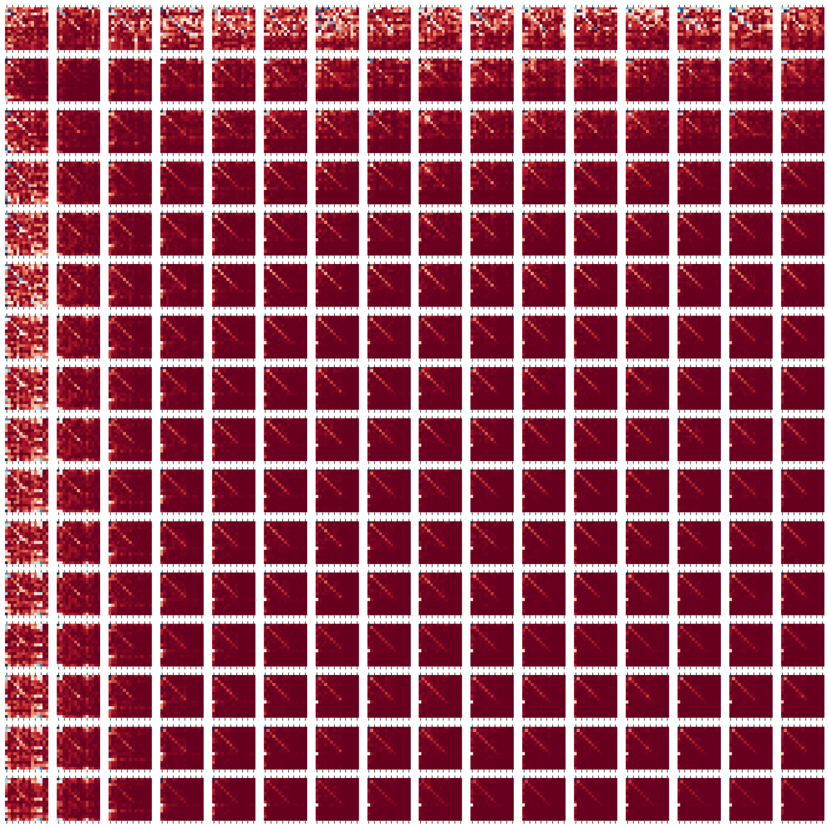


Figure 24: Fully-connected ResNet34 (Type 1 model) trained on MNIST.

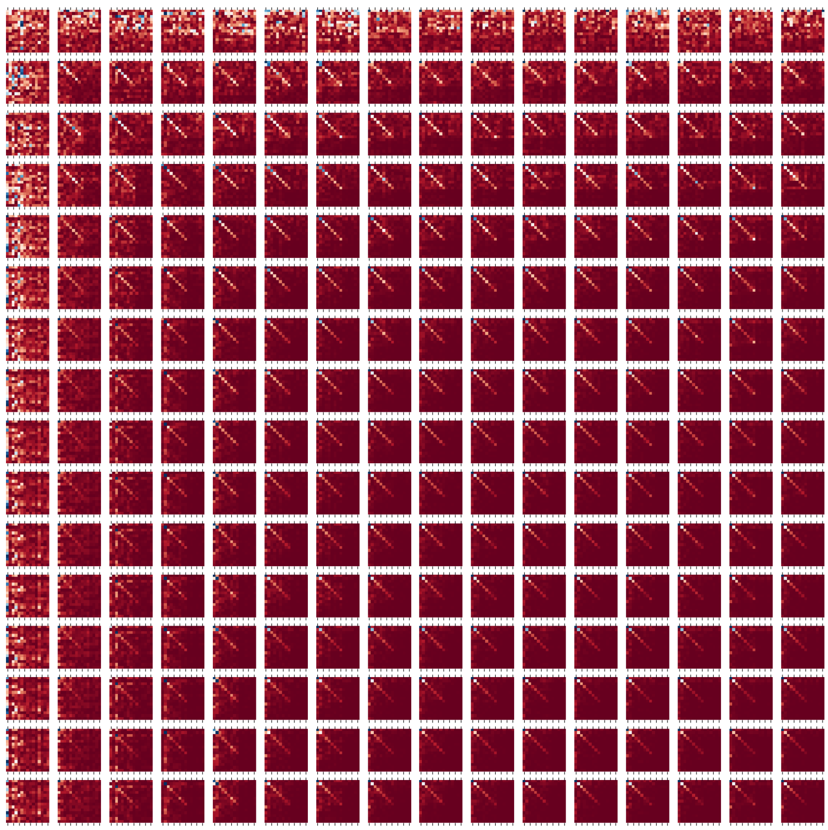


Figure 25: Fully-connected ResNet34 (Type 1 model) trained on FashionMNIST.

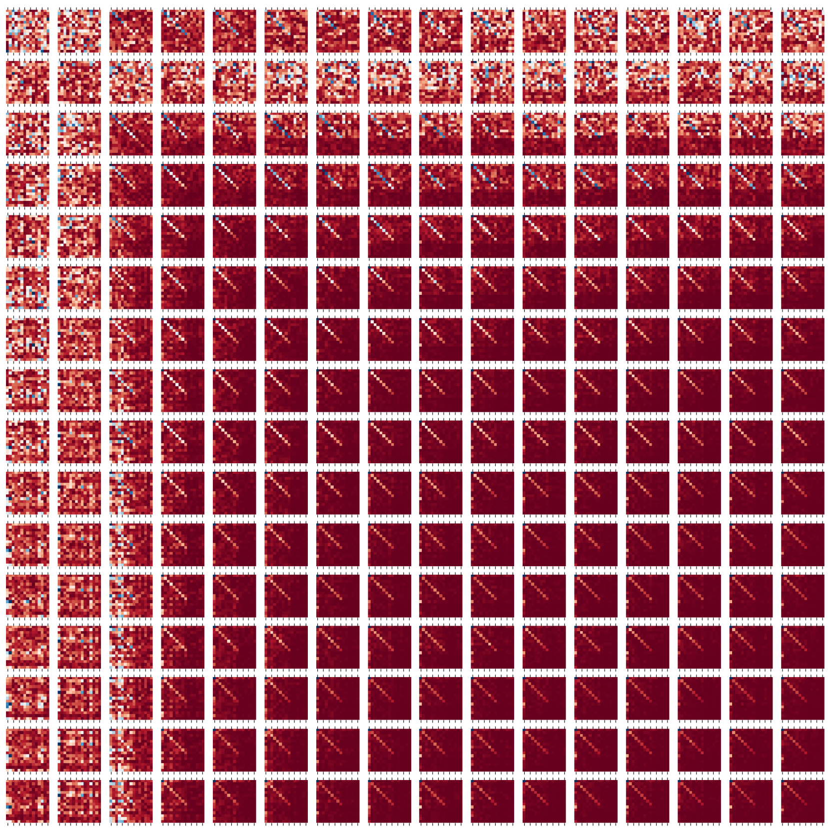


Figure 26: Fully-connected ResNet34 (Type 1 model) trained on CIFAR10.

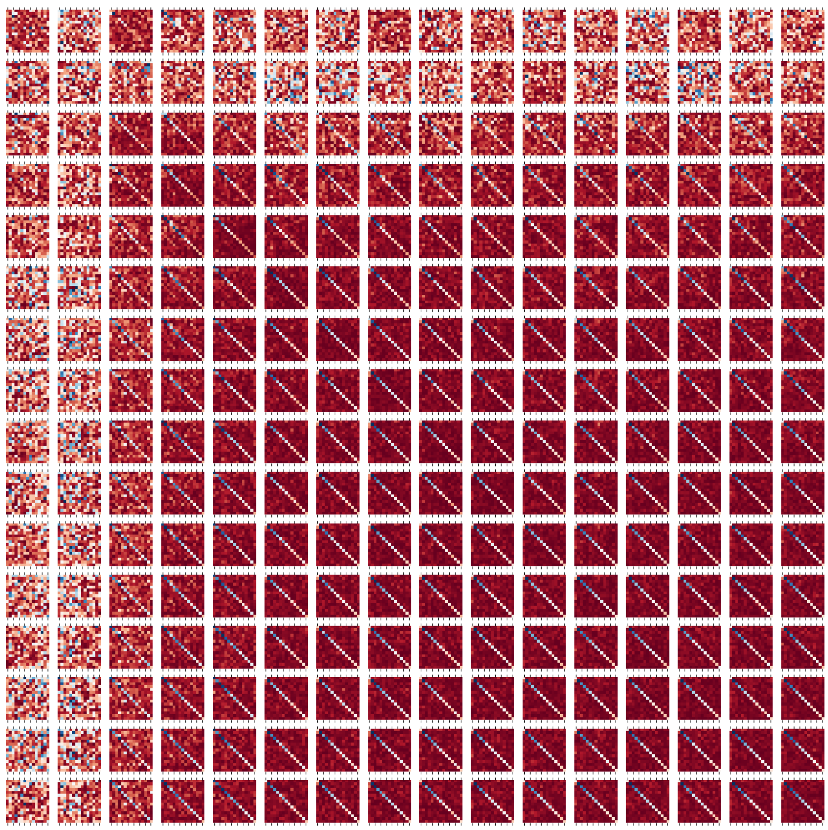


Figure 27: Fully-connected ResNet34 (Type 1 model) trained on CIFAR100.

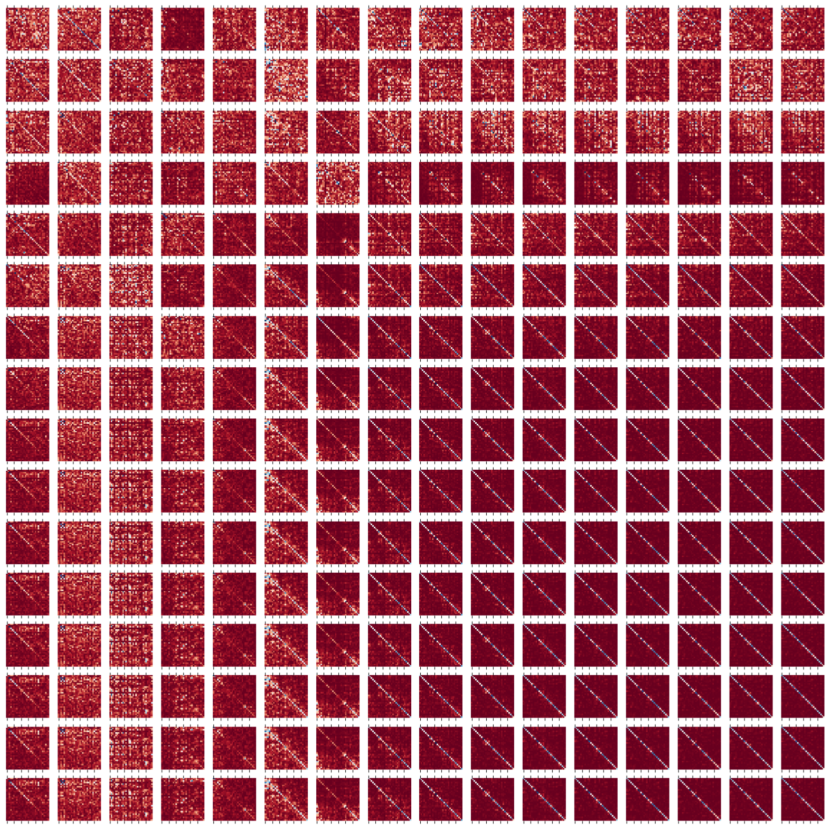


Figure 28: Convolutional ResNet34 (Type 2 model) trained on MNIST.

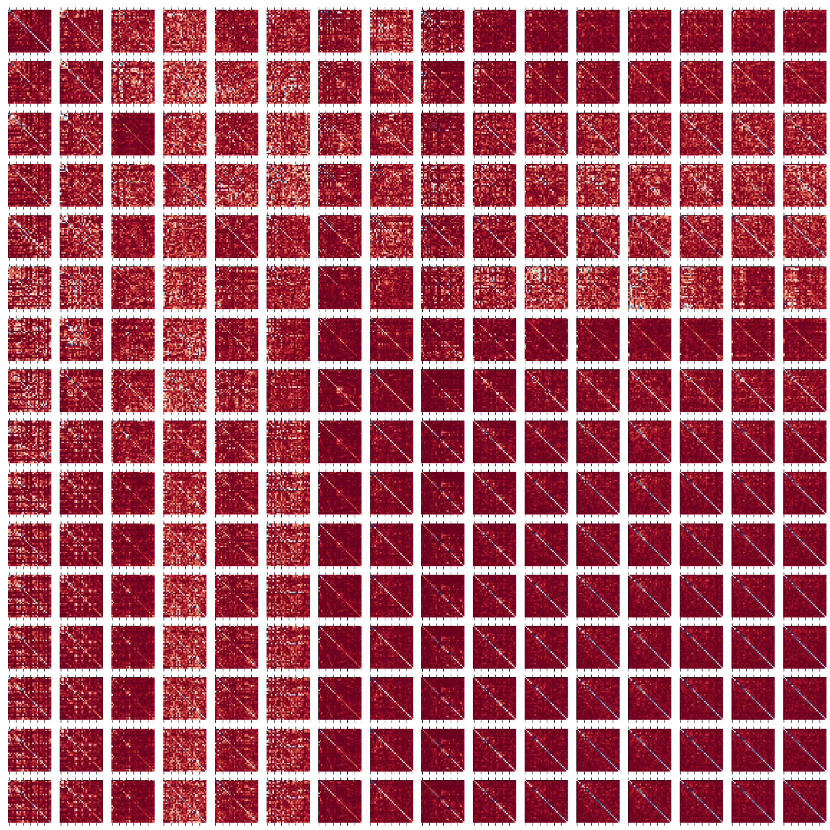


Figure 29: Convolutional ResNet34 (Type 2 model) trained on FashionMNIST.

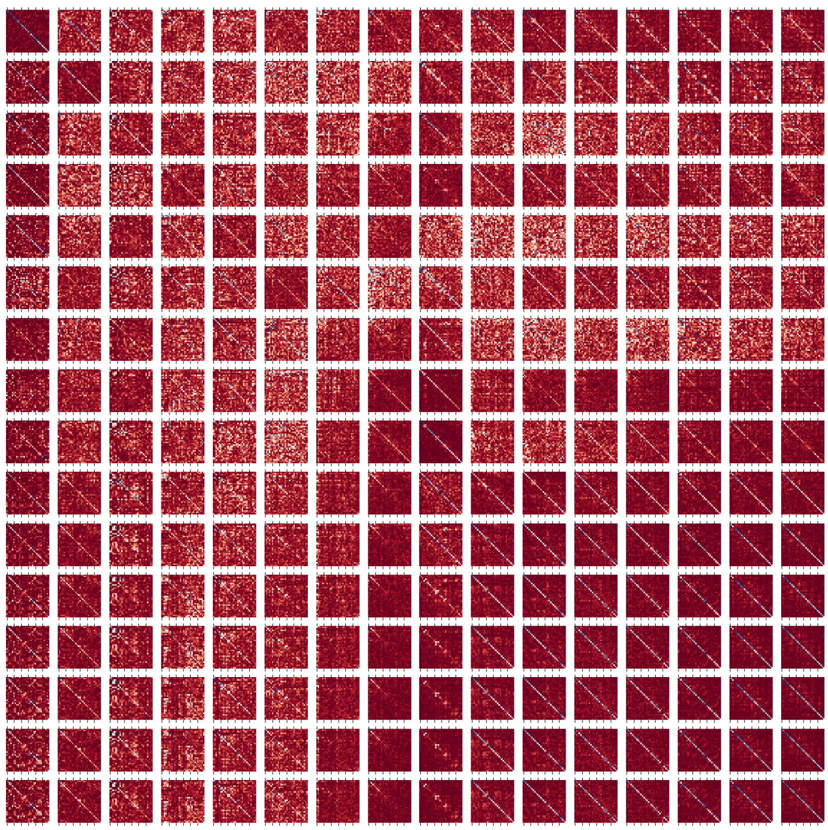


Figure 30: Convolutional ResNet34 (Type 2 model) trained on CIFAR10.

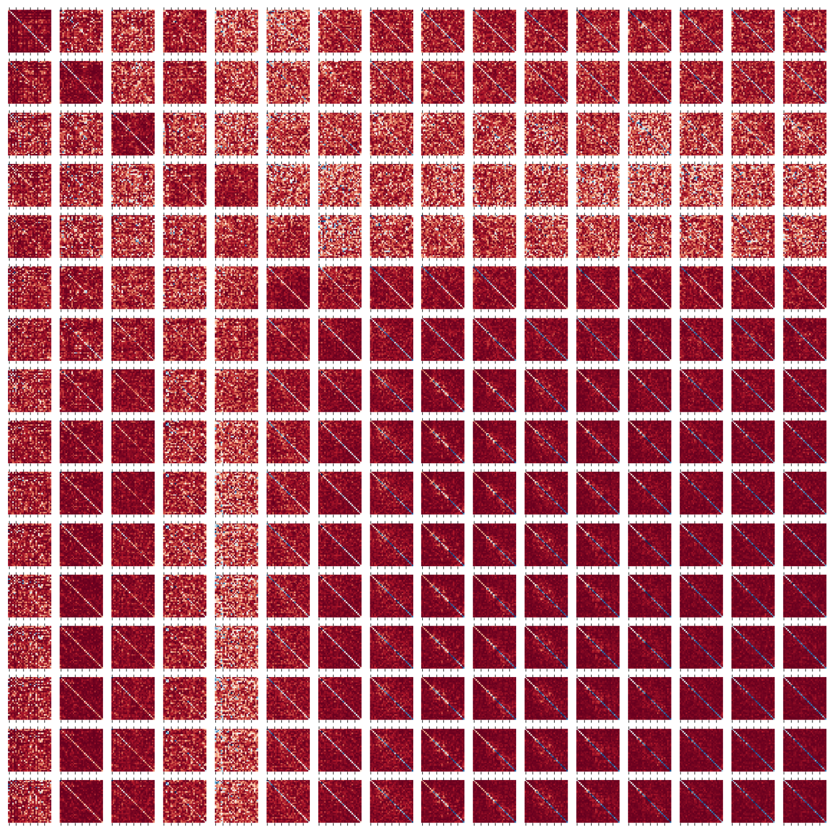


Figure 31: Convolutional ResNet34 (Type 2 model) trained on CIFAR100.

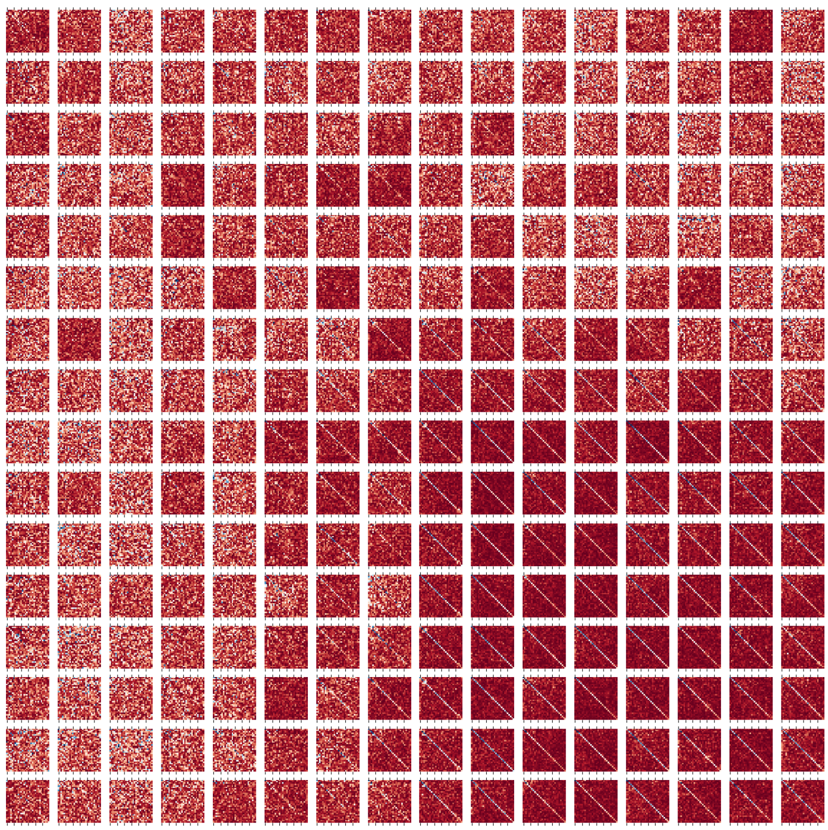


Figure 32: Convolutional ResNet34 (Type 2 model) trained on ImageNette.

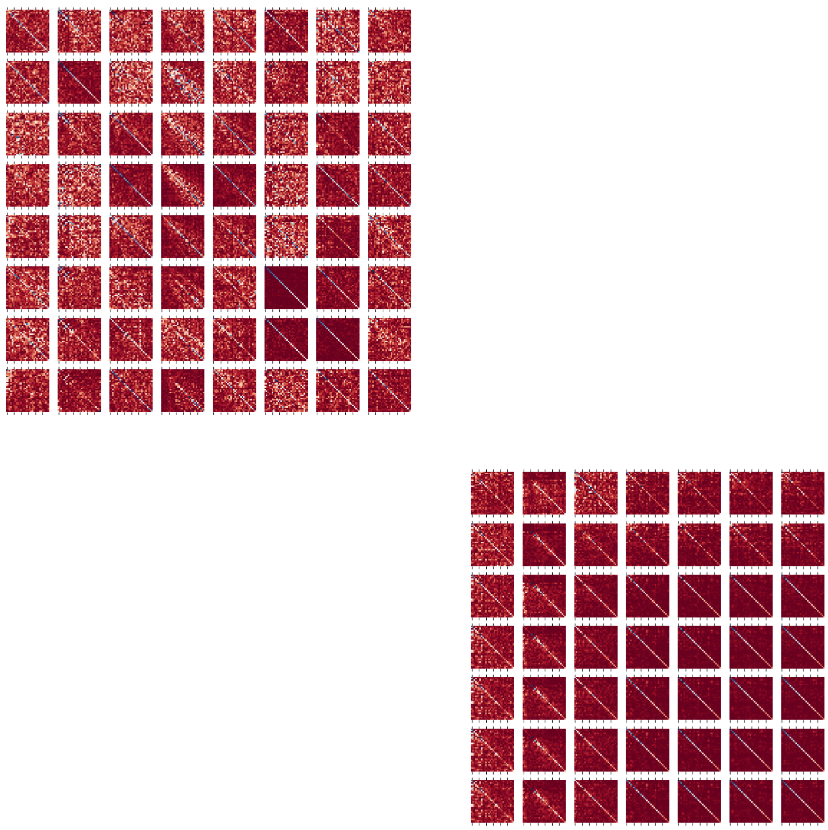


Figure 33: Convolutional ResNet34 with downsampling (Type 3 model) trained on MNIST.

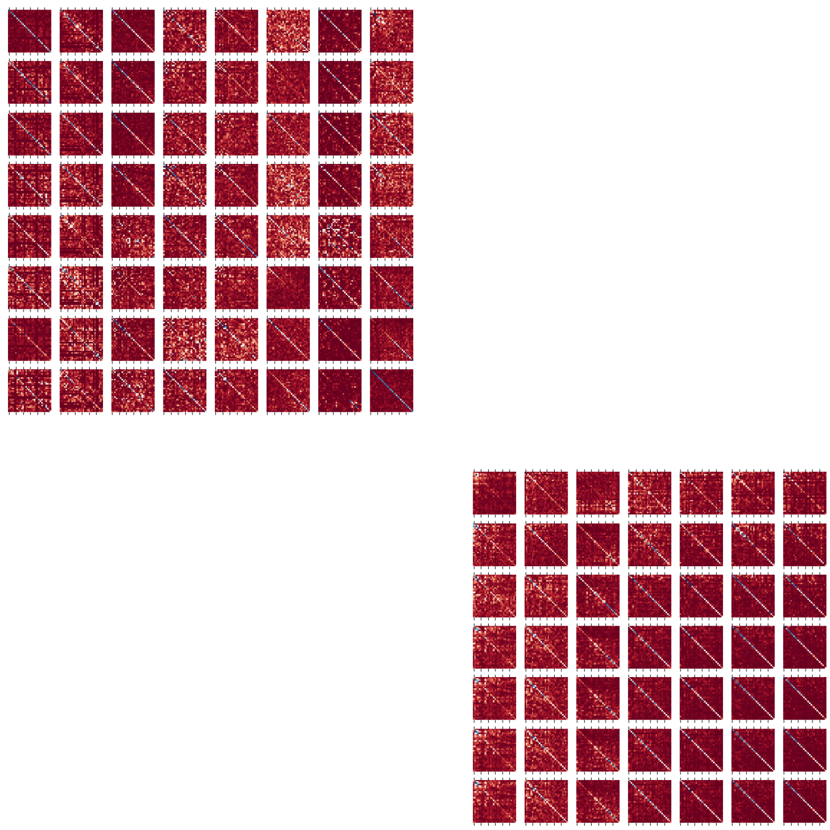


Figure 34: Convolutional ResNet34 with downsampling (Type 3 model) trained on FashionMNIST.

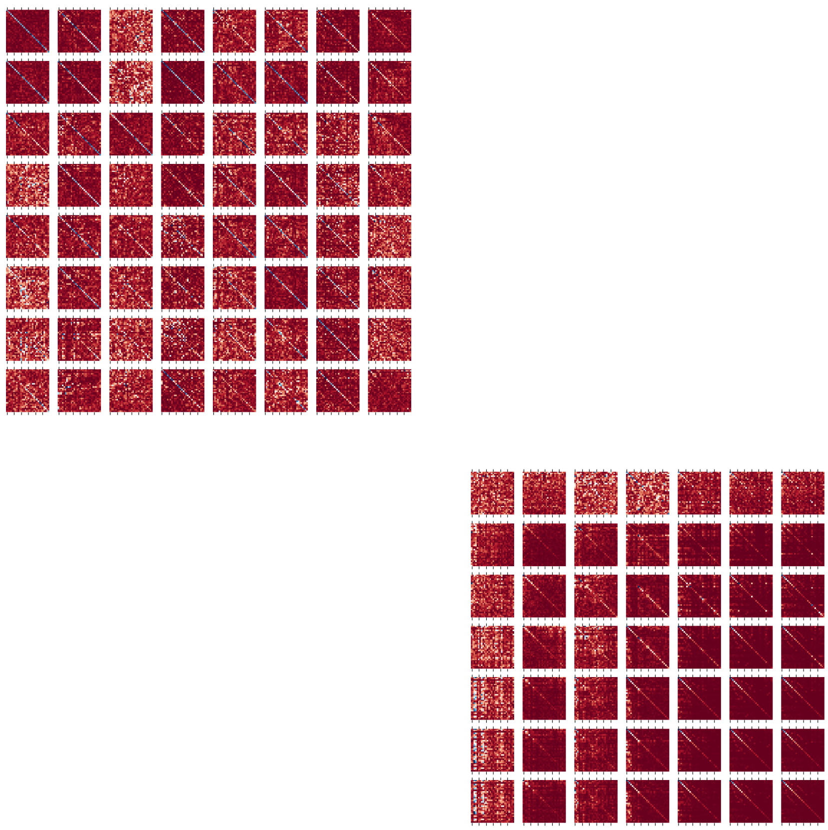
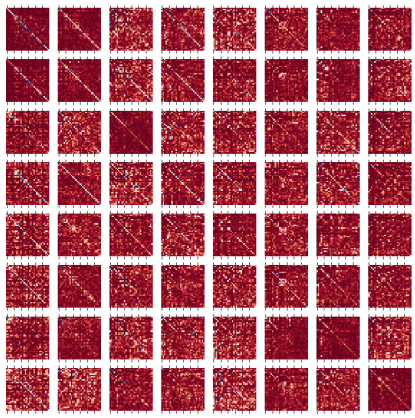


Figure 35: Convolutional ResNet34 with downsampling (Type 3 model) trained on CIFAR10.



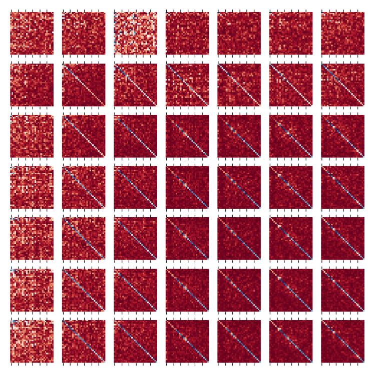
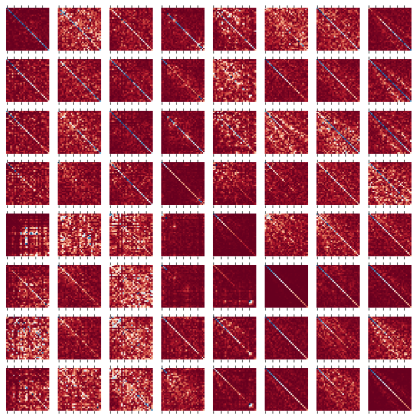


Figure 36: Convolutional ResNet34 with downsampling (Type 3 model) trained on CIFAR100.



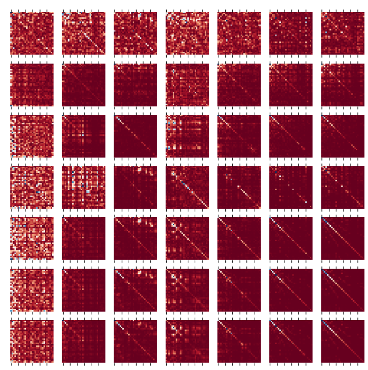


Figure 37: Convolutional ResNet34 with downsampling (Type 3 model) trained on ImageNette.

3.4 (RA3) and (RA4)

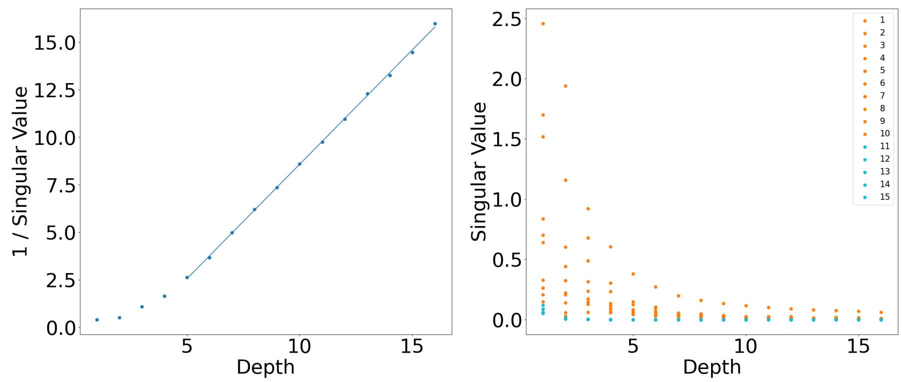


Figure 38: Fully-connected ResNet34 (Type 1 model) trained on MNIST.

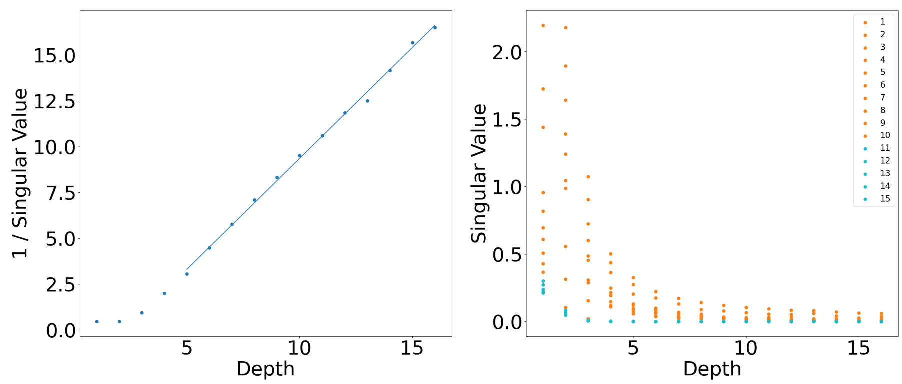


Figure 39: Fully-connected ResNet34 (Type 1 model) trained on FashionMNIST.

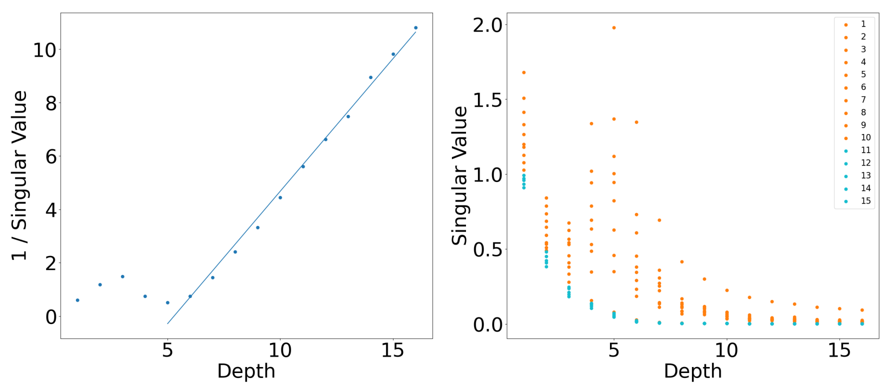


Figure 40: Fully-connected ResNet34 (Type 1 model) trained on CIFAR10.

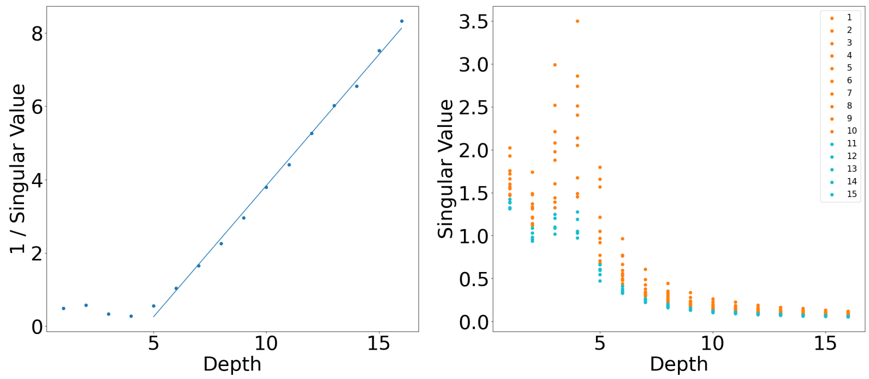


Figure 41: Fully-connected ResNet34 (Type 1 model) trained on CIFAR100.

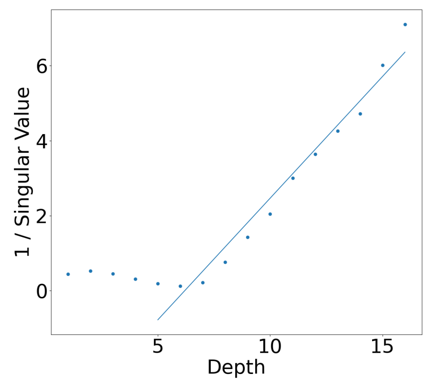


Figure 42: Convolutional ResNet34 (Type 2 model) trained on MNIST.

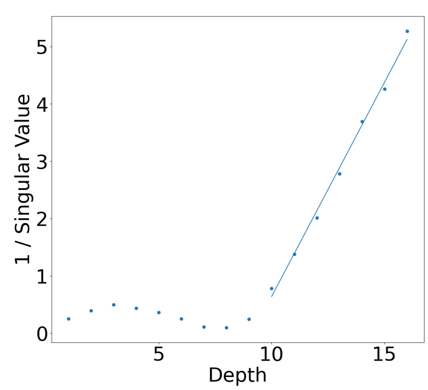


Figure 43: Convolutional ResNet34 (Type 2 model) trained on FashionMNIST.

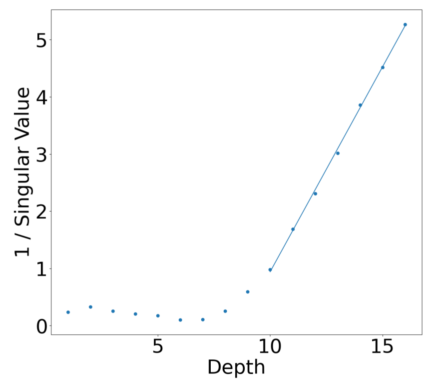


Figure 44: Convolutional ResNet34 (Type 2 model) trained on CIFAR10.

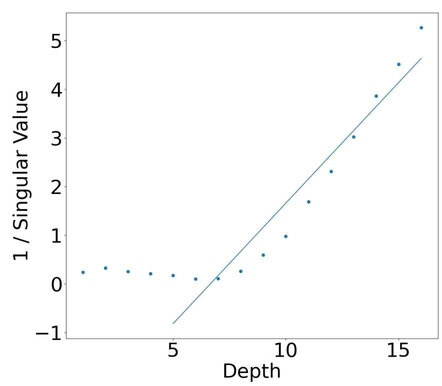


Figure 45: Convolutional ResNet34 (Type 2 model) trained on CIFAR100.

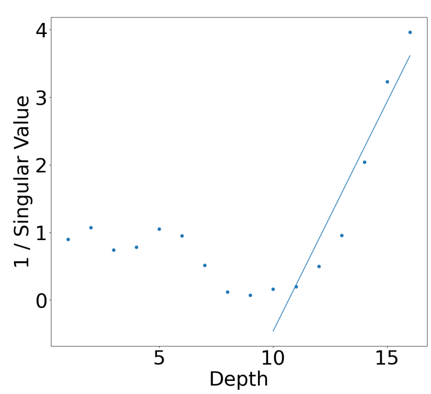


Figure 46: Convolutional ResNet34 (Type 2 model) trained on ImageNette.

4 Broader Impacts

This work presents foundational research and we do not foresee any potential negative societal impacts.

References

Hector Miranda and Robert C Thompson. A trace inequality with a subtracted term. *Linear algebra and its applications*, 185:165–172, 1993.

Leon Mirsky. A trace inequality of john von neumann. *Monatshefte für mathematik*, 79(4):303–306, 1975.