A. Proof of Theorem 1

Theorem. *Let X and Y be $n \times p$ matrices. Suppose s is invariant to invertible linear transformation in the first argument,* i.e. $s(X, Z) = s(XA, Z)$ *for arbitrary Z and any A with rank$(A) = p$. If rank(X) = rank$(Y) = n$, then $s(X, Z) = s(Y, Z)$.*

Proof. Let

$$X' = \begin{bmatrix} X \\ K_X \end{bmatrix} \qquad\qquad Y' = \begin{bmatrix} Y \\ K_Y \end{bmatrix},$$

where K_X is a basis for the null space of the rows of X and K_Y is a basis for the null space of the rows of Y. Then let $A = X'^{-1}Y'$.

$$\begin{bmatrix} X \\ K_X \end{bmatrix} A = \begin{bmatrix} Y \\ K_Y \end{bmatrix} \implies XA = Y.$$

Because X' and Y' have rank p by construction, A also has rank p. Thus, $s(X, Z) = s(XA, Z) = s(Y, Z)$. □

B. Orthogonalization and Invariance to Invertible Linear Transformation

Here we show that any similarity index that is invariant to orthogonal transformation can be made invariant to invertible linear transformation by orthogonalizing the columns of the input.

Proposition 1. *Let X be an $n \times p$ matrix of full column rank and let A be an invertible $p \times p$ matrix. Let $X = Q_X R_X$ and $XA = Q_{XA} R_{XA}$, where $Q_X^T Q_X = Q_{XA}^T Q_{XA} = I$ and R_X and R_{XA} are invertible. If $s(\cdot, \cdot)$ is invariant to orthogonal transformation, then $s(Q_X, Y) = s(Q_{XA}, Y)$.*

Proof. Let $B = R_X A R_{XA}^{-1}$. Then $Q_X B = Q_{XA}$, and B is an orthogonal transformation:

$$B^{\mathrm{T}} B = B^{\mathrm{T}} Q_X^{\mathrm{T}} Q_X B = Q_{XA}^{\mathrm{T}} Q_{XA} = I.$$

Thus $s(Q_X, Y) = s(Q_X B, Y) = s(Q_{XA}, Y)$. □

C. CCA and Linear Regression

C.1. Linear Regression

Consider the linear regression fit of the columns of an $n \times m$ matrix C with an $n \times p$ matrix A:

$$\hat{B} = \arg\min_B ||C - AB||_{\mathrm{F}}^2 = (A^{\mathrm{T}} A)^{-1} A^{\mathrm{T}} C.$$

Let $A = QR$, the thin QR decomposition of A. Then the fitted values are given by:

$$\begin{aligned} \hat{C} &= A\hat{B} \\ &= A(A^{\mathrm{T}} A)^{-1} A^{\mathrm{T}} C \\ &= QR(R^{\mathrm{T}} Q^{\mathrm{T}} QR)^{-1} R^{\mathrm{T}} Q^{\mathrm{T}} C \\ &= QRR^{-1}(R^{\mathrm{T}})^{-1} R^{\mathrm{T}} Q^{\mathrm{T}} C \\ &= QQ^{\mathrm{T}} C. \end{aligned}$$

The residuals $E = C - \hat{C}$ are orthogonal to the fitted values, *i.e.*

$$\begin{aligned} E^{\mathrm{T}} \hat{C} &= (C - QQ^{\mathrm{T}} C)^{\mathrm{T}} QQ^{\mathrm{T}} C \\ &= C^{\mathrm{T}} QQ^{\mathrm{T}} C - C^{\mathrm{T}} QQ^{\mathrm{T}} C = 0. \end{aligned}$$

Thus:

$$\begin{aligned} ||E||_{\mathrm{F}}^2 &= \mathrm{tr}(E^{\mathrm{T}}E) \\ &= \mathrm{tr}(E^{\mathrm{T}}C - E^{\mathrm{T}}\hat{C}) \\ &= \mathrm{tr}((C-\hat{C})^{\mathrm{T}}C) \\ &= \mathrm{tr}(C^{\mathrm{T}}C) - \mathrm{tr}(C^{\mathrm{T}}QQ^{\mathrm{T}}C) \\ &= ||C||_{\mathrm{F}}^2 - ||Q^{\mathrm{T}}C||_{\mathrm{F}}^2. \end{aligned} \tag{15}$$

Assuming that C was centered by subtracting its column means prior to the linear regression fit, the total fraction of variance explained by the fit is:

$$R^2 = 1 - \frac{||E||_{\mathrm{F}}^2}{||C||_{\mathrm{F}}^2} = 1 - \frac{||C||_{\mathrm{F}}^2 - ||Q^{\mathrm{T}}C||_{\mathrm{F}}^2}{||C||_{\mathrm{F}}^2} = \frac{||Q^{\mathrm{T}}C||_{\mathrm{F}}^2}{||C||_{\mathrm{F}}^2}. \tag{16}$$

Although we have assumed that Q is obtained from QR decomposition, any orthonormal basis with the same span will suffice, because orthogonal transformations do not change the Frobenius norm.

C.2. CCA

Let X be an $n \times p_1$ matrix and Y be an $n \times p_2$ matrix, and let $p = \min(p_1, p_2)$. Given the thin QR decompositions of X and Y, $X = Q_X R_X$, $Y = Q_Y R_Y$ such that $Q_X^{\mathrm{T}} Q_X = I$, $Q_Y^{\mathrm{T}} Q_Y = I$, the canonical correlations ρ_i are the singular values of $A = Q_X^{\mathrm{T}} Q_Y$ (Björck & Golub, 1973; Press, 2011) and thus the square roots of the eigenvalues of $A^{\mathrm{T}}A$. The squared canonical correlations ρ_i^2 are the eigenvalues of $A^{\mathrm{T}}A = Q_Y^{\mathrm{T}} Q_X Q_X^{\mathrm{T}} Q_Y$. Their sum is $\sum_{i=1}^{p} \rho_i^2 = \mathrm{tr}(A^{\mathrm{T}}A) = ||Q_Y^{\mathrm{T}} Q_X||_{\mathrm{F}}^2$.

Now consider the linear regression fit of the columns of Q_X with Y. Assume that Q_X has zero mean. Substituting Q_Y for Q and Q_X for C in Equation 16, and noting that $||Q_X||_{\mathrm{F}}^2 = p_1$:

$$R^2 = \frac{||Q_Y^{\mathrm{T}} Q_X||_{\mathrm{F}}^2}{p_1} = \frac{\sum_{i=1}^{p} \rho_i^2}{p_1}. \tag{17}$$

C.3. Projection-Weighted CCA

Morcos et al. (2018) proposed to compute projection-weighted canonical correlation as:

$$\bar{\rho}_{\mathrm{PW}} = \frac{\sum_{i=1}^{c} \alpha_i \rho_i}{\sum_{i=1} \alpha_i} \qquad\qquad \alpha_i = \sum_j |\langle \mathbf{h}_i, \mathbf{x}_j \rangle|,$$

where the $\mathbf{x}_j$ are the columns of X, and the $\mathbf{h}_i$ are the canonical variables formed by projecting X to the canonical coordinate frame. Below, we show that if we modify $\bar{\rho}_{\mathrm{PW}}$ by squaring the dot products and ρ_i, we recover linear regression. Specifically:

$$R^2_{\mathrm{MPW}} = \frac{\sum_{i=1}^{c} \alpha'_i \rho_i^2}{\sum_{i=1} \alpha'_i} = R^2_{\mathrm{LR}} \qquad\qquad \alpha'_i = \sum_j \langle \mathbf{h}_i, \mathbf{x}_j \rangle^2.$$

Our derivation begins by forming the SVD $Q_X^{\mathrm{T}} Q_Y = U \Sigma V^{\mathrm{T}}$. Σ is a diagonal matrix of the canonical correlations ρ_i, and the matrix of canonical variables $H = Q_X U$. Then R^2_{MPW} is:

$$\begin{aligned} R^2_{\mathrm{MPW}} &= \frac{||X^{\mathrm{T}} H \Sigma||_{\mathrm{F}}^2}{||X^{\mathrm{T}} H||_{\mathrm{F}}^2} \\ &= \frac{\mathrm{tr}((X^{\mathrm{T}} H \Sigma)^{\mathrm{T}} (X^{\mathrm{T}} H \Sigma))}{\mathrm{tr}((X^{\mathrm{T}} H)^{\mathrm{T}} (X^{\mathrm{T}} H))} \\ &= \frac{\mathrm{tr}(\Sigma H^{\mathrm{T}} X X^{\mathrm{T}} H \Sigma)}{\mathrm{tr}(H^{\mathrm{T}} X X^{\mathrm{T}} H)} \\ &= \frac{\mathrm{tr}(X^{\mathrm{T}} H \Sigma^2 H^{\mathrm{T}} X)}{\mathrm{tr}(X^{\mathrm{T}} H H^{\mathrm{T}} X)} \\ &= \frac{\mathrm{tr}(R_X^{\mathrm{T}} Q_X^{\mathrm{T}} H \Sigma^2 H^{\mathrm{T}} Q_X R_X)}{\mathrm{tr}(R_X^{\mathrm{T}} Q_X^{\mathrm{T}} Q_X U U^{\mathrm{T}} Q_X^{\mathrm{T}} Q_X R_X)}. \end{aligned} \tag{18}$$

Noting that $Q_X^{\mathrm{T}} H = U$ and $U\Sigma = Q_X^{\mathrm{T}} Q_Y V$:

$$\begin{aligned} R^2_{\text{MPW}} &= \frac{\text{tr}(R_X^{\mathrm{T}} U \Sigma^2 U^{\mathrm{T}} R_X)}{\text{tr}(R_X^{\mathrm{T}} Q_X^{\mathrm{T}} Q_X R_X)} \\ &= \frac{\text{tr}(R_X^{\mathrm{T}} Q_X^{\mathrm{T}} Q_Y V \Sigma U^{\mathrm{T}} R_X)}{\text{tr}(X^{\mathrm{T}} X)} \\ &= \frac{\text{tr}(X^{\mathrm{T}} Q_Y Q_Y^{\mathrm{T}} Q_X R_X)}{\text{tr}(X^{\mathrm{T}} X)} \\ &= \frac{\text{tr}(X^{\mathrm{T}} Q_Y Q_Y^{\mathrm{T}} X)}{\text{tr}(X^{\mathrm{T}} X)} \\ &= \frac{||Q_Y^{\mathrm{T}} X||_{\mathrm{F}}^2}{||X||_{\mathrm{F}}^2}. \end{aligned}$$

Substituting Q_Y for Q and X for C in Equation 16:

$$R^2_{\text{LR}} = \frac{||Q_Y^{\mathrm{T}} X||_{\mathrm{F}}^2}{||X||_{\mathrm{F}}^2} = R^2_{\text{MPW}}.$$

D. Notes on Other Methods

D.1. Canonical Ridge

Beyond CCA, we could also consider the "canonical ridge" regularized CCA objective (Vinod, 1976):

$$\begin{aligned} \sigma_i = \max_{\mathbf{w}_X^i, \mathbf{w}_Y^i} & \frac{(X\mathbf{w}_X^i)^{\mathrm{T}}(Y\mathbf{w}_Y^i)}{\sqrt{||X\mathbf{w}_X^i||^2 + \kappa_X||\mathbf{w}_X^i||_2^2}\sqrt{||Y\mathbf{w}_Y^i||^2 + \kappa_Y||\mathbf{w}_Y^i||^2}} \\ \text{subject to} \quad & \forall_{j<i} \;\; (\mathbf{w}_X^i)^{\mathrm{T}}(X^{\mathrm{T}}X + \kappa I)\mathbf{w}_X^j = 0 \\ & \forall_{j<i} \;\; (\mathbf{w}_Y^i)^{\mathrm{T}}(Y^{\mathrm{T}}Y + \kappa I)\mathbf{w}_Y^j = 0. \end{aligned} \tag{19}$$

Given the singular value decompositions $X = U_X \Sigma_X V_X^{\mathrm{T}}$ and $Y = U_Y \Sigma_Y V_Y^{\mathrm{T}}$, one can form "partially orthogonalized" bases $\tilde{Q}_X = U_X \Sigma_X (\Sigma_X^2 + \kappa_X I)^{-1/2}$ and $\tilde{Q}_Y = U_Y \Sigma_Y (\Sigma_Y^2 + \kappa_Y I)^{-1/2}$. Given the singular value decomposition of their product $\tilde{U}\tilde{\Sigma}\tilde{V}^{\mathrm{T}} = \tilde{Q}_X^{\mathrm{T}} \tilde{Q}_Y$, the canonical weights are given by $W_X = V_X(\Sigma_X^2 + \kappa_X I)^{-1/2}\tilde{U}$ and $W_Y = V_Y(\Sigma_Y^2 + \kappa_Y I)^{-1/2}\tilde{V}$, as previously shown by Mroueh et al. (2015). As in the unregularized case (Equation 13), there is a convenient expression for the sum of the squared singular values $\sum \tilde{\sigma}_i^2$ in terms of the eigenvalues and eigenvectors of XX^{T} and YY^{T}. Let the i^{th} left-singular vector of X (eigenvector of XX^{T}) be indexed as $\mathbf{u}_X^{\mathrm{i}}$ and let the i^{th} eigenvalue of XX^{T} (squared singular value of X) be indexed as λ_X^i, and similarly let the left-singular vectors of YY^{T} be indexed as $\mathbf{u}_Y^{\mathrm{i}}$ and the eigenvalues as λ_Y^i. Then:

$$\sum_{i=1}^{p_1} \tilde{\sigma}_i^2 = ||\tilde{Q}_Y^{\mathrm{T}} \tilde{Q}_X||_{\mathrm{F}}^2 \tag{20}$$

$$= ||(\Sigma_Y^2 + \kappa_Y I)^{-1/2} \Sigma_Y U_Y^{\mathrm{T}} U_X \Sigma_X (\Sigma_X^2 + \kappa_X I)^{-1/2}||_{\mathrm{F}}^2 \tag{21}$$

$$= \sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \frac{\lambda_X^i \lambda_Y^j}{(\lambda_X^i + \kappa_X)(\lambda_Y^j + \kappa_Y)} \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2. \tag{22}$$

Unlike in the unregularized case, the singular values σ_i do not measure the correlation between the canonical variables. Instead, they become arbitrarily small as κ_X or κ_Y increase. Thus, we need to normalize the statistic to remove the dependency on the regularization parameters.

Applying von Neumann's trace inequality yields a bound:

$$\sum_{i=1}^{p_1} \tilde{\sigma}_i^2 = \text{tr}(\tilde{Q}_Y \tilde{Q}_Y^{\text{T}} \tilde{Q}_X \tilde{Q}_X^{\text{T}}) \tag{23}$$

$$= \text{tr}((U_Y \Sigma_Y^2 (\Sigma_Y^2 + \kappa_Y I)^{-1} U_Y^{\text{T}})(U_X \Sigma_X^2 (\Sigma_X^2 + \kappa_X I)^{-1} U_X^{\text{T}})) \tag{24}$$

$$\leq \sum_{i=1}^{p_1} \frac{\lambda_X^i \lambda_Y^i}{(\lambda_X^i + \kappa_X)(\lambda_Y^i + \kappa_Y)}. \tag{25}$$

Applying the Cauchy-Schwarz inequality to (25) yields the alternative bounds:

$$\sum_{i=1}^{p_1} \tilde{\sigma}_i^2 \leq \sqrt{\sum_{i=1}^{p_1} \left(\frac{\lambda_X^i}{\lambda_X^i + \kappa_X}\right)^2} \sqrt{\sum_{i=1}^{p_1} \left(\frac{\lambda_Y^i}{\lambda_Y^i + \kappa_Y}\right)^2} \tag{26}$$

$$\leq \sqrt{\sum_{i=1}^{p_1} \left(\frac{\lambda_X^i}{\lambda_X^i + \kappa_X}\right)^2} \sqrt{\sum_{i=1}^{p_2} \left(\frac{\lambda_Y^i}{\lambda_Y^i + \kappa_Y}\right)^2}. \tag{27}$$

A normalized form of (22) could be produced by dividing by any of (25), (26), or (27).

If $\kappa_X = \kappa_Y = 0$, then (25) and (26) are equal to p_1. In this case, (22) is simply the sum of the squared canonical correlations, so normalizing by either of these bounds recovers R^2_{CCA}.

If $\kappa_Y = 0$, then as $\kappa_X \to \infty$, normalizing by the bound from (25) recovers R^2:

$$\lim_{\kappa_X \to \infty} \frac{\sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \frac{\lambda_X^i \lambda_Y^j}{(\lambda_X^i + \kappa_X)(\lambda_Y^j + 0)} \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2}{\sum_{i=1}^{p_1} \frac{\lambda_X^i \lambda_Y^i}{(\lambda_X^i + \kappa_X)(\lambda_Y^i + 0)}} \tag{28}$$

$$= \lim_{\kappa_X \to \infty} \frac{\sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \frac{\lambda_X^i}{\left(\frac{\lambda_X^i}{\kappa_X} + 1\right)} \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2}{\sum_{i=1}^{p_1} \frac{\lambda_X^i}{\left(\frac{\lambda_X^i}{\kappa_X} + 1\right)}} \tag{29}$$

$$= \frac{\sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \lambda_X^i \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2}{\sum_{i=1}^{p_1} \lambda_X^i} \tag{30}$$

$$= \frac{||U_Y^{\text{T}} U_X \Sigma_X||_{\text{F}}^2}{||X||_{\text{F}}^2} = \frac{||Q_Y^{\text{T}} X||_{\text{F}}^2}{||X||_{\text{F}}^2} = R^2_{\text{LR}}. \tag{31}$$

The bound from (27) differs from the bounds in (25) and (26) because it is multiplicatively separable in X and Y. Normalizing by this bound leads to $\text{CKA}(\tilde{Q}_X \tilde{Q}_X^{\text{T}}, \tilde{Q}_Y \tilde{Q}_Y^{\text{T}})$:

$$\frac{\sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \frac{\lambda_X^i \lambda_Y^j}{(\lambda_X^i + \kappa_X)(\lambda_Y^j + \kappa_Y)} \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2}{\sqrt{\sum_{i=1}^{p_1} \left(\frac{\lambda_X^i}{\lambda_X^i + \kappa_X}\right)^2} \sqrt{\sum_{i=1}^{p_2} \left(\frac{\lambda_Y^i}{\lambda_Y^i + \kappa_Y}\right)^2}} \tag{32}$$

$$= \frac{||\tilde{Q}_Y^{\text{T}} \tilde{Q}_X||_{\text{F}}^2}{||\tilde{Q}_X^{\text{T}} \tilde{Q}_X||_{\text{F}} ||\tilde{Q}_Y^{\text{T}} \tilde{Q}_Y||_{\text{F}}} = \text{CKA}(\tilde{Q}_X \tilde{Q}_X^{\text{T}}, \tilde{Q}_Y \tilde{Q}_Y^{\text{T}}). \tag{33}$$

Moreover, setting $\kappa_X = \kappa_Y = \kappa$ and taking the limit as $\kappa \to \infty$, the normalization from (27) leads to $\text{CKA}(XX^\text{T}, YY^\text{T})$:

$$\lim_{\kappa\to\infty} \frac{\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \frac{\lambda_X^i \lambda_Y^j}{(\lambda_X^i+\kappa)(\lambda_Y^j+\kappa)} \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2}{\sqrt{\sum_{i=1}^{p_1}\left(\frac{\lambda_X^i}{\lambda_X^i+\kappa}\right)^2}\sqrt{\sum_{i=1}^{p_2}\left(\frac{\lambda_Y^i}{\lambda_Y^i+\kappa}\right)^2}} \tag{34}$$

$$= \lim_{\kappa\to\infty} \frac{\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \frac{\lambda_X^i \lambda_Y^j}{\left(\frac{\lambda_X^i}{\kappa}+1\right)\left(\frac{\lambda_Y^j}{\kappa}+1\right)} \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2}{\sqrt{\sum_{i=1}^{p_1}\left(\frac{\lambda_X^i}{\frac{\lambda_X^i}{\kappa}+1}\right)^2}\sqrt{\sum_{i=1}^{p_2}\left(\frac{\lambda_Y^i}{\frac{\lambda_Y^i}{\kappa}+1}\right)^2}} \tag{35}$$

$$= \frac{\sum_{i=1}^{p_1}\sum_{j=1}^{p_2} \lambda_X^i \lambda_Y^j \langle \mathbf{u}_X^i, \mathbf{u}_Y^j \rangle^2}{\sqrt{\sum_{i=1}^{p_1}\left(\lambda_X^i\right)^2}\sqrt{\sum_{i=1}^{p_2}\left(\lambda_Y^i\right)^2}} \tag{36}$$

$$= \text{CKA}(XX^\text{T}, YY^\text{T}).$$

Overall, the hyperparameters of the canonical ridge objective make it less useful for exploratory analysis. These hyperparameters could be selected by cross-validation, but this is computationally expensive, and the resulting estimator would be biased by sample size. Moreover, our goal is not to map representations of networks to a common space, but to measure the similarity between networks. Appropriately chosen regularization will improve out-of-sample performance of the mapping, but it makes the meaning of "similarity" more ambiguous.

D.2. The Orthogonal Procrustes Problem

The orthogonal Procrustes problem consists of finding an orthogonal rotation in feature space that produces the smallest error:

$$\hat{Q} = \arg\min_Q ||Y - XQ||_\text{F}^2 \text{ subject to } Q^\text{T}Q = I. \tag{37}$$

The objective can be written as:

$$\begin{aligned} ||Y - XQ||_\text{F}^2 &= \text{tr}((Y - XQ)^\text{T}(Y - XQ)) \\ &= \text{tr}(Y^\text{T}Y) - \text{tr}(Y^\text{T}XQ) - \text{tr}(Q^\text{T}X^\text{T}Y) + \text{tr}(Q^\text{T}X^\text{T}XQ) \\ &= ||Y||_\text{F}^2 + ||X||_\text{F}^2 - 2\text{tr}(Y^\text{T}XQ). \end{aligned} \tag{38}$$

Thus, an equivalent objective is:

$$\hat{Q} = \arg\max_Q \text{tr}(Y^\text{T}XQ) \text{ subject to } Q^\text{T}Q = I. \tag{39}$$

The solution is $\hat{Q} = UV^\text{T}$ where $U\Sigma V^\text{T} = X^\text{T}Y$, the singular value decomposition. At the maximum of (39):

$$\text{tr}(Y^\text{T}X\hat{Q}) = \text{tr}(V\Sigma U^\text{T}UV^\text{T}) = \text{tr}(\Sigma) = ||X^\text{T}Y||_* = ||Y^\text{T}X||_*, \tag{40}$$

which is similar to what we call "dot product-based similarity" (Equation 1), but with the squared Frobenius norm of $Y^\text{T}X$ (the sum of the squared singular values) replaced by the nuclear norm (the sum of the singular values). The Frobenius norm of $Y^\text{T}X$ can be obtained as the solution to a similar optimization problem:

$$||Y^\text{T}X||_\text{F} = \max_W \text{tr}(Y^\text{T}XW) \text{ subject to } \text{tr}(W^\text{T}W) = 1. \tag{41}$$

In the context of neural networks, Smith et al. (2017) previously proposed using the solution to the orthogonal Procrustes problem to align word embeddings from different languages, and demonstrated that it outperformed CCA.

E. Architecture Details

All non-ResNet architectures are based on All-CNN-C (Springenberg et al., 2015), but none are architecturally identical. The Plain-10 model is very similar, but we place the final linear layer after the average pooling layer and use batch normalization because these are common choices in modern architectures. We use these models because they train in minutes on modern hardware.

Tiny-10
3×3 conv. 16-BN-ReLu $\times 2$
3×3 conv. 32 stride 2-BN-ReLu
3×3 conv. 32-BN-ReLu $\times 2$
3×3 conv. 64 stride 2-BN-ReLu
3×3 conv. 64 valid padding-BN-ReLu
1×1 conv. 64-BN-ReLu
Global average pooling
Logits

Table E.1. The Tiny-10 architecture, used in Figures 2, 8, F.3, . The average Tiny-10 model achieved 89.4% accuracy.

Plain-$(8n + 2)$
3×3 conv. 96-BN-ReLu $\times (3n - 1)$
3×3 conv. 96 stride 2-BN-ReLu
3×3 conv. 192-BN-ReLu $\times (3n - 1)$
3×3 conv. 192 stride 2-BN-ReLu
3×3 conv. 192 BN-ReLu $\times (n - 1)$
3×3 conv. 192 valid padding-BN-ReLu
1×1 conv. 192-BN-ReLu $\times n$
Global average pooling
Logits

Table E.2. The Plain-$(8n + 2)$ architecture, used in Figures 3, 5, 7, F.4, F.5, F.6, and F.7. Mean accuracies: Plain-10, 93.9%; Plain-18: 94.8%; Plain-34: 93.7%; Plain-66: 91.3%

Width-n
3×3 conv. n-BN-ReLu $\times 2$
3×3 conv. n stride 2-BN-ReLu
3×3 conv. n-BN-ReLu $\times 2$
3×3 conv. n stride 2-BN-ReLu
3×3 conv. n valid padding-BN-ReLu
1×1 conv. n-BN-ReLu
Global average pooling
Logits

Table E.3. The architectures used for width experiments in Figure 6.

F. Additional Experiments

F.1. Sanity Check for Transformer Encoders

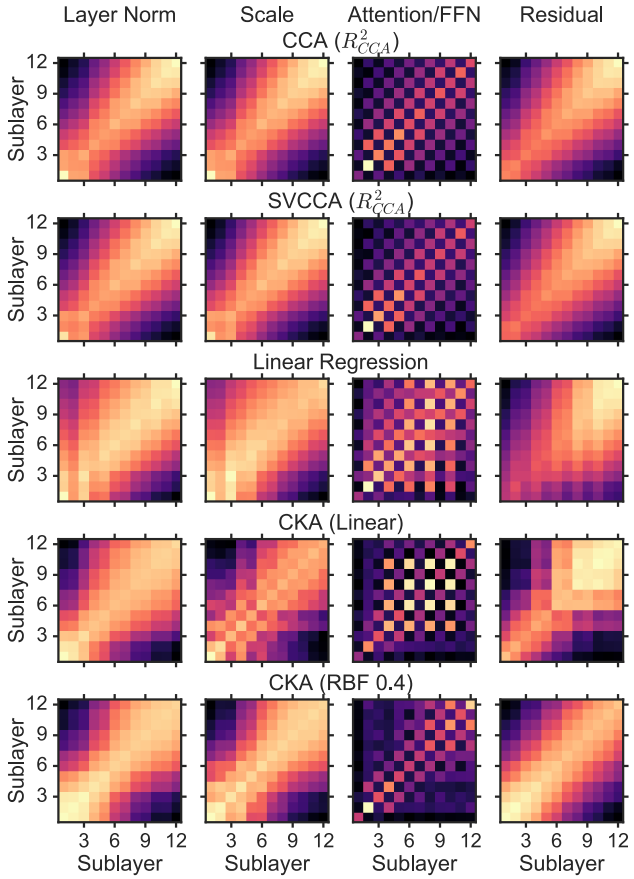

Figure F.1. All similarity indices broadly reflect the structure of Transformer encoders. Similarity indexes are computed between the 12 sublayers of Transformer encoders, for each of the 4 possible places in each sublayer that representations may be taken (see Figure F.2), averaged across 10 models trained from different random initializations.

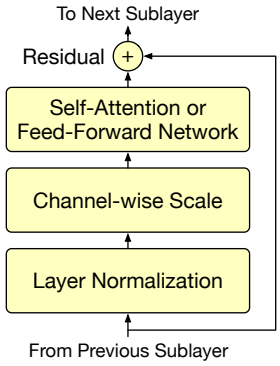

Figure F.2. Architecture of a single sublayer of the Transformer encoder used for our experiments. The full encoder includes 12 sublayers, alternating between self-attention and feed-forward network sublayers.

Index	Layer Norm	Scale	Attn/FFN	Residual
CCA ($\bar{\rho}$)	85.3	85.3	**94.9**	90.9
CCA (R^2_{CCA})	87.8	87.8	**95.3**	**95.2**
SVCCA ($\bar{\rho}$)	78.2	83.0	89.5	75.9
SVCCA (R^2_{CCA})	85.4	86.9	90.8	84.7
PWCCA	**88.5**	88.9	**96.1**	87.0
Linear Reg.	78.1	83.7	76.0	36.9
CKA (Linear)	78.6	**95.6**	86.0	73.6
CKA (RBF 0.2)	76.5	73.1	70.5	76.2
CKA (RBF 0.4)	**92.3**	**96.5**	89.1	**98.1**
CKA (RBF 0.8)	80.8	**95.8**	**93.6**	90.0

Table F.1. Accuracy of identifying corresponding sublayers based maximum similarity, for 10 architecturally identical 12-sublayer Transformer encoders at the 4 locations in each sublayer after which the representation may be taken (see Figure F.2). Results not significantly different from the best result are bold-faced ($p < 0.05$, jackknife z-test).

When applied to Transformer encoders, all similarity indexes we investigated passed the sanity check described in Section 6.1. In Figure F.1, we show similarity between the 12 sublayers of the encoders of 10 Transformer models (45 pairs) (Vaswani et al., 2017) trained from different random initializations to perform English to German translation. Each Transformer sublayer contains four operations, shown in Figure F.2, and results vary based which operation the representation is taken after. Table F.1 shows the accuracy with which we can identify corresponding layers between network pairs by maximal similarity.

The Transformer architecture alternates between self-attention and feed-forward network sublayers. The checkerboard pattern in similarity plots for the Attention/FFN layer in Figure F.1 indicates that representations of feed-forward network sublayers are more similar to other feed-forward network sublayers than to self-attention sublayers, and similarly, representations of self-attention sublayers are more similar to other self-attention sublayers than to feed-forward network layers. CKA also reveals a checkerboard pattern for activations after the channel-wise scale operation (before the self-attention/feed-forward network operation) that other methods do not. Because CCA is invariant to non-isotropic scaling, CCA similarities before and after channel-wise scaling are identical. Thus, CCA cannot capture this structure, even though it is consistent across different networks.

F.2. SVCCA at Alternative Thresholds

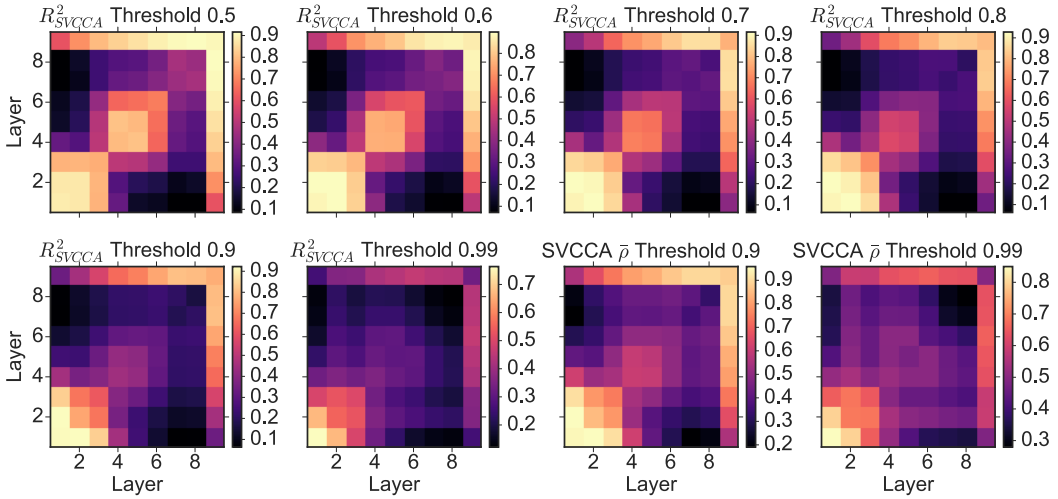

Figure F.3. Same as Figure 2 row 2, but for more SVCCA thresholds than the 0.99 threshold suggested by Raghu et al. (2017). No threshold reveals the structure of the network.

F.3. CKA at Initialization

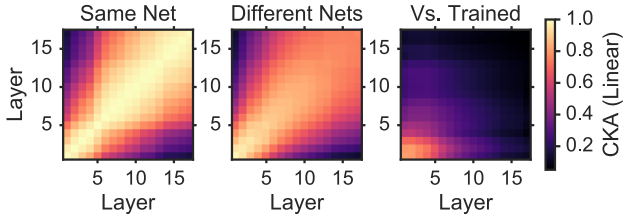

Figure F.4. Similarity of the Plain-18 network at initialization. **Left**: Similarity between layers of the same network. **Middle**: Similarity between untrained networks with different initializations. **Right**: Similarity between untrained and trained networks.

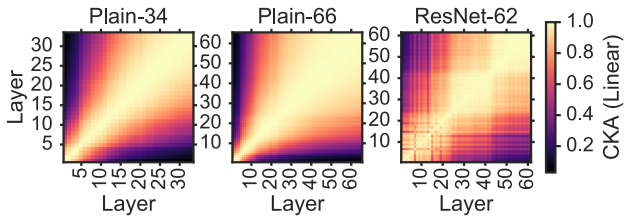

Figure F.5. Similarity between layers at initialization for deeper architectures.

F.4. Additional CKA Results

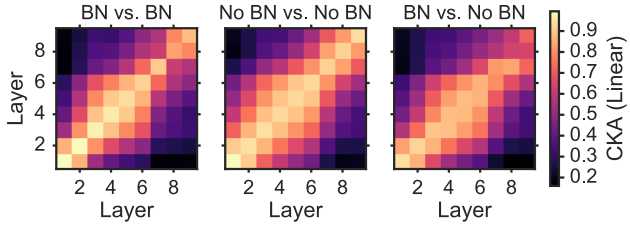

Figure F.6. Networks with and without batch normalization trained from different random initializations learn similar representations according to CKA. The largest difference between networks is at the last convolutional layer. Optimal hyperparameters were separately selected for the batch normalized network (93.9% average accuracy) and the network without batch normalization (91.5% average accuracy).

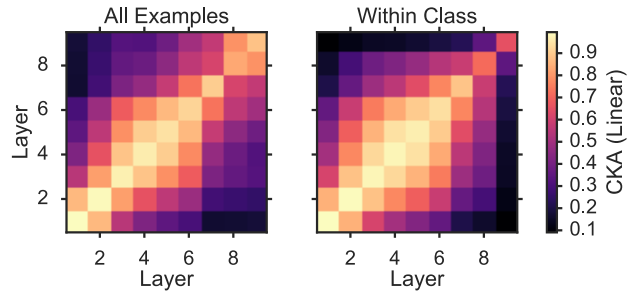

Figure F.7. Within-class CKA is similar to CKA based on all examples. To measure within-class CKA, we computed CKA separately for examples belonging to each CIFAR-10 class based on representations from Plain-10 networks, and averaged the resulting CKA values across classes.

F.5. Similarity Between Different Architectures with Other Indexes

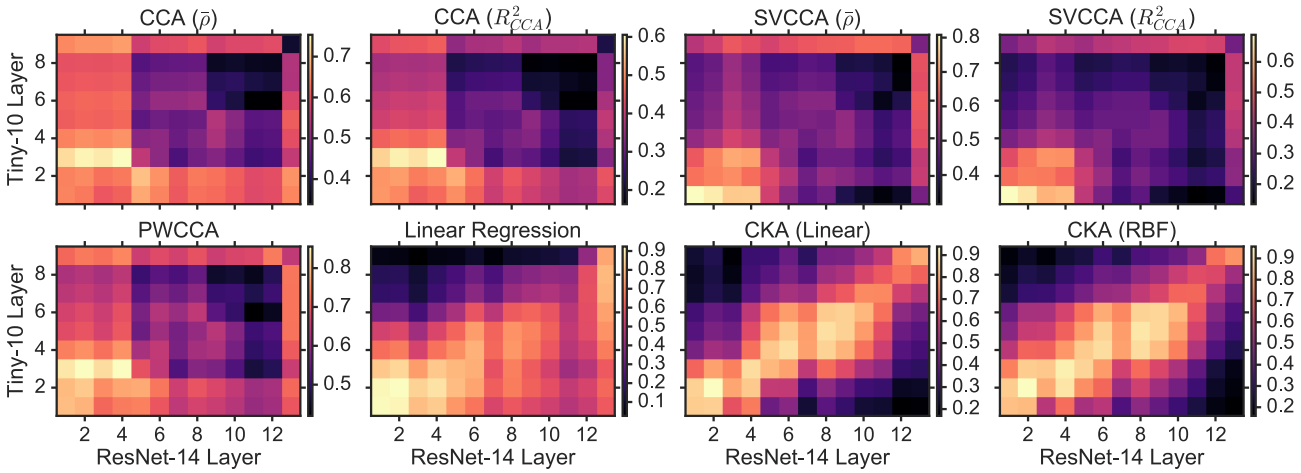

Figure F.8. Similarity between layers of different architectures (Tiny-10 and ResNet-14) for all methods investigated. Only CKA reveals meaningful correspondence. SVCCA results resemble Figure 7 of Raghu et al. (2017). In order to achieve better performance for CCA-based techniques, which are sensitive to the number of examples used to compute similarity, all plots show similarity on the CIFAR-10 training set.