למידה עמוקה וחבורות — תרגיל בית 4

Contents

שאלה 1 2
המרחב $\mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n)$ ובסיס האינדיקטורים (שיטה א׳). 2
כיוון ברנסייד. 3
כיוון האופיינים. 3
מסקנה. 4
שאלה 2 4
סעיף א. חבורת הסימטריה. 5
סעיף ב. שכבה לינארית שקילה (אפינית). 5
סעיף ג. דיאגרמת שיתוף הפרמטרים. 6
סעיף ד. שכבה לינארית אינווריאנטית. 8
סעיף ה. טנזורים תלת-ממדיים. 9
שאלה 3 10
מסגרת משותפת לכל סעיפי שאלה 3. 11
סעיף א. רשת מבוססת קנוניזציה. 11
סעיף ב. רשת מבוססת סימטריזציה. 12
סעיף ג. סימטריה דגומה. 13
סעיף ד. שכבות לינאריות שקולות (DeepSets). 15
סעיף ה. הגדלת נתונים באמצעות תמורות. 16
שאלה 4 19
מסגרת משותפת לשאלה 4. 20
סעיף א. Canonization-based network. 21
סעיף ב. Full symmetrization network. 21
סעיף ג. Sampled-symmetrization network. 21
סעיף ד. Linear equivariant layers with mean pooling. 22
סעיף ה. Data augmentation with permutations. 22
Bonus. Transformer עם positional encoding. 22
Bonus. שימוש ב-surface normals. 23
Bonus. סימטריה לסיבובים באמצעות pairwise distances. 23
תוצאות, ניתוח והמלצה. 24

שאלה 1

Question 1 (Character inner products and Burnside's lemma (20 pts)). *Let $G \le S_n$ be any subgroup, acting on $X = \{1, 2, \dots, n\}$ in the natural way and on $\mathbb{R}^n$ by the permutation representation σ, i.e. $(\sigma(g)x)_i = x_{g^{-1}(i)}$. A linear map $W : \mathbb{R}^n \to \mathbb{R}^n$ (equivalently, an $n \times n$ matrix) is called G-**equivariant** if*

$$W\,\sigma(g)\,x = \sigma(g)\,W\,x \qquad \text{for all } g \in G,\ x \in \mathbb{R}^n.$$

Write $\mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n)$ for the vector space of all G-equivariant linear maps $\mathbb{R}^n \to \mathbb{R}^n$. Its dimension is exactly the number of free parameters of a G-equivariant linear layer $\mathbb{R}^n \to \mathbb{R}^n$. In Tutorial 12 we computed this number in two seemingly unrelated ways.

Method A (orbits / Burnside). *Let G act on $X^2 = X \times X$ diagonally, $g \cdot (i, j) = (g(i), g(j))$. The entries W_{ij} of a G-equivariant matrix are constant on the orbits of this action, so*

$$\dim \mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n) \;=\; |X^2/G| \quad (\text{the number of orbits}).$$

Method B (irreducibles / Schur). *Decomposing σ into irreducibles and applying Schur's lemma gives*

$$\dim \mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n) \;=\; \langle \chi_\sigma, \chi_\sigma \rangle,$$

where χ_σ is the character of the permutation representation.

Problem. *Prove that the two methods always agree: for every subgroup $G \le S_n$,*

$$|X^2/G| \;=\; \frac{1}{|G|} \sum_{g \in G} \big|\mathrm{Fix}_X(g)\big|^2 \;=\; \langle \chi_\sigma, \chi_\sigma \rangle.$$

Conclude that Methods A and B count the same number of G-equivariant layers $\mathbb{R}^n \to \mathbb{R}^n$, for every finite subgroup $G \le S_n$ and every $n \in \mathbb{N}$.

Hints.

(1) *For the first equality, apply Burnside's lemma (the orbit-counting theorem) to the diagonal action of G on X^2, and show that the number of pairs $(i, j) \in X^2$ fixed by g equals $|\mathrm{Fix}_X(g)|^2$.*

(2) *For the second equality, recall that the permutation character satisfies $\chi_\sigma(g) = |\mathrm{Fix}_X(g)|$, that χ_σ is real-valued, and that $\langle \chi_\sigma, \chi_\sigma \rangle = \frac{1}{|G|} \sum_g \chi_\sigma(g)^2$.*

נקבע תת-חבורה $G \le S_n$ הפועלת על $X = \{1, \dots, n\}$, ותהי $\sigma: G \to \mathrm{GL}_n(\mathbb{R})$ הצגת התמורה המתאימה, $\sigma(g)e_j = e_{g(j)}$. נסמן ב-$\mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n)$ את מרחב ההעתקות הלינאריות ה-G-שקילות. נוכיח כי

$$\dim\mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n) \;=\; |X^2/G| \;=\; \frac{1}{|G|} \sum_{g \in G} |\mathrm{Fix}_X(g)|^2 \;=\; \langle \chi_\sigma, \chi_\sigma \rangle.$$

המרחב $\mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n)$ ובסיס האינדיקטורים (שיטה א׳).

מאחר ש-$\sigma(g)$ מטריצת תמורה, תנאי השקילות

$$W\sigma(g) = \sigma(g)W \qquad \forall g \in G$$

שקול, איבר-איבר, לתנאי הבא (תרגול 10):

$$W_{g(i),\,g(j)} = W_{i,j} \qquad \forall g \in G,\ i, j \in X.$$

כלומר, W היא G-שקילה אם ורק אם הערך W_{ij} קבוע על מסלולי הפעולה האלכסונית $g \cdot (i, j) := \big(g(i), g(j)\big)$ על X^2.

יהי $X^2 = O_1 \sqcup \cdots \sqcup O_m$ הפירוק למסלולים, ולכל מסלול O_k נגדיר את מטריצת האינדיקטור שלו $B_k \in \mathbb{R}^{n\times n}$, כלומר $(B_k)_{ij} = 1$ אם $(i,j) \in O_k$ ו-0 אחרת (תרגול 10). המטריצות $B_1, \dots, B_m$ פורשות את $\mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n)$: כל W שקילה נכתבת $W = \sum_k w_k B_k$ כאשר w_k הוא הערך המשותף של W על המסלול O_k. כמו כן הן בלתי-תלויות לינארית, שכן למסלולים שונים תומכים זרים בזוגות. לכן זהו בסיס, ומתקבל:

$$\dim \mathrm{Hom}_G(\mathbb{R}^n, \mathbb{R}^n) = m = |X^2/G|.$$

כיוון ברנסייד.

נשתמש בלמת ברנסייד כפי שנוסחה בקורס (משפט 3.9 בהערות; תרגול 4): עבור חבורה סופית G הפועלת על קבוצה סופית Y,

$$|Y/G| = \frac{1}{|G|}\sum_{g\in G} |Y^g|, \qquad Y^g := \{\, y \in Y : g\cdot y = y \,\}.$$

נפעיל את הלמה על $Y = X^2$ עם הפעולה האלכסונית:

$$|X^2/G| = \frac{1}{|G|}\sum_{g\in G} |(X^2)^g|.$$

כעת, זוג (i,j) מקיים $g\cdot(i,j) = (i,j)$ אם ורק אם $g(i) = i$ וגם $g(j) = j$ בו-זמנית, כלומר אם ורק אם $i,j \in \mathrm{Fix}_X(g)$. לכן

$$(X^2)^g = \mathrm{Fix}_X(g) \times \mathrm{Fix}_X(g), \qquad |(X^2)^g| = |\mathrm{Fix}_X(g)|^2,$$

ובהצבה בשוויון הקודם מתקבל:

$$|X^2/G| = \frac{1}{|G|}\sum_{g\in G} |\mathrm{Fix}_X(g)|^2 . \qquad (1)$$

כיוון האופיינים.

לכל $g \in G$ המטריצה $\sigma(g)$ היא מטריצת תמורה, ולכן איבר האלכסון $\big(\sigma(g)\big)_{ii}$ שווה 1 אם $g(i) = i$ ו-0 אחרת. מכאן (תרגול 8, תרגיל 1):

$$\chi_\sigma(g) = \mathrm{tr}\big(\sigma(g)\big) = \sum_{i\in X} \big(\sigma(g)\big)_{ii} = |\mathrm{Fix}_X(g)|. \qquad (2)$$

בפרט χ_σ ממשי-ערכים, ולכן לפי הגדרת המכפלה הפנימית שבשאלה, ובעזרת (2) בשוויון האחרון:

$$\langle \chi_\sigma, \chi_\sigma \rangle = \frac{1}{|G|}\sum_{g\in G} \chi_\sigma(g)\,\overline{\chi_\sigma(g)} = \frac{1}{|G|}\sum_{g\in G} \chi_\sigma(g)^2 = \frac{1}{|G|}\sum_{g\in G} |\mathrm{Fix}_X(g)|^2 . \qquad (3)$$

מ-(1) ו-(3) מתקבל:

$$|X^2/G| = \frac{1}{|G|}\sum_{g\in G} |\mathrm{Fix}_X(g)|^2 = \langle \chi_\sigma, \chi_\sigma \rangle,$$

כנדרש.

מסקנה.
משיטה א׳ קיבלנו $\dim\mathrm{Hom}_G(\mathbb{R}^n,\mathbb{R}^n) = |X^2/G|$, ומשיטה ב׳ (המבוססת על משפט שור ופירוק σ להצגות אי-פריקות, כפי שניתן בשאלה) מתקבל $\dim\mathrm{Hom}_G(\mathbb{R}^n,\mathbb{R}^n) = \langle\chi_\sigma,\chi_\sigma\rangle$.
מאחר שהוכחנו ששני הביטויים שווים, שתי השיטות סופרות את אותו מספר של שכבות לינאריות G-שקילות $\mathbb{R}^n \to \mathbb{R}^n$, עבור כל תת-חבורה סופית $G \le S_n$ וכל $n \in \mathbb{N}$. מ.ש.ל. ■

שאלה 2

Question 2 (Netflix rating matrices and group equivariance (20 pts)). *A streaming service stores its data as a rating matrix $A \in \mathbb{R}^{m\times n}$, where A_{ij} is the rating that user $i \in \{1,\dots,m\}$ gives to movie $j \in \{1,\dots,n\}$. The labels of the users and of the movies carry no intrinsic order: relabeling the users (permuting the rows) and relabeling the movies (permuting the columns) describe the same underlying data.*

Throughout this question, let $G = S_m \times S_n$ act on $\mathbb{R}^{m\times n}$ by

$$\big((\pi,\tau)\cdot A\big)_{ij} \;=\; A_{\pi^{-1}(i),\,\tau^{-1}(j)}, \qquad \pi \in S_m,\ \tau \in S_n,$$

*so that π permutes the rows and τ permutes the columns independently. A linear map $L : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ is G-**equivariant** if $L\big((\pi,\tau)\cdot A\big) = (\pi,\tau)\cdot L(A)$ for all $(\pi,\tau) \in G$ and all A. A linear map $\ell : \mathbb{R}^{m\times n} \to \mathbb{R}$ is G-**invariant** if $\ell\big((\pi,\tau)\cdot A\big) = \ell(A)$ for all (π,τ) and all A.*

Answer the following. Justify every answer with a proof or a clear derivation; for the parameter counts you may use the orbit-counting / character argument from Question 1 and Tutorial 12.

(a) ***Symmetry group.*** *Explain why $G = S_m \times S_n$, acting as above, is the natural symmetry group of the data, and describe its action on a rating matrix in words.*

(b) ***Equivariant linear layer.*** *Determine the general form of a G-equivariant affine map $L(A) = L_0(A) + B$, where L_0 is linear and $B \in \mathbb{R}^{m\times n}$ is a bias. Write $L(A)_{ij}$ explicitly as a formula involving the entry A_{ij}, the row sum $\sum_l A_{il}$, the column sum $\sum_k A_{kj}$, and the total sum $\sum_{k,l} A_{kl}$. State how many free parameters the linear part L_0 has and how many the bias B has.*

(c) ***Parameter-sharing diagram.*** *View the linear part L_0 as an $(mn)\times(mn)$ matrix acting on the vectorization of A. Draw its parameter-sharing pattern: indicate which entries of this matrix are tied to the same free parameter. Equivalently, fix a single output entry (i,j) and shade, on a copy of the $m \times n$ input grid, which input entries (k,l) are multiplied by each parameter from part (b). The case $m = n = 3$ suffices to show the pattern.*

(d) ***Invariant linear layer.*** *Determine the general form of a G-invariant linear map $\ell : \mathbb{R}^{m\times n} \to \mathbb{R}$, and state how many free parameters it has.*

(e) ***Three-way tensors.*** *Now suppose the data is a 3-tensor $A \in \mathbb{R}^{m\times n\times k}$ (for example, users $\times$ movies $\times$ time), with symmetry group $S_m \times S_n \times S_k$ acting by permuting the three index sets independently. How many free parameters does a general equivariant linear map $\mathbb{R}^{m\times n\times k} \to \mathbb{R}^{m\times n\times k}$ have? Describe (and sketch) the resulting parameter-sharing scheme. **Hint:** count the orbits of the action on pairs of index triples; the three coordinates contribute independently, so compare your count with the $\mathbb{R}^n \to \mathbb{R}^n$ and $\mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ cases.*

נתונה מטריצת דירוגים $A \in \mathbb{R}^{m\times n}$, שבה A_{ij} הוא הדירוג שנתן משתמש $i \in \{1,\dots,m\}$ לסרט $j \in \{1,\dots,n\}$. החבורה $G = S_m \times S_n$ פועלת על $\mathbb{R}^{m\times n}$ על-ידי

$$((\pi,\tau)\cdot A)_{ij} = A_{\pi^{-1}(i),\tau^{-1}(j)}, \qquad \pi \in S_m,\ \tau \in S_n,$$

כלומר π מסדרת מחדש את השורות ו-τ את העמודות, באופן בלתי-תלוי.

סעיף א. חבורת הסימטריה.

מספר המשתמשים ומספר הסרטים הם תוויות שרירותיות נטולות תוכן: שום דבר בנתונים אינו מבדיל את "משתמש 3" מ"משתמש 7" מלבד הדירוגים עצמם. לכן כל מספור מחדש של המשתמשים, יחד עם כל מספור מחדש של הסרטים, מניב מטריצה המתארת בדיוק את אותה מציאות, וחבורת הסימטריה חייבת להכיל את כל הזוגות (π,τ). שתי פעולות המספור הן בחירות בלתי-תלויות — משתמשים וסרטים הם עצמים מטיפוסים שונים, ואין דבר הקושר את π ל-τ — ולכן החבורה היא המכפלה הישרה $S_m \times S_n$, הפועלת רכיב-רכיב. זאת בניגוד לפעולה האלכסונית על X^2 שבשאלה 1, שבה אותו איבר g פעל על שני הרכיבים בו-זמנית (המצב המתאים, למשל, למטריצת שכנויות של גרף — שם שורה i ועמודה i מייצגות את אותו קודקוד). מנגד, אין להרחיב את החבורה ל-S_{mn} המערבבת תאים באופן כללי: ערבוב כזה מעביר דירוגים בין משתמשים שונים, הורס את המבנה של "אילו דירוגים שייכים לאותו משתמש", ולכן משנה את משמעות הנתונים — אין זו סימטריה של הבעיה.

תיאור הפעולה במילים: $(\pi,\tau)\cdot A$ היא המטריצה A שבה השורות סודרו מחדש לפי π והעמודות לפי τ — כל משתמש נושא עמו את שורת הדירוגים המלאה שלו למקומו החדש, וכל סרט נושא עמו את עמודתו. האיבר בשורה i ובעמודה j של המטריצה החדשה הוא הדירוג שנתן המשתמש $\pi^{-1}(i)$ לסרט $\tau^{-1}(j)$: במשבצת החדשה i יושב המשתמש ש-π שולחת אל i. ההופכיים הם שמבטיחים כי זו פעולה שמאלית:

$$((\pi_1,\tau_1)\cdot((\pi_2,\tau_2)\cdot A))_{ij} = A_{\pi_2^{-1}\pi_1^{-1}(i),\ \tau_2^{-1}\tau_1^{-1}(j)} = ((\pi_1\pi_2,\ \tau_1\tau_2)\cdot A)_{ij}.$$

סעיף ב. שכבה לינארית שקילה (אפינית).

נכתוב $L(A) = L_0(A) + B$, כאשר L_0 לינארית ו-$B \in \mathbb{R}^{m\times n}$ הטיה.

נטפל תחילה ב-L_0. קבוצת האינדקסים ("התאים") של המטריצה היא $P := \{1,\dots,m\} \times \{1,\dots,n\}$, והחבורה G פועלת עליה כמו בסעיף א. בהצגת L_0 כמטריצה בגודל $(mn)\times(mn)$ הפועלת על הווקטוריזציה של A, החלה איבר-איבר של תנאי השקילות — בדיוק כבשיטה א׳ של שאלה 1 (תרגול 10) — מראה כי L_0 שקילה אם ורק אם המטריצה המייצגת קבועה על מסלולי הפעולה האלכסונית של G על זוגות תאים:

$$(\pi,\tau)\cdot((i,j),(k,l)) = \big((\pi(i),\tau(j)),\ (\pi(k),\tau(l))\big).$$

מספר הפרמטרים החופשיים של L_0 הוא אפוא מספר המסלולים. מאחר ש-π ו-τ פועלות על שני צירי האינדקסים באופן בלתי-תלוי, הנתונים היחידים של זוג תאים הנשמרים תחת הפעולה הם שני המדדים הבינאריים $[i=k]$ ו-$[j=l]$, והם קובעים את המסלול במלואו: עבור $m,n \geq 2$ כל ארבעת הצירופים מופיעים, והמעבר בין שני זוגות בעלי אותו דפוס נובע מכך ש-S_m ו-S_n יכולות להעביר כל צירוף מתאים לכל צירוף מתאים אחר. לכן יש בדיוק $2\times 2 = 4$ מסלולים: אותו תא; אותה שורה ועמודה שונה; אותה עמודה ושורה שונה; שורה שונה ועמודה שונה.

בסיס מטריצות האינדיקטור של ארבעת המסלולים (כבשאלה 1) נותן את הצורה הכללית, בסכומים "מנוקבים":

$$L_0(A)_{ij} = \alpha A_{ij} + \beta \sum_{l\neq j} A_{il} + \gamma \sum_{k\neq i} A_{kj} + \delta \sum_{k\neq i,\, l\neq j} A_{kl},$$

ובהחלפת בסיס הפיכה (משולשת: כל סכום מלא שווה לסכום המנוקב המתאים בתוספת איברים מהמסלולים הקודמים. נציין כי המעבר הוא מותר כי אין דרישה לכך שהבסיס יהיה אורתוגונלי- מספיק לנו שהבסיס יהיה מורכב מ4 וקטורים בלתי תלויים) מתקבלת הצורה המבוקשת בשאלה:

$$L_0(A)_{ij} = a\,A_{ij} + b\sum_{l=1}^{n} A_{il} + c\sum_{k=1}^{m} A_{kj} + d\sum_{k=1}^{m}\sum_{l=1}^{n} A_{kl}.$$

כעת ההטיה. הצבת $L(A) = L_0(A) + B$ בתנאי השקילות $L\big((\pi,\tau)\cdot A\big) = (\pi,\tau)\cdot L(A)$ וצמצום השקילה נותנות L_0

$$B = (\pi,\tau)\cdot B \qquad \forall(\pi,\tau)\in G,$$

כלומר $B_{ij} = B_{\pi^{-1}(i),\,\tau^{-1}(j)}$ לכל π,τ ולכל תא (i,j). הפעולה של G על התאים טרנזיטיבית — לכל שני תאים קיים (π,τ) המעביר זה אל זה — ולכן כל איברי B שווים: $B = e\,J$, כאשר J מטריצת האחדות (שכל איבריה 1) ו-$e\in\mathbb{R}$ סקלר. להטיה פרמטר חופשי יחיד.

סך הכול, הצורה הכללית של שכבה אפינית G-שקילה היא

$$L(A)_{ij} = a\,A_{ij} + b\sum_{l} A_{il} + c\sum_{k} A_{kj} + d\sum_{k,l} A_{kl} + e,$$

כאשר $a,b,c,d,e\in\mathbb{R}$ סקלרים חופשיים: לחלק הלינארי L_0 יש 4 פרמטרים חופשיים, ולהטיה B פרמטר חופשי אחד (עבור $m,n\geq 2$). מ.ש.ל. ∎

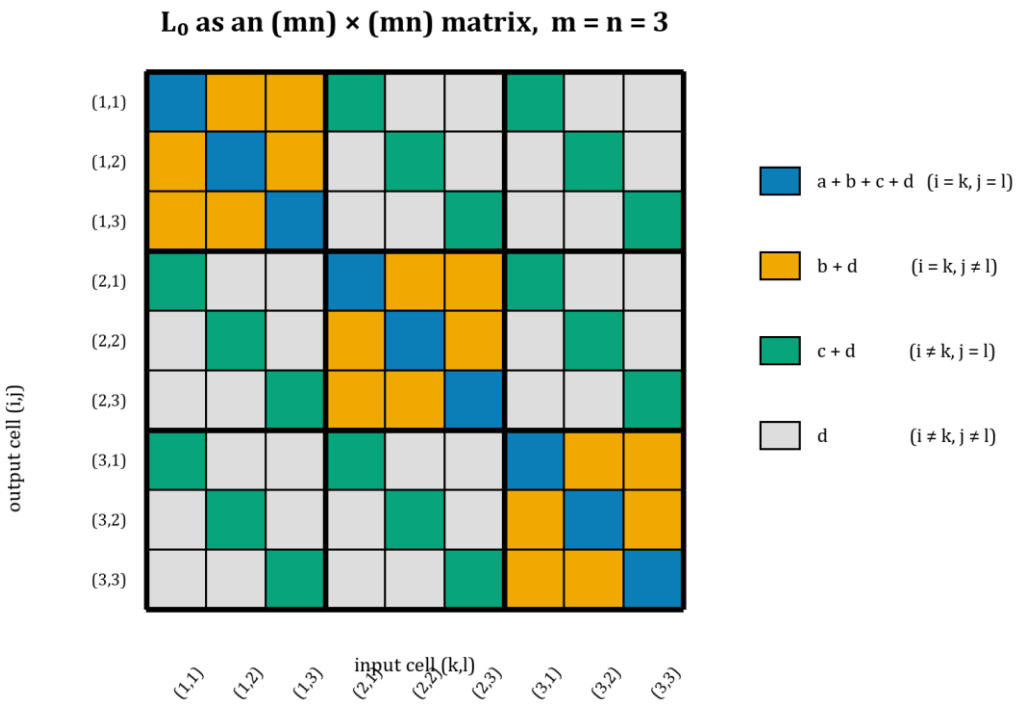
סעיף ג. דיאגרמת שיתוף הפרמטרים.

נסדר את התאים בסדר-שורות (וקטוריזציה): התא (i,j) ממופה לאינדקס $n(i-1)+j$. בבסיס זה L_0 היא מטריצה בגודל $(mn)\times(mn)$, שאיבריה — קריאה ישירה של הנוסחה מסעיף ב — הם

$$(L_0)_{(i,j),(k,l)} = a\,[i=k]\,[j=l] + b\,[i=k] + c\,[j=l] + d,$$

כאשר $[\cdot]$ מציין אינדיקטור. סידור זה חושף מבנה של $m\times m$ בלוקים בגודל $n\times n$: הבלוק (i,k) מכיל את המקדמים המוליכים משורת הקלט k אל שורת הפלט i, והמיקום בתוך הבלוק הוא (j,l). ארבעת מסלולי סעיף ב הופכים אפוא לארבע מחלקות-קשירה גאומטריות: בבלוקים שעל אלכסון-הבלוקים $(i=k)$ האלכסון הפנימי $(j=l)$ שווה $a+b+c+d$ ושאר הבלוק שווה $b+d$; בבלוקים שמחוץ לאלכסון $(i\neq k)$ האלכסון הפנימי שווה $c+d$ ושאר הבלוק שווה d. כל האיברים בתוך מחלקה אחת קשורים לאותו פרמטר חופשי — בבסיס המנוקב של סעיף ב אלו בדיוק $\alpha,\beta,\gamma,\delta$, פרמטר אחד לכל מחלקה — והסכומים לעיל משקפים את החפיפה בין ארבעת האיברים בפרמטריזציה בסכומים המלאים.

איור 1 מציג את התבנית עבור $m=n=3$, כנדרש בשאלה: המטריצה 9×9 צבועה לפי ארבע מחלקות-הקשירה, וקווים מעובים תוחמים את הבלוקים. כל צבע מייצג פרמטר משותף אחד; התבנית זהה לכל $m,n\geq 2$.



איור 1: דיאגרמת שיתוף הפרמטרים של L_0 עבור $m = n = 3$ — המטריצה המלאה בגודל $(mn) \times (mn)$.

סעיף ד. שכבה לינארית אינווריאנטית.

כל פונקציונל לינארי על מרחב המטריצות ניתן לכתיבה בצורה

$$\ell(A) = \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}A_{ij}$$

כמו באפיון באמצעות מסלולים בתרגול 10, בסעיף העוסק בהעתקות לינאריות אינווריאנטיות ובהטיה, תנאי האינווריאנטיות מחייב את מערך המקדמים להיות קבוע על מסלולי הפעולה של החבורה על קבוצת התאים.

פעולה זו טרנזיטיבית: לכל שני תאים קיימות תמורת שורות ותמורת עמודות המקיימות

$$\pi(i) = k, \quad \tau(j) = l$$

לכן כל התאים שייכים למסלול יחיד, ומכאן שכל המקדמים שווים לאותו סקלר:

$$c_{ij} = a \quad \forall i,j$$

לפיכך הצורה הכללית היא

$$\ell(A) = a\sum_{i=1}^{m}\sum_{j=1}^{n} A_{ij}\,, \quad a \in \mathbb{R}$$

אכן, לכל זוג תמורות מתקיים

$$\ell\big((\pi,\tau)\cdot A\big) = a\sum_{i=1}^{m}\sum_{j=1}^{n} A_{\pi^{-1}(i),\tau^{-1}(j)}$$

$$= a\sum_{k=1}^{m}\sum_{l=1}^{n} A_{kl} = \ell(A)$$

בשוויון השני החלפנו את משתני הסכימה באמצעות ההופכיים של התמורות. ההחלפה חוקית משום ששתי התמורות הן חד-חד-ערכיות ועל; לכן כל אינדקס שורה וכל אינדקס עמודה מופיע בדיוק פעם אחת.

לכן למרחב ההעתקות הלינאריות האינווריאנטיות יש פרמטר חופשי יחיד. מ.ש.ל. ■

סעיף ה. טנזורים תלת-ממדיים.

בהתאם לשיטת ספירת המסלולים מתרגול 12, מקדמי ההעתקה הלינארית מאונדקסים על ידי זוג של שלשות אינדקסים. המסלול של זוג כזה נקבע רק לפי השאלה האם אינדקס הפלט שווה לאינדקס הקלט בכל אחד משלושת הצירים:

$$\varepsilon_1 = 1(i = i'), \quad \varepsilon_2 = 1(j = j'), \quad \varepsilon_3 = 1(t = t').$$

לכל אחד משלושת התנאים שתי אפשרויות, והפעולות בשלושת הצירים בלתי-תלויות. לכן, בהנחה שכל אחד מן הממדים הוא לפחות 2, מספר המסלולים הוא

$$2^3 = 8.$$

אכן, שני זוגות של שלשות בעלי אותו דפוס שוויון ניתנים להעברה זה לזה באמצעות בחירה בלתי-תלויה של תמורה בכל ציר; לעומת זאת, תמורה אינה יכולה להפוך שוויון לאי-שוויון. לכן אלו בדיוק שמונת המסלולים, וכל מסלול קובע פרמטר משותף אחד. מכאן שלהעתקה לינארית שקילה כללית יש 8 פרמטרים חופשיים.

אם מפרשים את המונח "שכבה" כשכבה אפינית, נכתוב

$$L(A) = L_0(A) + B, \quad B_{\mathrm{ijt}} = b.$$

הפעולה על שלשות האינדקסים טרנזיטיבית, ולכן ההטיה חייבת להיות קבועה בכל התאים ומוסיפה פרמטר חופשי יחיד. לפיכך בניסוח הלינארי שבשאלה יש 8 פרמטרים, ואילו לשכבה אפינית יש 9 פרמטרים בסך הכול. מ.ש.ל. ∎

שאלה 3

Question 3 (Implementing set networks in PyTorch (25 pts)). *Consider set data with S_n symmetry. A set of n elements, each with d feature dimensions, is represented as a matrix $X \in \mathbb{R}^{n\times d}$ whose i-th row $X_i \in \mathbb{R}^d$ is the feature vector of the i-th element. A permutation $\pi \in S_n$ acts by permuting the rows, $(\pi \cdot X)_i = X_{\pi^{-1}(i)}$. Recall the two target behaviours:*

- *a network $F : \mathbb{R}^{n\times d} \to \mathbb{R}^p$ is **invariant** if $F(\pi \cdot X) = F(X)$ for all $\pi \in S_n$;*
- *a layer $f : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d'}$ is **equivariant** if $f(\pi \cdot X) = \pi \cdot f(X)$ for all $\pi \in S_n$.*

*You will implement the following architectures in PyTorch. For **each** one, supply (i) a `torch.nn.Module` class implementing it, and (ii) a **test function**. Your implementations must accept inputs of shape (n, d) with $d \geq 1$ arbitrary.*

Specification of the test functions. *First decide, for each architecture, whether it is invariant or equivariant; this determines which property the test checks. For parts (a)–(d) the test does **not** train the network: with randomly initialized parameters it must*

1. *generate a random input $X \in \mathbb{R}^{n\times d}$;*
2. *draw a random permutation $\pi \in S_n$ and form $\pi \cdot X$;*
3. *compare the network's outputs on X and on $\pi \cdot X$ (for an invariant net: $F(\pi \cdot X) \approx F(X)$; for an equivariant layer: $f(\pi \cdot X) \approx \pi \cdot f(X)$);*
4. *return a boolean: `True` iff the relevant equality holds up to a numerical tolerance (e.g. `atol=1e-5`) that you state.*

Architectures. *Throughout, "your favourite model" means any standard network you choose, e.g. an MLP applied to the flattened input; state which one you used.*

(a) ***Canonization-based network.*** *Apply a **canonization** function $c : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d}$ that reorders the rows by a fixed, deterministic rule (for instance, sort the rows in lexicographic order, with a deterministic tie-break), so that $c(\pi \cdot X) = c(X)$ for every π; then apply your favourite model to $c(X)$. State your canonization rule and why it is permutation-invariant.*

(b) ***Symmetrization network.*** *Symmetrize your favourite model h by averaging over the **whole** group: $F(X) = \frac{1}{n!}\sum_{\pi\in S_n} h(\pi \cdot X)$. (Because this enumerates all $n!$ permutations, run it only for small n.)*

(c) ***Sampled symmetrization.*** *Same as (b), but average over a random subset $P \subseteq S_n$ of B permutations: $F(X) = \frac{1}{|P|}\sum_{\pi\in P} h(\pi \cdot X)$. State B, and note that this network is only **approximately** invariant; choose the tolerance in your test accordingly and report it.*

(d) ***Linear equivariant layers.*** *Build a network by stacking S_n-equivariant linear layers (DeepSets layers) with pointwise nonlinearities. Recall the equivariant linear layer $f(X)_i = X_i W_1 + \left(\frac{1}{n}\sum_j X_j\right) W_2 + b$, with learnable $W_1, W_2 \in \mathbb{R}^{d\times d'}$ and $b \in \mathbb{R}^{d'}$. Test the equivariance of a single layer; if you also build an invariant network, pool over the set (e.g. sum or mean over the n rows) at the end and test invariance of the whole network.*

(e) ***Data augmentation.*** *Implement a standard (not architecturally invariant) model on the flattened input, and train it with data augmentation (apply a fresh random permutation to each example each time it is seen). The **regression target** is the coordinate-wise variance of the set,*

$$g(X) \in \mathbb{R}^d, \qquad g(X)_c \;=\; \frac{1}{n}\sum_{i=1}^{n}\big(X_{ic} - \bar{X}_c\big)^2, \quad \bar{X}_c = \frac{1}{n}\sum_{i=1}^{n} X_{ic},$$

which is permutation-invariant. Devise a full training procedure (loss, optimizer, data generation), train the model, and then report (1) the approximation error to g on held-out random

sets and (2) an ***invariance error****, e.g.* $\mathbb{E}_{X,\pi}\|F(\pi \cdot X) - F(X)\|$*, measured before and after training. Discuss whether augmentation makes the network approximately invariant.*

In your PDF report, include for each part a short description, the outcome of the test function (pass/fail and tolerance), and for (e) the two error curves or numbers.

מסגרת משותפת לכל סעיפי שאלה 3.

כל המחלקות מומשו ב-PyTorch כך שממדי הקלט והפלט נקבעים בעת יצירת המודל. הטבלה מתארת את הבחירות המשותפות לניסויים המדווחים; בכל סעיף נפרט רק את הבחירות הייחודיות לאותה ארכיטקטורה.

בחירה משותפת	ערך
ספריית המימוש	PyTorch
התקן חישוב	CUDA
GPU	NVIDIA GeForce RTX 3080
seed	2319
דיוק נומרי	float32
מספר תכונות לקלט	d=3
פונקציית הפעלה	ReLU
טווח תקציב הפרמטרים	53,571-55,976

בחרנו את הממדים ואת תקציב הפרמטרים מתוך מחשבה על ההשוואה בשאלה 4. לכן במודלים הסקלביליים השתמשנו ב-$n = 256, d = 3, p = 40$, המתאימים לענן הנקודות של שאלה 4. בסעיפים ב-ג נדרש לעבור על תמורות רבות ולכן השתמשנו ב-$n = 7$; בסעיף ה ממד הפלט הוא שלוש משום שהמטרה היא שונות של שלוש קואורדינטות.

תקציב של כ-54-56 אלף פרמטרים מאפשר השוואה הוגנת בין הארכיטקטורות בלי להעמיס שלא לצורך על ה-RTX 3080. ה-seed הקבוע, אותו דיוק נומרי ואותה סביבת CUDA מבטיחים שחזרות הניסוי יהיו ניתנות לשחזור; הבדלים נוספים מופיעים רק במקום שבו מבנה הסעיף מחייב אותם.

סעיף א. רשת מבוססת קנוניזציה.

בהתאם לדוגמת הקנוניזציה מתרגול 12, נגדיר את $c(X)$ כמיון לקסיקוגרפי עולה של שורות הקלט. תמורה משנה רק את סדר השורות ולא את קבוצת הערכים, ולכן לכל $\pi \in S_n$ מתקיים

$$c(\pi \cdot X) = c(X).$$

אם שתי שורות זהות, החלפתן אינה משנה את המטריצה ולכן אין צורך בכלל שבירת שוויון נוסף. לאחר המיון משטחים את המטריצה ומפעילים MLP רגיל h. מכאן ש-$F = h \circ c$ אינווריאנטית לכל ערכי המשקלים.

מימוש והיפרפרמטרים.

היפרפרמטר או בחירת מימוש	ערך
n	256
p	40
כלל הקנוניזציה	lexicographic sort עולה של השורות
ארכיטקטורת ה-MLP	768 → 64 → 64 → 40
מספר פרמטרים	55,976
atol	1e-5
rtol	0

הגודל $n = 256$ נבחר משום שהקנוניזציה סקלבילית ומתאימה ישירות לממדי שאלה 4. המיון מומש באמצעות stable argsort מן הקואורדינטה האחרונה לראשונה, וכך מתקבל סדר לקסיקוגרפי דטרמיניסטי בלי לעבור על תמורות.

תוצאות.

מדד או בדיקה	תוצאה
ערך ההחזרה של פונקציית הבדיקה	True
שגיאה מוחלטת מרבית	0
בדיקה עם שורות כפולות	PASS

דיון בתוצאות.

השגיאה התקבלה כאפס בדיוק משום שלאחר הקנוניזציה שני הקלטים המושווים הם אותו טנזור, ולא רק טנזורים קרובים נומרית. גם בדיקת השורות הכפולות עברה, ולכן המימוש מכסה את מקרה השוויון בלי לפגוע בדטרמיניזם. הבדיקה מאשרת את המימוש; האינווריאנטיות עצמה נובעת מן השוויון של פונקציית הקנוניזציה שהוכח לעיל.

סעיף ב. רשת מבוססת סימטריזציה.

לפי נוסחת מיצוע החבורה מתרגול 12, מפעילים MLP תלוי-סדר על כל תמורות הקלט וממצעים את הפלטים:

$$F(X) = \frac{1}{7!} \sum_{\pi \in S_7} h(\pi \cdot X).$$

לכל $\sigma \in S_7$ ההעתקה $\pi \mapsto \rho = \pi\sigma$ היא בייקציה של החבורה לעצמה. לכן ניתן לשנות את אינדקס הסכימה ולקבל

$$F(\sigma \cdot X) = \frac{1}{7!} \sum_{\rho \in S_7} h(\rho \cdot X) = F(X).$$

לפיכך המיצוע המלא אינווריאנטי גם כאשר ה-MLP הבסיסי עצמו אינו אינווריאנטי.

מימוש והיפרפרמטרים.

היפרפרמטר או בחירת מימוש	ערך
n	7
p	40
ארכיטקטורת ה-MLP הבסיסי	21 → 207 → 207 → 40
מספר פרמטרים	55,930
מספר התמורות במיצוע	7! = 5,040
אופן עיבוד התמורות	CUDA batches of 1,024 permutations
atol	1e-5
rtol	0

בחרנו $n = 7$ כדי שחישוב כל $7! = 5040$ התמורות יהיה מעשי. רוחבי ה-MLP נבחרו כך שמספר הפרמטרים יישאר קרוב לתקציב המשותף, וההערכות חולקו ל-CUDA batches רק מטעמי זיכרון; החלוקה אינה משנה את הממוצע המתמטי.

תוצאות.

מדד או בדיקה	תוצאה
שגיאת האינוריאנטיות של ה-MLP הבסיסי	0.206946
שגיאת האינוריאנטיות לאחר מיצוע מלא	1.49e-8
בדיקה שכל 5,040 התמורות מופיעות פעם אחת	PASS
בדיקת גרדיאנטים סופיים	PASS
ערך ההחזרה של פונקציית הבדיקה	True

דיון בתוצאות.

השגיאה 0.206946 של ה-MLP הבסיסי מוודאת שהבדיקה אינה עוברת במקרה בגלל מודל שכבר אינוריאנטי. לאחר המיצוע השגיאה ירדה לסדר גודל של 10^{-8}, מתחת לסבילות שנבחרה; הסטייה הזעירה מאפס היא שגיאת עיגול של float32. בדיקת מבנה החבורה מוודאת שכל תמורה נכללה פעם אחת, ובדיקת הגרדיאנטים מראה שהמימוש נשאר ניתן לאימון.

סעיף ג. סימטריה דגומה.

דגמנו פעם אחת תת-קבוצה אחידה ללא החזרה $P \subset S_7$ והשתמשנו באותה תת-קבוצה בשני אגפי ההשוואה:

$$F_P(X) = \frac{1}{B}\sum_{\pi \in P} h(\pi \cdot X).$$

בניגוד לחבורה המלאה, בדרך כלל $P\sigma \neq P$. לכן שינוי האינדקס מסעיף ב מחליף את קבוצת הסכימה, והמודל רק בקירוב אינוריאנטי.

כדי לבחור את B השתמשנו בשונות של ממוצע מדגם ללא החזרה. עבור קואורדינטת פלט j נסמן

$$\Delta_{\pi,j} = h(\pi\sigma \cdot X)_j - h(\pi \cdot X)_j.$$

אם $s_{\Delta,j}^2$ היא שונות האוכלוסייה על כל $N = 5040$ התמורות, אז

$$\mathrm{Var}(F_P(\sigma \cdot X) - F_P(X))_j = \frac{N-B}{B(N-1)} s_{\Delta,j}^2.$$

מדד ההכרעה הוא average RMS על דוגמאות הבדיקה וקואורדינטות הפלט:

$$E_{\mathrm{RMS}} = \sqrt{\frac{1}{Mp} \sum_{r=1}^{M} \sum_{j=1}^{p} \left| F_P(\sigma_r \cdot X_r)_j - F_P(X_r)_j \right|^2}.$$

מימוש והיפרפרמטרים.

היפרפרמטר או בחירת מימוש	ערך
n	7
p	40
ארכיטקטורת ה-MLP הבסיסי	21 → 207 → 207 → 40
מספר פרמטרים	55,930
גודל החבורה N	5,040
גודל תת-הקבוצה B	252
שיעור הדגימה	5%
מספר דוגמאות הבדיקה M	100
מדד ההכרעה	average RMS
atol	1e-2

הבחירה ב-5% מן החבורה נותנת הפחתה של פי 20 בחישוב ועדיין מובילה, לפי נוסחת מונטה-קרלו ומקדם האוכלוסייה שנמדד עבור ה-MLP, לתחזית שגיאה 0.005811 הנמוכה מן הסבילות 0.01. אותה תת-קבוצה נשמרת בשני החישובים כדי שהמדידה תשקף רק את השפעת שינוי הקלט ולא רעש משתי דגימות שונות.

תוצאות.

מדד או בדיקה	תוצאה
תחזית מונטה-קרלו	0.005811
שגיאה אמפירית לפי המדד	0.005727
פער יחסי בין התחזית למדידה	1.44%
אחוזון 95 של שגיאת ה-RMS	0.008496
הפרש מוחלט מרבי, לאבחון בלבד	0.026787
תוצאת הבדיקה לפי המדד שנבחר	PASS

דיון בתוצאות.

השגיאה האמפירית קרובה לתחזית התיאורטית בפער יחסי של 1.44%, ולכן מודל מונטה-קרלו מתאר היטב את סדר הגודל של הטעות. גם אחוזון 95 נמוך מן הסבילות. ההפרש המוחלט המרבי גבוה מ-0.01, אך הוא מדווח כאבחון בלבד ואינו סותר את הבדיקה, משום שהקריטריון שנבחר מראש הוא average RMS ולא שגיאת המקרה הגרוע. התוצאה מאשרת אינווריאנטיות מקורבת בלבד, כצפוי מתת-קבוצה שאינה סגורה לפעולת החבורה.

סעיף ד. שכבות לינאריות שקולות (DeepSets).

בהתאם לצורת השכבה השקולה מן ההרצאות בשבועות 11-12 (שקופיות 45-46), מימשנו

$$L(X)_i = X_i W_1 + \frac{1}{n}\sum_j X_j W_2 + b.$$

לכל תמורה π ממוצע השורות נשמר, ואיבר השורה המקומי באינדקס i נעשה $X_{\pi^{-1}(i)}$. לכן

$$L(\pi \cdot X)_i = L(X)_{\pi^{-1}(i)} = (\pi \cdot L(X))_i.$$

ReLU פועלת נקודתית ולכן מתחלפת עם התמורה; הרכבת שכבות כאלה נשארת שקולה. לבסוף מפעילים mean pooling על ייצוגי השורות, וכיוון שממוצע אינו תלוי בסדר מתקבלת רשת אינווריאנטית:

$$F(X) = \frac{1}{n}\sum_i H(X)_i, F(\pi \cdot X) = F(X).$$

מימוש והיפרפרמטרים.

היפרפרמטר או בחירת מימוש	ערך
n	256
p	40
רוחבי הבלוק השקול	3 → 128 → 128
אופן הפעלת אי-הלינאריות	pointwise ReLU
איגום	mean pooling
ראש הסיווג	128 → 128 → 40
מספר פרמטרים	55,464
atol	1e-5
rtol	0

כאן ניתן להשתמש ישירות ב-$n = 256$ משום שהחישוב לינארי במספר הנקודות ואינו מונה תמורות. רוחב 128 נבחר כך שמספר הפרמטרים יהיה קרוב לתקציב המשותף ולמודלים שישמשו בשאלה 4. ReLU מופעלת בנפרד על כל שורה כדי לא לשבור שקילות.

תוצאות.

מדד או בדיקה	תוצאה
שגיאה מרבית בשכבה שקולה יחידה	2.38e-7
שגיאה מרבית בבלוק השקול המלא	1.79e-7
שגיאה מרבית לאחר mean pooling	1.68e-8
בדיקת גרדיאנטים סופיים	PASS
ערך ההחזרה של פונקציית הבדיקה	True

דיון בתוצאות.

כל שלוש השגיאות קטנות משמעותית מן הסבילות 1e-5. השגיאות הזעירות בשכבה ובבלוק נובעות מסדר פעולות שונה ב-float32, ואילו mean pooling מקטין עוד את ההפרש. בדיקת הגרדיאנטים מאשרת שהשכבות אינן רק שקולות מבחינה אלגברית אלא גם מתאימות לאימון רגיל ב-PyTorch.

סעיף ה. הגדלת נתונים באמצעות תמורות.

כדי לבדוק אם הגדלת נתונים לבדה מעודדת אינווריאנטיות, בחרנו MLP רגיל על הקלט המשוטח; לכן האינווריאנטיות אינה מובטחת על ידי הארכיטקטורה. לכל דוגמה r דגמנו תוחלת ושונות חדשות ולאחר מכן את שורות הקלט לפי

$$\begin{aligned} \mu^{(r)} &\sim \mathrm{N}(0, I_3), \\ v_c^{(r)} &\sim \text{Rayleigh}(1), \\ X_i^{(r)} &\sim \mathrm{N}(\mu^{(r)}, \text{diag}(v^{(r)})). \end{aligned}$$

המטרה היא השונות האמפירית בכל אחת משלוש הקואורדינטות, עם מחלק n (unbiased=False). דגימת פרמטרים חדשים לכל דוגמה מונעת מן הרשת להסתפק בתחזית קבועה של התפלגות יחידה.

במודל המוגדל הוחלה תמורה אחידה חדשה בכל פעם שדוגמה הוצגה לאימון. מודל הביקורת קיבל אותם משקלים התחלתיים, אותה חלוקת נתונים ואותו batch order, אך ללא תמורות. כך ההבדל היחיד בין שני המודלים בכל זוג היה הגדלת הנתונים.

מימוש והיפרפרמטרים.

היפרפרמטר	ערך
מספר דוגמאות	1,000
חלוקת הנתונים	train/validation/test = 700/150/150
n בניסוי הראשי	256
n בניסוי ההשוואה	7
p	3
ארכיטקטורה עבור n=256	768 → 64 → 64 → 3
ארכיטקטורה עבור n=7	21 → 349 → 130 → 3
מספר פרמטרים בכל מודל	53,571
פונקציית הפסד	MSELoss
אופטימייזר	Adam
learning rate	1e-4
batch size	64
מספר epochs מרבי	2,000
patience	100
כלל הגדלת הנתונים	תמורה אחידה חדשה בכל הצגה לאימון
קשר בין ניסויי n	עבור n=7 נלקחו שבע השורות הראשונות מכל דוגמת n=256

התפלגות Rayleigh מבטיחה שונויות חיוביות, בעוד שהתוחלות הגאוסיות יוצרות מגוון מיקומים. החלוקה 70/15/15 משאירה קבוצות validation ו-test נפרדות; MSELoss מתאים לבעיית רגרסיה, ו-Adam עם learning rate 1e-4 הוא בחירה יציבה לרשת הצפופה. early stopping מגביל התאמת-יתר ומשחזר את המודל בעל validation MSE הנמוך ביותר.

הניסוי הראשי משתמש ב-$n = 256$. כדי לבחון את ההשערה שמרחב התמורות הוא גורם מרכזי, הוספנו זוג ניסויים עם $n = 7$. לקחנו את שבע השורות הראשונות מאותן דוגמאות

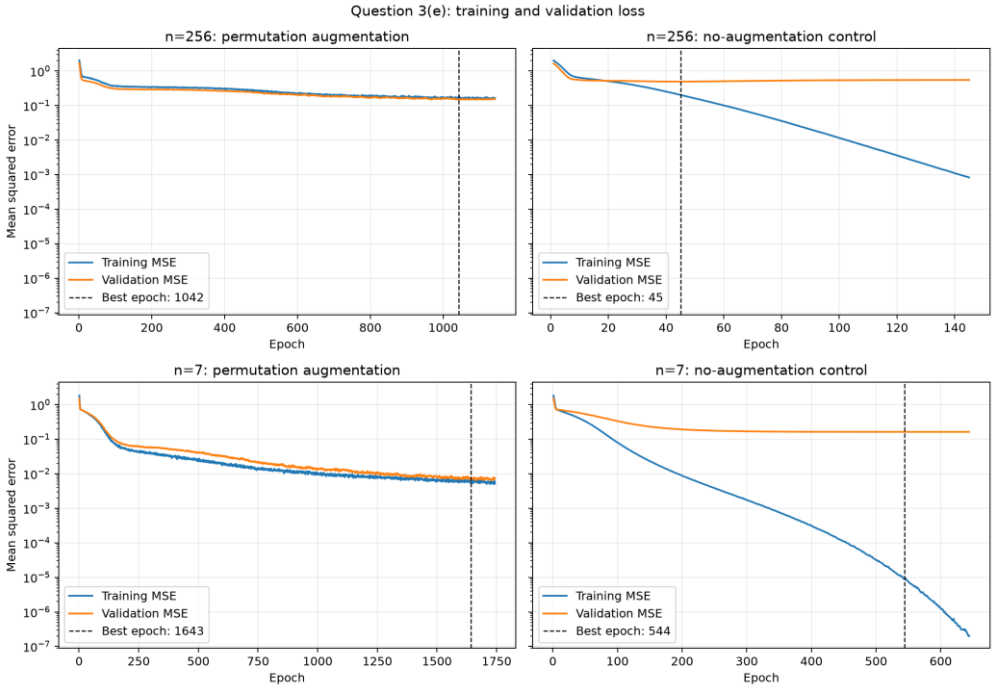
וחישבנו מחדש את המטרה; ל-MLP הקטן נבחרו רוחבים שנותנים בדיוק 53,571 פרמטרים, כמו במודל הגדול. כך צומצם הבלבול בין השפעת גודל החבורה לבין קיבולת המודל.

שגיאת הקירוב היא MSE מול השונות האמפירית. את שגיאת האינווריאנטיות חישבנו על $M =$ 150 דוגמאות test ובאותו אוסף קבוע של תמורות לכל המודלים המושווים:

$$E_{\text{inv}} = \frac{1}{M}\sum_{r=1}^{M}\left\|F(\pi_r \cdot X^{(r)}) - F(X^{(r)})\right\|_2.$$

תוצאות.

גרפי ה-training loss וה-validation loss של ארבעת המודלים מוצגים להלן; הקו המקווקו מציין את ה-epoch שנשמר על ידי early stopping.



n	מדד	לפני אימון	לאחר הגדלת נתונים	לאחר ביקורת
256	Test MSE	2.119865	0.177690	0.581739
256	שגיאת אינווריאנטיות ממוצעת	0.187865	0.710025	0.486882
256	מיטבי validation MSE	—	0.147973	0.496459
256	מיטבי epoch	—	1042	45
7	Test MSE	1.998376	0.006223	0.141360
7	שגיאת אינווריאנטיות ממוצעת	0.157501	0.164389	0.505162
7	מיטבי validation MSE	—	0.006402	0.162713
7	מיטבי epoch	—	1643	544

דיון בתוצאות.

עבור $n = 256$ הגדלת הנתונים הפחיתה בשיעור 69.46% את Test MSE ביחס לביקורת, ולכן שיפרה את יכולת החיזוי. עם זאת, שגיאת האינווריאנטיות הייתה 0.710025 לעומת 0.486882 בביקורת. אין להסיק מכך שמודל הביקורת למד פונקציה אינווריאנטית טובה יותר: המדד משווה בין שני פלטי המודל ואינו בודק אם הם קרובים למטרה. מודל כמעט קבוע יכול לקבל שגיאת אינווריאנטיות נמוכה אף שתחזיתו גרועה. ואכן, מודל הביקורת נעצר כבר ב-epoch 45, כאשר פלטיו החלשים היו פחות רגישים לשינוי סדר.

עבור $n = 7$ התקבלה מגמה הפוכה וברורה: הגדלת הנתונים הפחיתה בשיעור 95.60% את Test MSE, ובשיעור 67.46% את שגיאת האינווריאנטיות ביחס לביקורת. מאחר שמספר הפרמטרים, היפרפרמטרי האימון והמשתנים החבויים של הדוגמאות הותאמו בין שני ערכי n, התוצאה תומכת בהשערה שמרחב התמורות הוא גורם מרכזי.

בפועל לא השתמשנו בתמורה אחת משותפת ל-batch: לכל דוגמה בתוך ה-batch נדגמה תמורה בלתי־תלויה, ולכן דוגמה מסוימת ראתה תמורה אחת בכל epoch. כדי לכמת את הכיסוי, נסמן $N = n!$ וב-D_k את מספר התמורות השונות שנראו לדוגמה לאחר k epochs. עבור תמורה קבועה, ההסתברות שלא תידגם באף אחת מן ההצגות היא $\left(1\text{-}\frac{1}{N}\right)^k$. סכימת האינדיקטורים לכך שכל תמורה הופיעה לפחות פעם אחת ולינאריות התוחלת נותנות

$$\mathbb{E}[D_k] = N(1 - \left(1\text{-}\frac{1}{N}\right)^k).$$

עבור $n = 7$ ו-$k = 1643$ מתקבל $\mathbb{E}[D_k] \approx 1402$, כלומר כ-27.8% מתוך $7! = 5040$ לכל דוגמה. לעומת זאת, עבור $n = 256$ ו-$k = 1042$, מספר התמורות השונות הצפוי הוא בקירוב 1,042, אך שיעורן מתוך $256!$ זניח. נוסף על כך, ב-batch מלא בגודל 64 נדגמות 64 תמורות בלתי־תלויות ולא תמורה משותפת אחת; הדבר מגדיל את כיסוי החבורה ומפחית את התלות בין תרומות הדוגמאות לגרדיאנט. חישוב זה מחזק את ההסבר לכך שהגדלת הנתונים הייתה יעילה בהרבה עבור $n = 7$.

זהו אישוש אמפירי חזק אך לא הוכחה מוחלטת, משום ששינוי n משנה גם את ממד הקלט ואת השונות האמפירית המשמשת כמטרה. נוסף על כך, גם עבור $n = 7$ שגיאת האינווריאנטיות לאחר הגדלת נתונים הייתה 0.164389, מעט מעל הערך ההתחלתי 0.157501. לכן הגדלת נתונים מעודדת אינווריאנטיות אך אינה מבטיחה אותה; ההשוואה המבוקרת מצביעה על כך שהקושי לחקור את $256!$ התמורות מסביר חלק מרכזי מן התוצאה בניסוי הגדול.

שאלה 4

Question 4 (Benchmarking permutation-invariant networks on 3D shape classification (35 pts))**.** *In this question you compare the approaches of Question 3 on a real task: classifying 3D point clouds, treated as sets of points, into object categories. You will evaluate accuracy and computational cost, and analyze the trade-offs.*

Learning task and data. *Classify shapes from the ModelNet40 dataset (40 object categories; 9,843 training and 2,468 testing CAD meshes, sampled to point clouds).*

- ***Download.*** *Use the ModelNet40 "normal resampled" release:*

  https://www.dropbox.com/scl/fi/qcddtiathbfcp6v3wimoi/modelnet40_normal_resampled.zip?rlkey=3t9jz6lr954wxmfj4xp6u9m2a&dl=0

  If this link is unavailable, any standard mirror of ModelNet40 "normal resampled" is acceptable; ***state your source*** *in the report.*

- ***Format.*** *Each point cloud is a text file under a per-class folder; each row holds six numbers: the first three are the (x, y, z) coordinates and the next three are the normal direction.* ***Unless stated otherwise, use the first 256 points of each cloud****, i.e. represent each cloud as a 256×3 matrix of coordinates. Example loader:*

```
import numpy as np
data = np.loadtxt('airplane/airplane_0001.txt', delimiter=',')[:256]
coords = data[:, :3]  # x, y, z
```

- ***Sanity check.*** *Visualize several classes with a 3D scatter plot and confirm the data looks reasonable; include one or two such plots.*

Architectures. *Use permutation-invariant classifiers built from the Question 3 components:*

(a) *Canonization-based network.*

(b) *Symmetrization network. (Full symmetrization is infeasible for 256 points; use a small number of points — possibly fewer than 10 — for which the experiments below run in reasonable time. State the number you used.)*

(c) *Sampled-symmetrization network. (Sample 5% of the possible permutations; you may use more points than in (b), up to all 256.)*

(d) *Linear equivariant layers with pointwise nonlinearities, followed by an invariant pooling and a classifier head.*

(e) *Data augmentation with a standard (non-invariant) architecture.*

For the canonization-based and symmetrization networks, implement the base model as a standard MLP. ***Bonus (up to 5 pts):*** *also implement them with a Transformer plus positional encoding.*

Experimental protocol.

- *Train with* `CrossEntropyLoss`*. Fix and report a random seed.*
- ***Validation / early stopping.*** *Hold out a fixed validation set — carve a disjoint 20% of your selected training samples (stratified across classes) — and use it for early stopping.* ***Report final accuracy on the official ModelNet40 test set*** *(not on validation).*
- ***Training-set size.*** *Repeat training with 5, 10, and 50 samples per class. If these give no appreciable difference, adjust the values and say so.*
- ***Time budget.*** *Cap each single experiment at 0.5 hours of wall time. If an experiment exceeds the cap, mark it "Out of Time" (OOT) and report the last test accuracy reached.*

- ***Bonus (up to 3 pts):*** *repeat using the surface normals as additional input features and report the effect.*

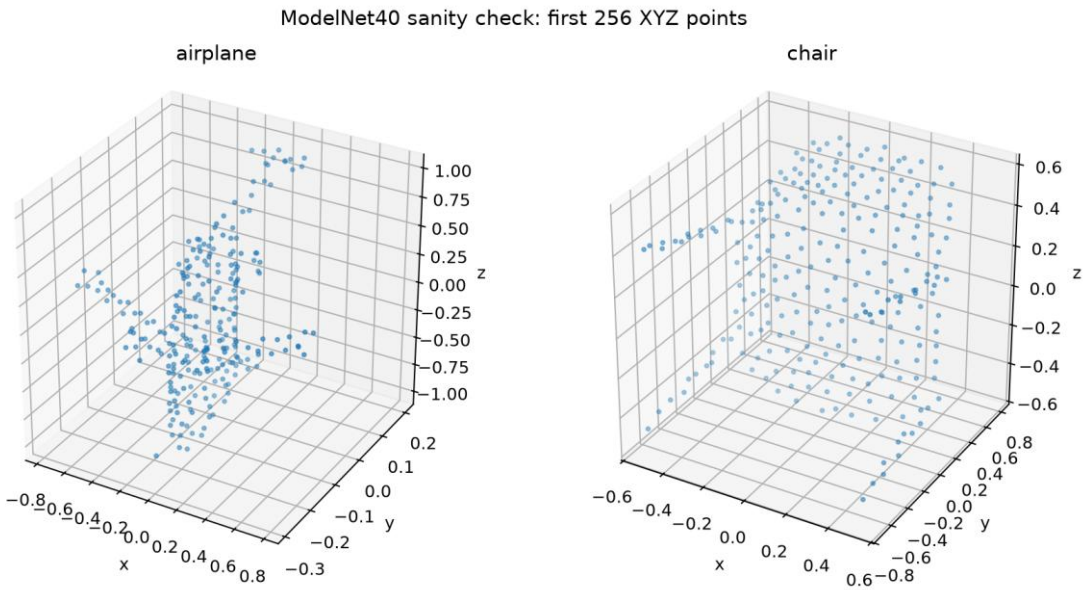
Analysis and reporting.

1. *One plot of test accuracy vs. number of training examples per class, one curve per architecture.*
2. *A table of the (wall-clock) runtime of each architecture.*
3. *A discussion of the trade-offs between the approaches in terms of: accuracy; computational efficiency; scalability with set size and feature dimension; and ease of implementation.*
4. *The challenges you met during implementation and experimentation.*
5. *A recommendation: which approach(es) suit which scenario (e.g. small vs. large sets, scarce vs. abundant compute).*
6. *Why linear equivariant layers may be especially well suited to this task, compared with the other architectures.*
7. ***Bonus (up to 5 pts).*** *This task has a further symmetry: rotating the whole point cloud does not change its class. Consider the matrix of pairwise distances between points. Which group leaves it invariant? Use it to design rotation-invariant per-point features.* ***Suggestion:*** *first "centralize" the cloud so the points have mean zero and the point of maximal norm lies on the unit sphere.*

מסגרת משותפת לשאלה 4.

המשימה היא סיווג עננים נקודות תלת-ממדיים מתוך ModelNet40 לאחת מ-40 מחלקות. כל דוגמה מיוצגת כמטריצה $X \in \mathbb{R}^{n\times 3}$, והסימטריה הרלוונטית היא שינוי סדר השורות על ידי S_n. השתמשנו במהדורת ModelNet40 normal resampled מן הקישור בשאלה, בשלוש קואורדינטות XYZ הראשונות וב-256 הנקודות הראשונות בכל ענן, ללא הנורמלים, מרכוז או שינוי קנה מידה.

בדיקת הסבירות הבאה מציגה את 256 הנקודות הראשונות מדוגמאות של airplane ושל chair. המבנים הגאומטריים ניתנים לזיהוי והטווחים סופיים, ולכן טעינת הנתונים תקינה.



קבענו seed 2319 ובנינו לכל מחלקה מאגר מדורג ומשותף של 5, 10 ו-50 דוגמאות. מכל חמישייה הוקצו ארבע דוגמאות עבור training ואחת עבור validation, ולכן התקבלו בהתאמה 4/1, 8/2 ו-40/10 דוגמאות למחלקה. קבוצת test הרשמית, הכוללת 2,468

דוגמאות, לא השתתפה בבחירת המודל. כל הארכיטקטורות קיבלו אותם פיצולים ואותו batch order, ומשקלי ההתחלה הותאמו כאשר מבנה MLP הבסיסי היה זהה.

היפרפרמטר או בחירת מימוש	ערך
מקור הנתונים	ModelNet40 normal resampled
seed	2319
דוגמאות שנבחרו למחלקה	5, 10, 50
training/validation למחלקה	4/1, 8/2, 40/10
פונקציית הפסד	CrossEntropyLoss
optimizer	Adam
learning rate	1e-4
weight decay	1e-4
batch size	64
מספר epochs מרבי	2,000
early-stopping patience	100
מגבלת זמן לניסוי	1,800 seconds
dropout / scheduler	none / none
חומרה	NVIDIA GeForce RTX 3080

המודלים תוכננו בתקציב משותף של כ-56,000 פרמטרים כדי שההשוואה לא תשקף הבדל שרירותי בקיבולת. כל החישובים בוצעו באמצעות PyTorch על NVIDIA GeForce RTX 3080. בניסויי full symmetrization וכן sampled symmetrization השתמשנו ב-$n = 7$: שיטת full symmetrization אינה ישימה עבור 256 נקודות, והבחירה הזהה בסעיפים ב-ג מאפשרת לבודד את השפעת הדגימה. משום כך, בהשוואה למודל equivariant network ולמודל canonization-based network יש גם הבדל בכמות המידע בקלט, ולא רק בארכיטקטורה.

סעיף א. Canonization-based network.

מיינו את 256 השורות בסדר lexicographic יציב, שיטחנו את המטריצה והעברנו אותה דרך MLP במבנה 40 → 64 → 64 → 768 עם ReLU. לרשת 55,976 פרמטרים. אם c היא פעולת canonization, אז לכל תמורה π מתקיים $c(\pi X) = c(X)$, ולכן מתקבלת exact invariance ללא תלות במשקלי MLP.

סעיף ב. Full symmetrization network.

עבור שבע הנקודות הראשונות השתמשנו במודל MLP בסיסי במבנה 21 → 207 → 207 → 40 וב-55,930 פרמטרים. לכל קלט חישבנו logits עבור כל $7! = 5040$ התמורות ומיצענו אותם לפני חישוב loss. הממוצע על החבורה המלאה מבטיח exact invariance, אך העלות הפקטוריאלית מגבילה את השיטה לקבוצות קטנות.

סעיף ג. Sampled-symmetrization network.

השתמשנו באותו MLP בסיסי ובאותן שבע נקודות, אך דגמנו פעם אחת ללא החזרה תת-קבוצה קבועה של $B = 252$ תמורות, כלומר 5% מתוך S_7. מיצוע logits על אותה תת-קבוצה בכל החישובים נותן Monte Carlo approximation למיצוע המלא והפחתה תאורטית של פי 20 במספר הערכות MLP, אך אינו מבטיח exact invariance.

סעיף ד. Linear equivariant layers with mean pooling.

אופרטור ההטלה על מרחב equivariant maps הוא

$$P(W) = \frac{1}{|G|}\sum_{g\in G} \rho_{\text{out}}(g^{-1}) W \rho_{\text{in}}(g).$$

עבור פעולת התמורות על אינדקסי הנקודות קיימים שני orbits של זוגות אינדקסים: $i = j$ ו- $i \neq j$. לכן פרמטרנו ישירות את תמונת ההטלה באמצעות שני בלוקי פרמטרים משותפים:

$$y_i = Ax_i + B\sum_j x_j.$$

מימשנו שתי linear equivariant layers ברוחבים 3 → 128 → 128 עם pointwise ReLU, לאחריהן mean pooling וראש סיווג 128 → 128 → 40. למודל equivariant network יש 55,464 פרמטרים. השכבות הפנימיות הן equivariant לכל תמורה לפי הבנייה, ופעולת mean pooling יוצרת invariant output.

סעיף ה. Data augmentation with permutations.

השתמשנו במודל standard non-invariant MLP במבנה 768 → 64 → 64 → 40 וב- 55,976 פרמטרים. לפני כל הצגה של דוגמה באימון דגמנו תמורת שורות חדשה ובלתי-תלויה; בתוך batch כל דוגמה קיבלה תמורה אחרת. כך האימון מעודד invariance, אך מרחב הפונקציות של הרשת נשאר תלוי-סדר ואין ערובה להתנהגות על תמורות שלא נצפו.

Bonus. Transformer עם positional encoding.

בהתאם לנוסח השאלה החלפנו את ה-MLP הבסיסי של שיטות ה-canonization וה- full symmetrization באותו Transformer תלוי-סדר. כל נקודה הוטלה לממד 56, ולייצוג במיקום i נוסף fixed sinusoidal positional encoding לפי

$$p_{i,2k} = \sin\left(\frac{i}{10000^{\frac{2k}{56}}}\right), p_{i,2k+1} = \cos\left(\frac{i}{10000^{\frac{2k}{56}}}\right).$$

הייצוג עבר בשתי TransformerEncoder layers עם ארבעה attention heads, רוחב feed-forward של 104, ReLU ו-dropout=0. לאחר mean pooling הופעל ראש סיווג 56 → 56 → 40. לכל אחד משני המודלים 55,408 פרמטרים, קרוב לתקציב המשותף של כ-56 אלף.

במודל הראשון מיינו את כל 256 הנקודות בסדר lexicographic ורק לאחר מכן הפעלנו את ה-Transformer. לכן לכל תמורה π מתקיים $T(c(\pi X)) = T(c(X))$, והמודל exactly invariant. ה-positional encoding מאפשר ל-self-attention להשתמש במיקום בתוך הסדר הקנוני.

במודל השני השתמשנו בשבע הנקודות הראשונות ובאותו מיצוע מלא כמו בסעיף ב:

$$F_T(X) = \frac{1}{7!}\sum_{\pi\in S_7} T(\pi X).$$

ה-positional encoding הופך את מודל הבסיס לתלוי-סדר, ואילו המיצוע על כל 5,040 התמורות מחזיר exact invariance. לצורך זיכרון בלבד עובדו התמורות ב-CUDA chunks של 512. באימון השתמשנו ב-microbatch size 32 וב-gradient accumulation לשמירת effective batch size 64.

בדיקות CUDA אישרו שה-Transformer הבסיסי אכן תלוי בסדר. שגיאת האינווריאנטיות המרבית לאחר canonization הייתה אפס, ולאחר המיצוע המלא הייתה 2.98e-8, מתחת ל- atol=1e-5. גם צורת הפלט ובדיקת הגרדיאנטים עברו.

Bonus. שימוש ב-surface normals.

בניסוי ה-Bonus חזרנו על כל 15 הניסויים לאחר שהרחבנו כל נקודה מן הייצוג (x, y, z) לייצוג (x, y, z, n_x, n_y, n_z). בכל permutation המיקום וה-normal של אותה נקודה הועברו יחד. במודל canonization המיון ה-lexicographic נותר מבוסס על קואורדינטות XYZ בלבד, וכל השורה בת ששת המאפיינים הועברה לפי אותו סדר. לא נמצאו שתי נקודות בעלות אותן קואורדינטות XYZ באף אחד מ-4,468 עננים הנקודות שנבדקו. חלוקות הנתונים, seed 2319, רוחבי השכבות וה-hyperparameters של האימון נשמרו ללא שינוי.

ההשוואה אינה parameter-matched, משום שהכפלת מספר מאפייני הקלט מ-3 ל-6 הגדילה באופן שונה את מספר הפרמטרים בכל ארכיטקטורה. לאחר ההרחבה היו 105,128 פרמטרים במודלים canonization-based network ו-data augmentation, 60,277 פרמטרים במודלים full symmetrization ו-sampled symmetrization, ו-56,232 פרמטרים במודל equivariant network. לכן אין לפרש את התוצאות כהשוואה שמבודדת את השפעת הסימטריה בלבד.

Bonus. סימטריה לסיבובים באמצעות pairwise distances.

החבורה המשאירה את מטריצת ה-pairwise distances ללא שינוי היא O(3). אמנם סימטריית הסיבובים עצמה היא SO(3), אך מרחקים נשמרים גם תחת reflections, ולכן הם invariant תחת החבורה הגדולה יותר O(3). תחילה מרכזנו ונרמלנו כל ענן נקודות לפי

$$\bar{x} = \frac{1}{n}\sum_{i=1} x_i \,, s = \max_j \| x_j - \bar{x} \|_2, \hat{x}_i = \frac{x_i - \bar{x}}{s}.$$

לאחר פעולה זו ממוצע הנקודות הוא אפס והנקודה בעלת הנורמה המרבית נמצאת על ספירת היחידה. הגדרנו את מטריצת המרחקים הריבועיים

$$D_{ij} = \| \hat{x}_i - \hat{x}_j \|_2^2.$$

הריבוע חוסך חישוב שורש ואינו משנה את האינוריאנטיות או את סדר המרחקים. נסמן $v_{ij} = \hat{x}_i - \hat{x}_j$. לכל $Q \in O(3)$ מתקיים $Q^T Q = I$, ולכן

$$D_{ij}(Q\hat{X}) = \left(Qv_{ij}\right)^T (Qv_{ij}) = v_{ij}^T Q^T Q v_{ij} = v_{ij}^T v_{ij} = D_{ij}(\hat{X}).$$

תמורת נקודות $P \in S_n$ פועלת על המטריצה לפי $D(PX) = PD(X)P^T$. שורה גולמית של D עדיין תלויה בסדר העמודות. לכן הגדרנו לכל נקודה את וקטור המאפיינים

$$\varphi_i(X) = \mathrm{sort}(D_{i1}, \dots, D_{in}) \in \mathbb{R}^n.$$

מיון כל שורה מסיר את סדר הנקודות האחרות, אך שומר את השורה מחוברת לנקודת העוגן שלה. לכן המאפיינים הם O(3)-invariant ו-permutation-equivariant: כאשר מסדרים מחדש את נקודות הקלט, שורות המאפיינים מסתדרות מחדש באותו אופן.

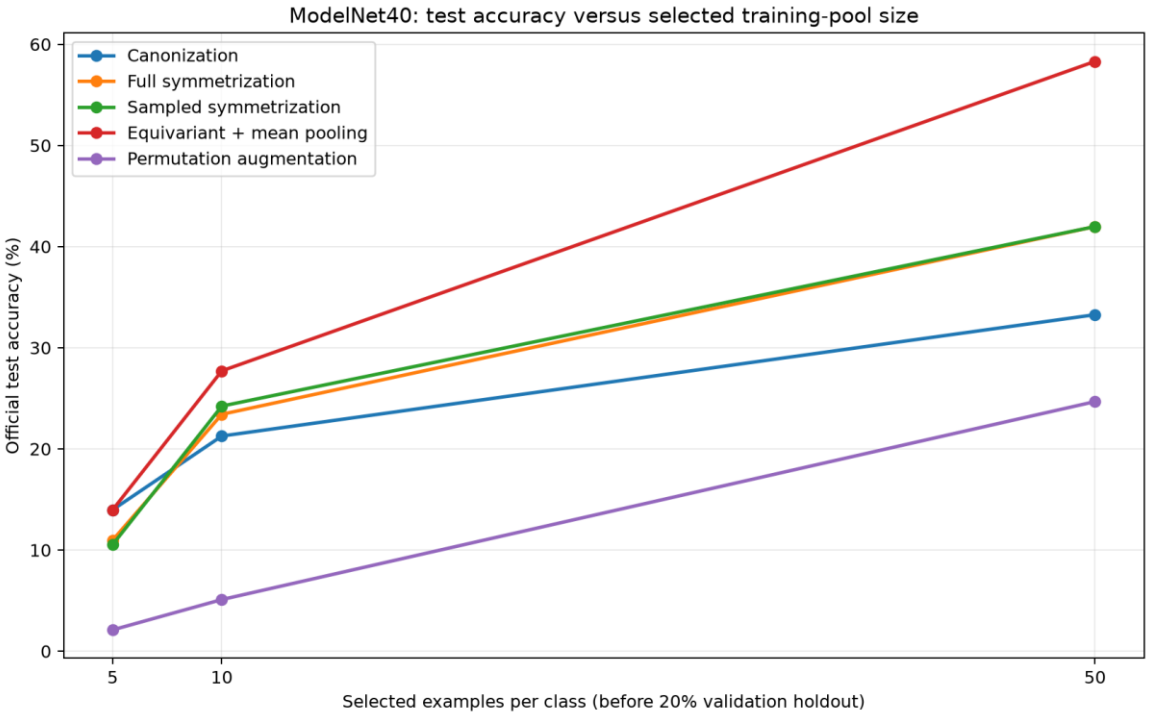
בניסוי השתמשנו ב-256 קואורדינטות XYZ בלבד כדי לבודד את השפעת סימטריית הסיבובים. כל נקודה קיבלה 256 מרחקים ריבועיים ממוינים, ולאחר מכן הופעלה רשת DeepSets במבנה 72 → 72 → 256 עם pointwise ReLU, אחריה mean pooling וראש סיווג 72 → 40 → 72. למודל 55,552 פרמטרים, לעומת 55,464 במודל equivariant network המקורי. כל שאר חלוקות הנתונים וה-hyperparameters נשמרו ללא שינוי.

בדיקת CUDA שכללה rotation, reflection, הזזה ושינוי קנה מידה נתנה שגיאת מאפיינים מרבית של 1.43e-6 ושגיאת logits מרבית של 4.47e-8. שגיאת ה- permutation equivariance של מאפייני הנקודות הייתה 4.77e-7, ושגיאת ה-permutation invariance של הפלט הייתה 2.98e-8. גם בדיקת הגרדיאנטים עברה.

תוצאות, ניתוח והמלצה.

1. Test accuracy כתלות במספר דוגמאות האימון.

הגרף הנדרש מציג את חמש הארכיטקטורות המרכזיות על קלט XYZ. כדי לשמור על קריאותו, כל תוצאות ה-Bonus מוצגות יחד עם תוצאות הבסיס בטבלה המאוחדת שאחריו ולא נוספו לגרף כ-13 עקומות צפופות.



בטבלת הדיוק כל תא מציג overall accuracy ולאחריו mean-class accuracy. כל שורת הרחבה מסומנת במפורש כ-Bonus, ותוצאת ניסוי שהגיע למגבלת 30 הדקות מסומנת OOT.

50	10	5	ארכיטקטורה / variant
33.27% / 31.25%	21.27% / 19.12%	13.98% / 13.58%	Canonization (XYZ)
41.98% / 41.84%	23.42% / 21.83%	10.98% / 11.61%	Full symmetrization (XYZ)
41.98% / 40.79%	24.23% / 22.16%	10.53% / 11.43%	Sampled symmetrization (XYZ)
58.31% / 55.82%	27.71% / 26.83%	13.98% / 14.91%	Equivariant network (XYZ)
24.68% / 20.77%	5.11% / 7.86%	2.11% / 5.40%	Data augmentation (XYZ)
56.08% / 54.72%	31.89% / 30.37%	21.03% / 21.91%	Bonus: Canonization + Transformer (XYZ)
42.67% / 40.36% (OOT)	27.15% / 27.58%	17.75% / 16.90%	Bonus: Full symmetrization + Transformer (XYZ)
35.01% / 32.94%	22.85% / 19.73%	19.69% / 16.63%	Bonus: Canonization + surface normals
38.13% / 37.01%	3.85% / 3.86%	3.40% / 3.18%	Bonus: Full symmetrization + surface normals
38.01% / 37.01%	3.77% / 3.81%	3.77% / 3.22%	Bonus: Sampled symmetrization + surface normals
68.64% / 65.80%	33.87% / 31.45%	21.07% / 23.32%	Bonus: Equivariant network + surface normals
25.69% / 22.19%	4.38% / 7.92%	2.31% / 6.53%	Bonus: Data augmentation + surface normals
35.98% / 37.17%	20.83% / 22.82%	18.52% / 20.36%	Bonus: O(3)-invariant pairwise-distance features

2. Wall-clock runtime.

זמני ה-wall-clock של האימון וה-validation מוצגים בשניות עבור אותן שורות ובאותו סדר.
OOT מציין שהאימון נעצר במגבלת 1,800 השניות, ולאחריו מופיע הזמן שנמדד בפועל.

ארכיטקטורה / variant	5	10	50
Canonization (XYZ)	3.94	4.95	17.32
Full symmetrization (XYZ)	28.47	146.84	751.42
Sampled symmetrization (XYZ)	5.93	32.26	134.91
Equivariant network (XYZ)	6.82	13.36	147.82
Data augmentation (XYZ)	3.75	5.51	68.17
Bonus: Canonization + Transformer (XYZ)	24.98	32.50	120.52
Bonus: Full symmetrization + Transformer (XYZ)	760.74	1495.32	OOT (1800.2)
Bonus: Canonization + surface normals	5.38	4.55	15.73
Bonus: Full symmetrization + surface normals	12.03	25.07	711.79
Bonus: Sampled symmetrization + surface normals	2.92	5.26	144.76
Bonus: Equivariant network + surface normals	11.57	18.03	272.26
Bonus: Data augmentation + surface normals	4.44	6.33	90.78
Bonus: O(3)-invariant pairwise-distance features	22.23	50.44	980.86

3. Trade-offs בין כל הגישות.

התוצאה הטובה ביותר בכל שלושת גדלי האימון התקבלה בשורת ה-Bonus של equivariant network + surface normals: 21.07%, 33.87% ו-68.64%. ב-5 דוגמאות היתרון על Canonization + Transformer היה 0.04 נקודת אחוז בלבד, אך ב-10 וב-50 דוגמאות גדל הפער. התוצאה מחזקת את היתרון של pointwise weight sharing כאשר כל 256 הנקודות וה-surface normals זמינים.

החלפת ה-MLP לאחר canonization ב-Transformer העלתה את הדיוק מ-13.98%, 21.27% ו-33.27% ל-21.03%, 31.89% ו-56.08%. Self-attention מסוגל ללמוד קשרים בין נקודות ומנצל את הסדר הקנוני באמצעות positional encoding. עם זאת, המודל עדיין יורש את אי-הרציפות של המיון, ועלות ה-attention גדלה ריבועית במספר הנקודות.

גם Full symmetrization + Transformer שיפר את הדיוק ב-5 וב-10 דוגמאות ל-17.75% ול-27.15%, אך דרש 760.7 ו-1,495.3 שניות. בניסוי של 50 דוגמאות התקבלה תוצאת (OOT) 42.67%. מכאן שה-Transformer משפר את מודל הבסיס, אך אינו פותר את העלות הפקטוריאלית של מיצוע על $n!$. עבור אותו בסיס, canonization היא הדרך המעשית בהרבה לקבל exact permutation invariance.

במיצוע מבוסס MLP, דגימת 252 תמורות שמרה על דיוק קרוב מאוד למיצוע המלא וקיצרה את זמן האימון פי 4.55-5.57. ההאצה קטנה מן היחס התאורטי 20 משום שהעברת batches, פעולות ה-optimizer, ה-validation ותקורת Python/CUDA אינן קטנות באותו יחס, וה-GPU יעיל יותר על אצוות גדולות.

הוספת surface normals שיפרה במיוחד את ה-equivariant network, אך לא את שתי שיטות ה-symmetrization בעלות שבע הנקודות. שתי האחרונות קיבלו פחות מידע וחסרות internal pointwise weight sharing; תוצאותיהן הכמעט זהות מראות שהכשל אינו נובע בעיקר מן ה-Monte Carlo approximation. ההשוואה אינה parameter-matched, ולכן המסקנה מוגבלת לתנאי הניסוי.

מאפייני ה-pairwise distances נתנו 18.52%, 20.83% ו-35.98%. ב-5 דוגמאות האילוץ של O(3) invariance שימש inductive bias יעיל; עם יותר נתונים בלט אובדן המידע הנגרם ממיון עצמאי של כל שורת מרחקים, וכן האפשרות שהכיוון הקנוני של ModelNet40 מכיל מידע מועיל. בניית המטריצה והמיון גדלים לפחות ריבועית ב-set size.

מבחינת scalability, עלות full symmetrization היא פקטוריאלית; sampled symmetrization מחליפה אותה במספר דגימות קבוע במחיר שגיאת קירוב; Transformer attention ומאפייני מרחקים גדלים ריבועית; ואילו linear equivariant layers עם mean pooling גדלות לינארית במספר הנקודות ומספר הפרמטרים שלהן אינו תלוי בו. מבחינת ease of implementation, ה-MLP עם canonization או data augmentation הם הפשוטים ביותר; שכבות שקולות דורשות מימוש ייעודי אך אינן דורשות enumeration בזמן אימון או inference.

4. אתגרי המימוש והניסוי.

האתגרים המרכזיים היו התאמת מספרי הפרמטרים בין ארכיטקטורות שונות, בניית פיצולים מדורגים ומאוזנים, שמירת אותה תת-קבוצת תמורות, ועיבוד 5,040 תמורות בלי לחרוג מזיכרון ה-GPU. ב-Transformer המדויק נדרש שילוב של CUDA chunks, microbatches ו-gradient accumulation. בנוסף נדרש לשמור surface normals מחוברים לנקודותיהם, ולוודא בנפרד אינוריאנטיות ל-O(3), הזזה וקנה מידה. כל checkpoint הוערך על קבוצת ה-test הרשמית בלבד לאחר בחירתו לפי validation.

5. Recommendation לפי תרחיש.

לענני נקודות גדולים ולדיוק מרבי נעדיף equivariant network עם mean pooling, ובפרט עם surface normals כאשר הן זמינות. אם מבקשים בסיס Transformer עם exact invariance, נעדיף Canonization + Transformer: הוא ניצל את כל 256 הנקודות והשיג דיוק גבוה בלי עלות פקטוריאלית. Full symmetrization + Transformer מתאים רק לקבוצות זעירות ולתקציב חישוב גדול.

עבור קבוצה קטנה ודרישת מיצוע מדויק ניתן להשתמש ב-full symmetrization; כאשר החישוב מוגבל, sampled symmetrization נותנת פשרה טובה. מאפייני pairwise distances מתאימים כאשר rotation invariance היא דרישה קשיחה והקבוצה אינה גדולה, אך עדיף מודל גרפי השומר התאמות בין מרחקים. Data augmentation מתאימה בעיקר כאשר חייבים לשמר ארכיטקטורה קיימת ויש נתונים רבים; היא אינה מבטיחה אינוריאנטיות.

6. מדוע linear equivariant layers מתאימות במיוחד.

כפי שנגזר בסעיף ד, linear equivariant layers מפרמטרות ישירות את תמונת אופרטור ההטלה: שני orbits של זוגות אינדקסים נותנים שני בלוקי משקלים משותפים. לכן כל שכבה היא equivariant by design, ו-mean pooling נותנת פלט אינוריאנטי בלי למנות תמורות ובלי ללמוד את הסימטריה מן הנתונים. מנגנון weight sharing מצמצם את מרחב הפונקציות, משפר sample efficiency ומשאיר את מספר הפרמטרים בלתי תלוי במספר הנקודות.

ההשוואה המאוחדת מחזקת טענה זו: ה-equivariant network היה היעיל ביותר בניצול המידע הנוסף מן ה-surface normals, עקף את שיטות המיצוע בקלט הגדול, ונמנע הן מן העלות הפקטוריאלית של full symmetrization והן מן העלות הריבועית של Transformer attention ומטריצת המרחקים. לכן זוהי הארכיטקטורה המתאימה ביותר למשימת הסיווג שנבדקה.