FUNCTIONAL SUBSPACE, WHERE LANGUAGE MODELS CAN USE VECTOR ALGEBRA TO SOLVE PROBLEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) were invented for natural language tasks such as translation, but they have proved that they can perform highly complex functions across domains. Additionally, they have been thought to develop new skills without being trained on them. These learning capabilities lead to LLMs adoption in a wide range of domains. Thus, it is imperative that we understand their operating mechanisms and limitations for proper diagnostics and repair. The earlier studies proposed that high level concepts are encoded as linear directions in LLMs activation space and that the geometry of embeddings have semantic meanings. Inspired by these studies, we hypothesize that LLMs may use subspaces and vector algebra in subspaces to perform tasks. To address this hypothesis, we analyze LLMs' functional modules and residual streams collected from LLMs engaging in in-context learning (ICL), one of the emergent abilities. Our analyses suggest that 1) LLMs can create subspaces, where evidence can be accumulated and 2) ICL tasks can be solved via simple algebraic operations in subspaces.

1 INTRODUCTION

The explosive growth in LLMs applications indicates that LLMs are highly capable learners (Naveed et al., 2025; Laskar et al., 2024; Minaee et al., 2025). After being pretrained to predict the next tokens, LLMs can be fine-tuned for complex tasks. Notably, a line of studies suggests that fine-tuning may not be strictly necessary, as LLMs can obtain "emergent" abilities (Matarazzo & Torlone, 2025; Lu et al., 2024; Brown et al., 2020; Wei et al., 2022). Pretraining does not aim to train LLMs to write code, but still, after pretraining, they can generate code. However, it remains poorly understood how LLMs can generalize and develop emergent abilities, making the diagnosis of LLMs' operations almost impossible. Before LLMs are being fully utilized in safety-critical domains, it is imperative to better understand the principles behind LLMs' generalization and emergent abilities to diagnose and repair LLMs' operations

To this end, we investigate how LLMs support in-context learning (ICL), one of LLMs' emergent abilities (Brown et al., 2020; Wei et al., 2022). ICL allows users to reconfigure LLMs to perform specific tasks without retraining or fine-tuning. For instance, a pretrained LLM can function as a thesaurus to an input prompt query, if the input prompt provides a pair of synonyms. ICL tasks are often simpler than general tasks such as coding, but LLMs can still 1) extract task information from input prompts and 2) apply the information to answer users' queries. These two basic functions are essential for LLMs to interpret users' intentions and generate correct responses, which means that investigating ICL may lead us to a better understanding of how LLMs perform more general tasks.

Our study is inspired by two lines of studies. First, linear representation hypothesis posits that high level concepts are encoded as linear directions in LLMs' activation space (Park et al., 2023; Jiang et al., 2024b). According to this hypothesis, tokens (i.e., sub-words) associated with 'happiness' are mapped onto vectors pointing to a 'happy' direction, and those associated with 'unhappiness' are mapped onto vectors pointing to a 'sad' direction. Second, the geometry of word embeddings may be linked to semantic meanings. For example, in the embedding space, the word 'king' can be converted to the word 'queen' if it is subtracted by 'man' and added by 'woman' (Mikolov et al., 2013b;a).

Based on these studies, we hypothesize that LLMs can support ICL by projecting input tokens into a subspace, where a desired task can be solved by using vector algebra. To address this hypothesis, we use principal component analysis (PCA) and linear regression analysis to determine the existence

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

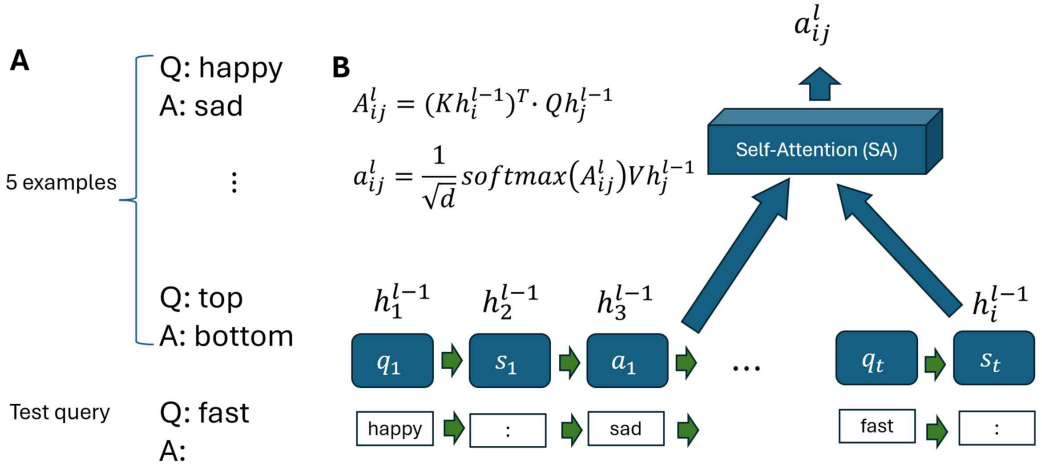


Figure 1: ICL prompts. (A), The structure of the used ICL prompts. (B), SA layer l computes interactions between tokens h_i^{l-1} and h_j^{l-1} using K, Q and V, which are trainable matrices.

of a LLM subspace, where the answer can be inferred by vector algebra. Our empirical evaluations suggest that ICL tasks can be considered vector algebra problems in some of the LLMs' subspaces.

2 Subspace, naturally generated by LLMs

In this section, we discuss how transformers' functional modules generate subspaces and use them to support ICL. In section 2.1, we briefly discuss ICL tasks used in this study, and in section 2.2, we discuss potential mechanisms by which LLMs' functional modules may use subspaces to support ICL.

2.1 ICL tasks

Brown et al. (2020) found that LLMs can infer correct answers, if they are exposed to a few examples of desired tasks; see (Wei et al., 2022) as well. When a few pairs of synonyms are included in the input prompts, LLMs return a synonym to the last query word. Thus, ICL prompts contain multiple examples of desired tasks and a query. In our study, we use the dataset generator released by an earlier study (Todd et al., 2024) to create ICL prompts for 6 different tasks, antonyms, synonyms, countries-their capital cities (country-capital), English-French words, company-product and person-sport. All 6 tasks can be demonstrated with two words separated by a separator. Throughout this study, we refer to them as queries, separators and answers. For all tasks, we provide 5 example pairs of queries and answers and a single test query (without a corresponding answer) and direct LLMs to find its counterpart using example pairs. Each query and answer are marked by the separators, 'Q:' and 'A:' respectively (Fig. 1A).

2.2 Transformer blocks and subspaces

LLMs first convert words in input prompts into tokens (i.e., sub-words) and use a sequence of transformer layers to process them progressively, each of which consists of self-attention (SA) layer and Feed-Forward Network (FFN) (Vaswani et al., 2023). A transformer layer l receives inputs h^{l-1} from its previous layer and generates h^l using its own SA and FFN. As the transfer layer uses residual connections, h^l can be recursively expressed as $h^l = h^{l-1} + a^l + m^l(a^l + h^{l-1})$ (Meng et al., 2023), where a^l and m^l denote the output of SA layer and FFN in transformer layer l, respectively. That is, a transformer block l combines a token h_i^{l-1} from a previous layer with the output of SA layer a_i^l. It is notable that a_i^l allows a token h_i^{l-1} to understand the context in the preceding texts (Fig. 1B).

2.2.1 FFNs

Recent studies have suggested that FFNs can function as associative memory modules that can accept a key and return a value(Geva et al., 2021; Meng et al., 2022; 2023). For instance, when a FFN accepts the 'Space Needle' as an input (i.e., a key), it returns 'Seattle' as an output (i.e., a value) (Meng et al., 2022). We note that human words often convey multiple meanings in different contexts, suggesting that FFNs need to store multiple values for a single key and choose the right one accordingly. For instance, if a LLM is asked to explain the function of the Space Needle, it should recall that it is an observatory and retrieve the 'observatory' instead of 'Seattle'.

Then, how does a FFN store multiple keys and choose the right one according to the context evaluated by a SA layer? FFNs in LLMs consist of two synaptic layers, ($\boldsymbol{w}^{1st}$ and $\boldsymbol{w}^{2nd}$), and one hidden layer of hidden cells m_i (Eq. 1). As hidden cells m_is operate independently, we consider the input $h_i^{l-1} + a_i^l$ and output v_i^l of a single neuron m_i

$$\begin{aligned} x_i &= w_F^i \cdot (h_j^{l-1} + a_i^l), \\ v_i^l &= w_S g(x_i), \end{aligned} \tag{1}$$

where w_F and w_S denote a row of $\boldsymbol{w}^{1st}$ and a column of $\boldsymbol{w}^{2nd}$, respectively, and $g(x)$ is the activation function, which is commonly the ReLU or one of its variants. From this, it is clear that m_i becomes active only when its input is aligned with w_F^i. Geva et al. (2021) suggested that hidden neurons m_i in a FFN were indeed memory cells. In this case, w_F^i can be considered a feature. More importantly, this observation raises the possibility that a FFN can encode semantic meanings of human words with lower concepts (i.e., features).

In the example above, three memory cells may be sufficient for a FFN to selectively recall Seattle or the observatory. Let's consider a case, in which three memory cells m_0, m_1 and m_2 of a FFN are configured to detect 'Space Needle', 'geography' and 'facility's function'. When m_0 and m_1 are activated together by $h_i^{l-1} + a_i^l$, its output v^l will be determined by the outputs of m_0 and m_1, $v^l = \alpha w_S^0 + \beta w_S^1$. When m_0 and m_2 are activated together, its output $\tilde{v}^l$ will be determined by the outputs of m_0 and m_2, $\tilde{v}^l = \alpha w_S^0 + \gamma w_S^2$. If a LLM can map v^l and $\tilde{v}^l$ onto the token of Seattle and that of the observatory, it can selectively recall either Seattle or the observatory.

As the Space Needle can be defined as its function or location, most words, if not all, can be defined with lower-level concepts. For example, a motor vehicle can be described with multiple features such as the number of its wheels and the inclusion of a cargo bed. This line of reasoning leads us to hypothesize that LLMs may be trained to use a large and unknown set of context vectors c_ks (e.g., geography or the facility's function in the example above) to encode semantic meanings of human words that are hierarchically organized, and these contexts can be stored and utilized by FFNs.

According to this hypothesis, single entities/words can be encoded by a finite number of context vectors c_ks. In other words, they are encoded in subspaces spanned by c_ks.

2.2.2 ICL AS A VECTOR OPERATION

In this study, we treat an ICL task as a function that accepts a query token ($\vec{q}$) and a separator token ($\vec{s}$) and returns an answer token ($\vec{a}$) (Eq. 2).

$$\vec{a} = f(\vec{q} + \vec{s}) \tag{2}$$

Inspired by the observation that the embedding of the word 'king' is converted to that of the word 'queen' by linear operations in an embedding space, we hypothesize that the linear operations can also implement a necessary function f in a subspace (Eq. 3).

$$\vec{a} = \alpha\vec{q} + \beta\vec{s} + \gamma \tag{3}$$

, where γ is an intercept (i.e., constant).

To address this hypothesis, we conduct the linear regression analysis with 30 principal components using scikit-learn (Pedregosa et al., 2011). If $\vec{a}$ is well described by linear function of $\vec{q}$ and $\vec{s}$, the

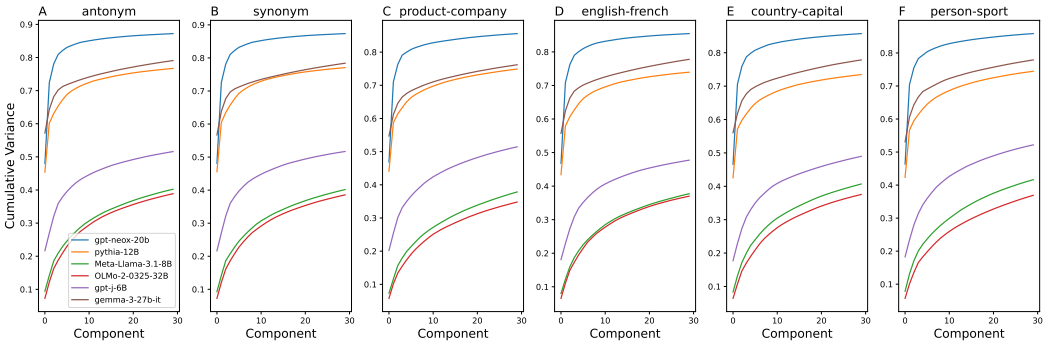


Figure 2: Explained variance accounted for by 30 principal components. x-axis denotes principal components, and y-axis denotes the cumulative explained variance. (A), Explained variance evaluated from LLMs engaging the antonym task. In the panel, 6 models are clarified using different colors (see the inset). (B)-(F), the same as (A), but the task is synonym, country-capital and English-French, country-capital and person-sport, respectively.

quality of linear regression would be high. Otherwise, its quality would be low. Below, we examine the coefficient of determination R^2 to evaluate the quality of the linear regression analysis with 30 principal components (i.e., the potential bases for LLMs' subspaces).

3 Empirical Evaluation

In this study, we probe 6 pretrained LLMs publicly available: GPT-j-6B (Wang & Komatsuzaki, 2021), Meta-Llama-3.1-8B(Grattafiori et al., 2024), OLMO-2-0325-32B(OLMo et al., 2024), Phythia-12B(Biderman et al., 2023), gemma-3-27b-it (Team et al., 2025) and GPT-NEOX-20B(Black et al., 2022). All models are instantiated and tested with the publicly available machine learning libraries, Pytorch (Paszke et al., 2017) and Transformers (Wolf et al., 2020). All experiments and analyses are conducted using two consumer grade workstations. One workstation is equipped with Intel's Core I9 CPU (64GB DRAM) and NVIDIA's GTX 4090 (24GB VRAM), and the other, with AMD's Strix Halo with 128GB unified memory.

3.1 Principal components as potential bases for subspaces

Since residual streams of LLMs store all hidden representations, we analyze them to identify subspaces (i.e., components). Specifically, we create 200 ICL prompts for 6 different tasks (Todd et al., 2024), antonyms, synonyms, countries-their capital cities (country-capital), English-French words, product-company and person-sport. Each prompt contains 5 examples and 1 question, and queries and answers are separated by separators (':'); see Fig. 1A. In the experiment, we collect residual streams of all tokens from all layers, while LLMs process ICL prompts. After collecting residual streams, we convert them into $2D$ arrays. Each row of the arrays is a residual stream, and thus the size of the row vectors is the same as the model dimension d. For instance, the model dimension d of Meta-LLama-3.1-8B is 4096. The number of rows is the same as the total number of tokens. As some words are tokenized into multiple tokens, the exact number of tokens vary from one model to another. As the residual streams are not bounded, each row is normalized to have a unit norm $\vec{s} \leftarrow N(\vec{s})$. Additionally, all columns are centered $\vec{s} \leftarrow S(\vec{s})$, as it is required for PCA.

Next, 30 principal components are evaluated using scikit-learn, an open-source machine learning library(Pedregosa et al., 2011). Fig. 2 shows the accumulated explained variance for 30 principal components. As shown in the figure, $40 - 90\%$ of variability is accounted for by just 30 principal components, although $d \approx 5000$, suggesting that LLMs can be mapped onto low-dimensional subspace.

Table 1: Comparison of R^2 with the baselines. We compares the mean values (V_1) of top-3 from our original experiments and those (V_2) from the control experiments. For each model and task, we display (V_1, V_2). That is, the first number in the parenthesis denotes the mean value of top 3 R^2 from the original experiment, whereas the second number denotes the mean value of top 3 R^2 from the control experiment. ANT, SYN, CC, EF, PC, PS stand for antonym, synonym, country-capital, English-French, Product-Campany and Person-Sport, respectively. Neox, Olmo, Py, GPT, Llma and Gamma denote GPT-neox-20B, OLMo-2-0325-32B, pythia-12B Llma-3.1-8B and gemma-3-27b-it, respectively.

Model	ANT	SYN	CC	EF	PC	PS
Neox	(0.97, 0.02)	(0.97, 0.01)	(0.98, 0.01)	(0.97, 0.01)	(0.96, 0.01)	(0.99, 0.02)
Olmo	(0.86, 0.01)	(0.8, 0.01)	(0.89, 0.01)	(0.9, 0.01)	(0.85, 0.02)	(0.95, 0.02)
Pyt	(0.96, 0.01)	(0.97, 0.01)	(0.99, 0.02)	(0.97, 0.01)	(0.98, 0.01)	(0.99, 0.01)
GPT	(0.97, 0.01)	(0.95, 0.01)	(0.98, 0.01)	(0.96, 0.01)	(0.96, 0.01)	(0.97, 0.01)
Llma	(0.94, 0.01)	(0.81, 0.01)	(0.96, 0.01)	(0.88, 0.01)	(0.85, 0.01)	(0.97, 0.02)
Gemma	(0.91, 0.01)	(0.91, 0.01)	(0.92, 0.01)	(0.65, 0.01)	(0.89, 0.01)	(0.94, 0.02)

3.2 ICL TASKS TRANSLATED AS VECTOR ALGEBRA PROBLEMS IN SUBSPACES

Although 30 principal components can explain the majority of residual streams' variances, it cannot guarantee that LLMs would use them to support ICL. Thus, to further test whether LLMs actively utilize subspaces, we ask if the answer token $\vec{a}$ can be a linear combination of query $\vec{q}$ and separator $\vec{s}$ tokens. When the words are tokenized into multiple tokens, only the last tokens of query and answer words are used in this analysis, which is consistent with the earlier studies (Todd et al., 2024; Yang et al., 2024; Meng et al., 2023). Specifically, for a transformer layer l and a principal component PC_i, we project answers, queries and separators in 1000 examples (5 in-context examples in 200 prompts) to obtain $\vec{a}_i^l$, $\vec{q}_i^l$, $\vec{s}_i^l$ and regress $\vec{a}_i^l$ into $\vec{q}_i^l$, $\vec{s}_i^l$ (Eq. 4).

$$\vec{a}_i^l = \alpha \vec{q}_i^l + \beta \vec{s}_i^l + \gamma_i^l, \tag{4}$$

where $\vec{a}_i^l$, $\vec{q}_i^l$, $\vec{s}_i^l$ denote 1-d vectors whose length is 1000 (5 in-context examples in 200 prompts).

To evaluate the quality of regression and identify a subspace, where the answer token can be described by the linear combination of query and separator tokens, we use R^2. When the regression is perfect, $R^2 = 1$. By contrast, R^2 approaches 0, when the regression fails. That is, if R^2 is close to 1, the principal component can be a basis for a subspace that we aim to identify. Fig. 3 shows R^2 collected from all layers of 6 models engaging in the 'antonym' task. Across all 6 models, R^2 is high along with a few components. That is, answer tokens are well approximated by linear combinations of query and separator tokens (Eq. 4) in a subspace spanned by these components, supporting that LLMs do use subspaces and that ICL tasks can be solved via simple vector operations. We made two more observations. First, R^2 is low in early layers and becomes higher in late layers, raising the possibility that the early layers of LLMs gradually transform LLM's activation spaces into subspaces, where ICL can be solved by simple vector operations. Second, the principal components with high R^2 appear to be largely consistent across the layers. For instance, R^2 is consistently high along component 9 of GPT-NEOX-20B (Fig. 3A).

These observations suggest that the components with high R^2 encode the concept 'antonym', which is consistent with linear representation hypothesis (Park et al., 2023) but further suggests that 1) there are multiple directions associated with concepts and 2) actual computations associated with concepts occur according to these directions. We repeat the same analysis with 5 more tasks and observe the equivalent results with different tasks, 'country-capital' (Fig. 4), 'English-French' (Supplementary Fig. 2), 'synonym' (Supplementary Fig. 3), 'product-company' (Supplementary Fig. 4) and 'person-sport' (Supplementary Fig. 5).

3.3 TOKEN TYPES IN SUBSPACES

If LLMs do utilize subspaces and vector operations in it, one can expect that tokens are clustered together depending on types. We test this possibility by visualizing answer, separator and query tokens in $3D$ subspace spanned by the top 3 principal components according to R^2. In this analysis,

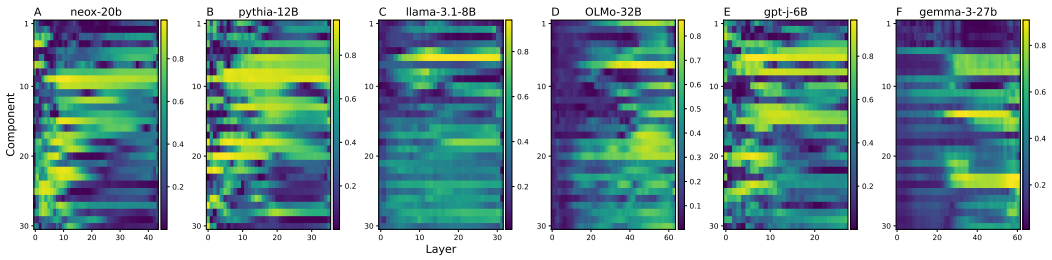


Figure 3: Quality of linear regression evaluated. R^2 is evaluated using LLMs engaging in 'antonym' task. x-axis denotes the transformer layer, and y-axis denotes the principal components. (A)-(F) shows R^2 estimated from GPT-j-6B, Meta-Llama-3.1-8B, OLMo-2-0325-32B, Phythia-12B, gemma-3-27b-it and GPT-NEOX-20B.

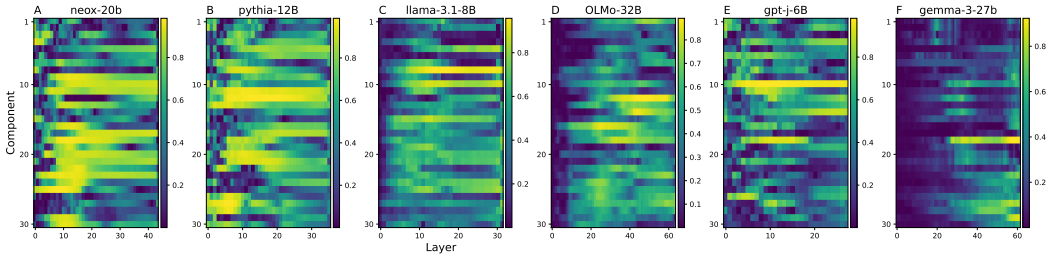


Figure 4: The same as Fig. 3, but the task is country-capital.

we choose a single layer with the highest R^2 from individual models engaging in antonym (Fig. 5), country-capital (Fig. 6), English-French (Supplementary Fig. 6), synonym (Supplementary Fig. 7), product-company (Supplementary Fig. 8) and person-sport (Supplementary Fig. 9). As shown in the figures, answer, separator and query tokens are well clustered with the same token type, and the token types are separated from other types. These results suggest that tokens encode task specific information (i.e., query, separator or answer), supporting that the subspace is associated with LLMs' decision making.

3.4 Control experiments

Finally, to test if R^2s are statistically meaningful and higher than random baselines, we conduct control experiments by shuffling $\vec{a}_i^l$. As $\vec{a}_i^l$ is randomized, the linear regression analysis tries to predict answer tokens using incorrect queries and separator tokens, and thus R^2 can serve as the baseline of R^2 for stochastic noises. To compare R^2 with the baselines, 1) we evaluate top 3 R^2 with the correct and randomized answer tokens from all layers of models and all 30 components, and 2) we report the mean values of these top 3 R^2 in Table 1. As shown in the table, R^2s are close to 0 in the control experiments with randomized answer tokens, whereas R^2 are close to 1 in the original experiments with correct answer tokens. This result confirms that R^2s, which are evaluated using the correct $\vec{a}_i^l$, are not stochastic artifacts. Additionally, we also replace answer tokens $\vec{a}_i^l$ with neutral tokens, which are 'Q', 'A' and line breakers. Then, we replace each answer token with one of the neutral tokens collected from the same input prompt to obtain R^2 and find that R^2 is close to 0 in this additional control experiment (Supplementary Table 1).

3.5 Causal links between subspaces and ICL tasks

The results above suggest that LLMs can use some subspaces to support ICL, which leads us to assume that perturbation on subspaces would have substantial effects on LLMs' predictions. With this possibility in mind, we test our hypothesis further by perturbing subspaces. Specifically, we edit the last separator tokens $\vec{s}$ in the subspaces spanned by top-3 and bottom-3 components. We choose top-3 components with the highest R^2 and bottom-3 components with the lowest R^2. Following Alg.

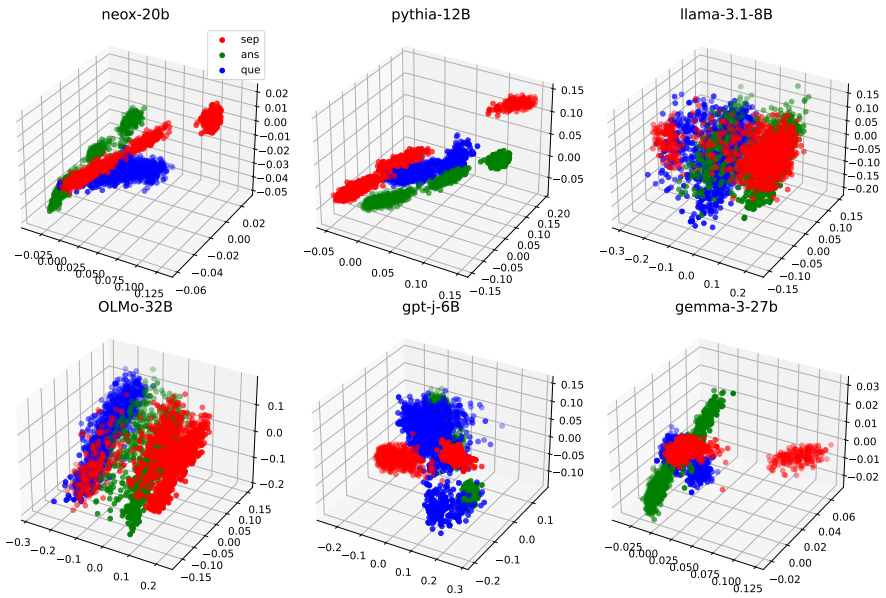


Figure 5: Query (que), separator (sep) and answer (ans) tokens in subspace spanned by 3 principal components with the highest R^2s. Red, green and blue dots represent separator, answer and query tokens, respectively (see the inset). All tokens are collected from 6 LLMs engaging in the antonym task. The model is specified by the name above each plot.

Table 2: This table shows how often LLMs change their answers when the last token is perturbed via top-3 and bottom-3 components. For each case, the first number in the parenthesis shows the ratio of the trials, in which LLMs change their answers when the perturbation is delivered via top-3 components. The second number shows the ratio of the trials, in which LLMs change their answers when perturbation is delivered via bottom-3 components. PS, CC and PC stand for Person-Sport, Country-Capital and Product-Company, respectively.

Model	PS	CC	PC
pythia-12B	(1.0, 0.1)	(1.0, 0.25)	(1.0, 0.16)
gpt-neox-20b	(1.0, 0.09)	(0.96, 0.08)	(1.0, 0.18)
gpt-j-6B	(0.98, 0.12)	(0.99, 0.01)	(1.0, 0.08)
gemma-3-27b-it	(0.7, 0.02)	(0.72, 0.0)	(0.09, 0.06)

1, we evaluate the perturbed $\vec{s}_{top}$ and $\vec{s}_{bottom}$ for the top-3 components and the bottom-3 components. It should be noted that perturbation δ_{bottom} and δ_{top} have the same magnitude.

Using both δ_{top} and δ_{bottom}, we compare the subspace, where ICL tasks are solved using linear vector operations, to the subspace, which is weakly associated with ICL tasks. In this experiment, we test 4 models with 3 tasks using 100 ICL prompts by perturbing the last query $\vec{s}$ in 5 intermediate layers (Alg. 1). For the same prompt, we perturb top-3 components and the bottom-3 components independently, and the results are shown in Table 2.

Although we perturb the last separators in 5 layers only, LLMs change their answers almost always when the separator is perturbed via top-3 components. By contrast, when the separators are perturbed via bottom-3 components, LLMs change their answers quite rarely (Table 2).These results suggest that the subspace can have causal impacts on LLMs' decision-making.

We additionally test the correlations between subspaces and LLMs' decisions and report the results in A.1.

Algorithm 1 Algorithm used to create perturbation. We perturb 5 intermediate layers: from layer 25 to layer 29 of gemma-3-27b-it and from layer 11 to layer 15 of all other models (pythia-12B, gpt-neox-20b and gpt-j-6B).

1. The last token token $\vec{s}$ is normalized $\vec{s} \leftarrow N(\vec{s})$ and centered $\vec{s} \leftarrow S(\vec{s})$
2. Project $\vec{s}$ to top-3 components V_0, V_1 and V_2 with highest R^2 and obtain coefficients c_0, c_1 and c_2.
3. Estimate perturbation $\delta_{top} = c_0V_0 + c_1V_1 + c_2V_2$.
4. Obtain perturbation for top-3 components, $\vec{s}_{top} \leftarrow \vec{s} + \delta_{top}$.
5. Project the last separator token $\vec{s}$ to bottom-3 components B_0, B_1 and B_2 with lowest R^2 and obtain coefficients b_0, b_1 and b_2.
6. Estimate perturbation $\delta_{bottom} = sign(b_0)|c_0|B_0 + sign(b_1)|c_1|B_1 + sign(b_2)|c_2|B_2$.
7. Obtain Perturbation for bottom-3 components, $\vec{s}_{bottom} \leftarrow \vec{s} + \delta_{bottom}$.
8. Project $\vec{s}_i$ back into token space with $\vec{s} \leftarrow S^{-1}(\vec{s}_i)$ and $\vec{s} \leftarrow N^{-1}(\vec{s}_i)$, where i is either top or bottom.

4 DISCUSSION

Our analyses suggest that tokens in ICL prompts can be encoded in subspaces and that their relations are explained by the linear operations. Below, we discuss the implications of our analyses for ICL capabilities of LLMs, links to the earlier works and limitations.

4.1 SUBSPACE AND ICL

The residual stream of a LLM, which stores activations from all of its layers, allows its last layer to effectively access all subspaces $\{\Omega_i\}$. This raises the possibility that LLMs can encode and retrieve the relevant information regarding entities in the input prompt in parallel. For example, the Space Needle in an input prompt may activate multiple subspaces associated with the Space Needle. Two of them can be the location subspace Ω_1 to recall, which encodes that the Space Needle is located in Seattle, and the facility function subspace Ω_2, which encodes that the Space Needle is an observatory.

It seems natural to assume that the degree of activation strengths of subspaces Ω_1 and Ω_2 would follow the distribution of the training set. If the 'Space Needle' and 'Seattle' appear together 40% of the time in the training set, and the 'Space Needle' and the 'observatory' appear together 20% of the time, Ω_1 is activated two times more strongly than Ω_2. However, if the 'Space Needle' is paired with 'Seattle' in the input prompt, Ω_1 would be more strongly activated.

With this possibility in mind, we propose that the biased competition among subspaces can underlie ICL capabilities of LLMs. The last query in the prompt can activate many subspaces $\{\Omega_i^q\}$, each of which encodes information associated with the query. Each example (i.e., pair of a query and an answer) can activate its own set of subspaces $\{\Omega_i^{e_j}\}$. If a common subspace $\Omega_i^{q_i \cap e_i}$ is jointly activated by the query and the example, this subspace will become strongly activated and likely become a winner which determines the LLMs' answer to the final query. When more examples are given, the common subspaces $\Omega_i^{q_i \cap \{e_i\}}$ are more likely to exist. Using $\Omega_i^{q_i \cap \{e_i\}}$, the LLM can find more accurate answer to the last query.

4.2 RELATED WORK

A line of studies suggests that high level concepts are encoded in deep neural networks (Park et al., 2023; Jiang et al., 2024a; Alain & Bengio, 2016; Kim et al., 2018). Platonic representation hypothesis (Huh et al., 2024) posits that neural networks can converge to a 'shared statistical model of reality in their representation space', even when they have different architectures and are trained independently. Linear representation hypothesis (Park et al., 2024) further proposes that counterfactual concepts are encoded linearly in LLM activation space. However, it remains unclear how concept directions emerge in LLM activation space and whether concepts could be encoded via directions of single vectors (Zhao et al., 2025; Engels et al., 2025).

Our analysis of ICL provides valuable insights into underlying mechanisms by which LLMs generate linear representations and their significance in their learning and generalization capabilities. After

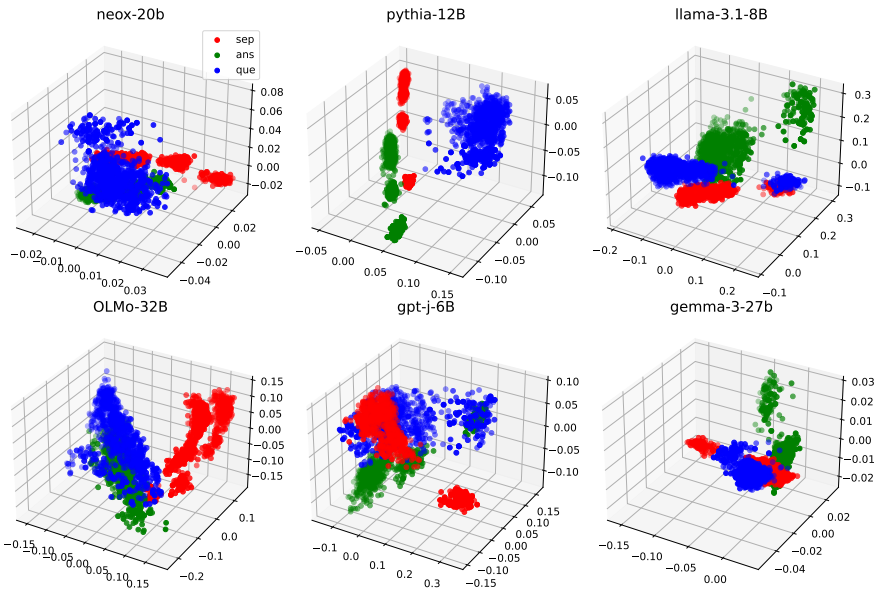


Figure 6: The same as Fig. 5, but the task is country-capital.

LLMs are pretrained to predict next tokens, residual streams may encode potential answers in parallel and thus can be decomposed into a set of components corresponding to potential answers. In our analysis, tokens in the input prompts can be mapped onto the same subspaces (Figs. 5 and 6), and linear operations can solve the desired tasks (Figs. 3 and 4). Although our analysis focuses on ICL, it is based on generic properties of LLMs and thus, can be generalized to other language tasks. If this generalization holds, our analysis raises the possibility that LLMs map information (i.e., tokens) existing in input prompts onto data points in subspaces, whose geometry is directly associated with desired tasks. In this view, a linear representation of a concept emerges naturally in a subset of subspaces, which are functionally crucial to desired tasks. Additionally, we note that LLMs can create a large number of subspaces and track them all in parallel via residual connections, and multi-head attention can create independent subspaces for the same inputs. Together with our analysis, we propose that these subspaces can perform distinct functions and that as a result LLMs can readily implement a wide range of functions after pretraining.

Below, we review two lines of studies on ICL and the potential links between our work and them

First, Xie et al. (Xie et al., 2022) proposed that ICL can be considered as implicit Bayesian inference. Specifically, they noted that LLMs may learn the distribution of latent concepts Θ, and for a $\theta \in \Theta$, LLMs can generate output tokens using latent concept space: $p(O_1, O_2, \ldots, O_T) = p(O_1, O_2, \ldots, O_T|\theta)p(\theta)$. In principle, Bayesian inference rule allows us to predict $p(\theta|O_1, O_2, \ldots, O_T)$ from $p(O_1, O_2, \ldots, O_T|\theta)p(\theta)$. They provided theoretical arguments how to use derive a formal framework on ICL from Bayesian rule. This study, however, does not present how LLMs can recover a latent concept θ from ICL examples. Our analyses suggest that LLMs use subspaces to encode concepts. Conversely, the collection of subspaces can map onto latent concept space Θ.

Second, another theoretical view is based on dual gradient updates (Von Oswald et al., 2023; Dai et al., 2023). Dai et al. (Dai et al., 2023) proposed evidence supporting strong 'ICL-GD (gradient descent) correspondence' in pretrained LLMs by probing the outputs and updates of attention heads. Specifically, they found that the updates and outputs of attention heads are well aligned between ICL and fine-tuning. However, this evidence for ICL-GD correspondence was questioned by a more recent study (Deutch et al., 2024). Although it is still unclear if dual gradient descent underlies ICL, it may explain how LLM generate subspaces, where simple vector operations can solve ICL tasks.

We also discuss the limitations of our study in A.2

AI USE STATEMENT

We have not used generative AI tools in this work.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv*, 2016. URL http://arxiv.org/abs/1610.01644.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling, 2023. URL https://arxiv.org/abs/2304.01373.

Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. Gpt-neox-20b: An open-source autoregressive language model, 2022. URL https://arxiv.org/abs/2204.06745.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. Why can GPT learn in-context? language models secretly perform gradient descent as meta-optimizers. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 4005–4019, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.247. URL https://aclanthology.org/2023.findings-acl.247/.

Gilad Deutch, Nadav Magar, Tomer Natan, and Guy Dar. In-context learning and gradient descent revisited. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1017–1028, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.58. URL https://aclanthology.org/2024.naacl-long.58/.

Joshua Engels, Eric J Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not all language model features are one-dimensionally linear. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=d63a4AM4hb.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5484–5495, 2021.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego

Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim

Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The platonic representation hypothesis. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 20617–20642. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/huh24a.html.

Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In *Forty-first International Conference on Machine Learning*, 2024a. URL https://openreview.net/forum?id=otuTw4Mghk.

Yibo Jiang, Goutham Rajendran, Pradeep Kumar Ravikumar, Bryon Aragam, and Victor Veitch. On the origins of linear representations in large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 21879–21911. PMLR, 21–27 Jul 2024b. URL https://proceedings.mlr.press/v235/jiang24d.html.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2668–2677. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/kim18d.html.

Md Tahmid Rahman Laskar, Sawsan Alqahtani, M Saiful Bari, Mizanur Rahman, Mohammad Abdullah Matin Khan, Haidar Khan, Israt Jahan, Amran Bhuiyan, Chee Wei Tan, Md Rizwan Parvez, Enamul Hoque, Shafiq Joty, and Jimmy Xiangji Huang. A systematic survey and critical review on evaluating large language models: Challenges, limitations, and recommendations. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 13785–13816, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.764. URL https://aclanthology.org/2024.emnlp-main.764/.

Sheng Lu, Irina Bigoulaeva, Rachneet Sachdeva, Harish Tayyar Madabushi, and Iryna Gurevych. Are emergent abilities in large language models just in-context learning? In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5098–5139, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.279. URL https://aclanthology.org/2024.acl-long.279/.

Andrea Matarazzo and Riccardo Torlone. A survey on large language models with some insights on their capabilities and limitations, 2025. URL https://arxiv.org/abs/2501.04040.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems*, 35, 2022.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013a. URL http://arxiv.org/abs/1301.3781.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff (eds.), *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 746–751, Atlanta, Georgia, June 2013b. Association for Computational Linguistics. URL https://aclanthology.org/N13-1090/.

Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models: A survey, 2025. URL https://arxiv.org/abs/2402.06196.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models. *ACM Trans. Intell. Syst. Technol.*, 16(5), August 2025. ISSN 2157-6904. doi: 10.1145/3744746. URL https://doi.org/10.1145/3744746.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Michal Guerquin, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 OLMo 2 Furious, 2024. URL https://arxiv.org/abs/2501.00656.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *Causal Representation Learning Workshop at NeurIPS 2023*, 2023. URL https://openreview.net/forum?id=T0PoOJg8cK.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39643–39666. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/park24c.html.

Adam Paszke, Sam Gross, Soumith Chintala, Edward Chanan, Gregory Yang, Zachary DeVito, Alban Lin, Zeming Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*, 2017.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Eric Todd, Millicent Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AwyxtyMwaG.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Ben Wang and Aran Komatsuzaki. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals,

Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models, 2022. URL https://arxiv.org/abs/2206.07682.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020. URL https://arxiv.org/abs/1910.03771.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=RdJVFCHjUMI.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? In *Association for Computational Linguistics*, 2024. URL https://aclanthology.org/2024.acl-long.550.

Haiyan Zhao, Heng Zhao, Bo Shen, Ali Payani, Fan Yang, and Mengnan Du. Beyond single concept vector: Modeling concept subspace in LLMs with gaussian distribution. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=CvttyK4XzV.

A Appendix

A.1 Correlations between subspaces and LLMs' decisions

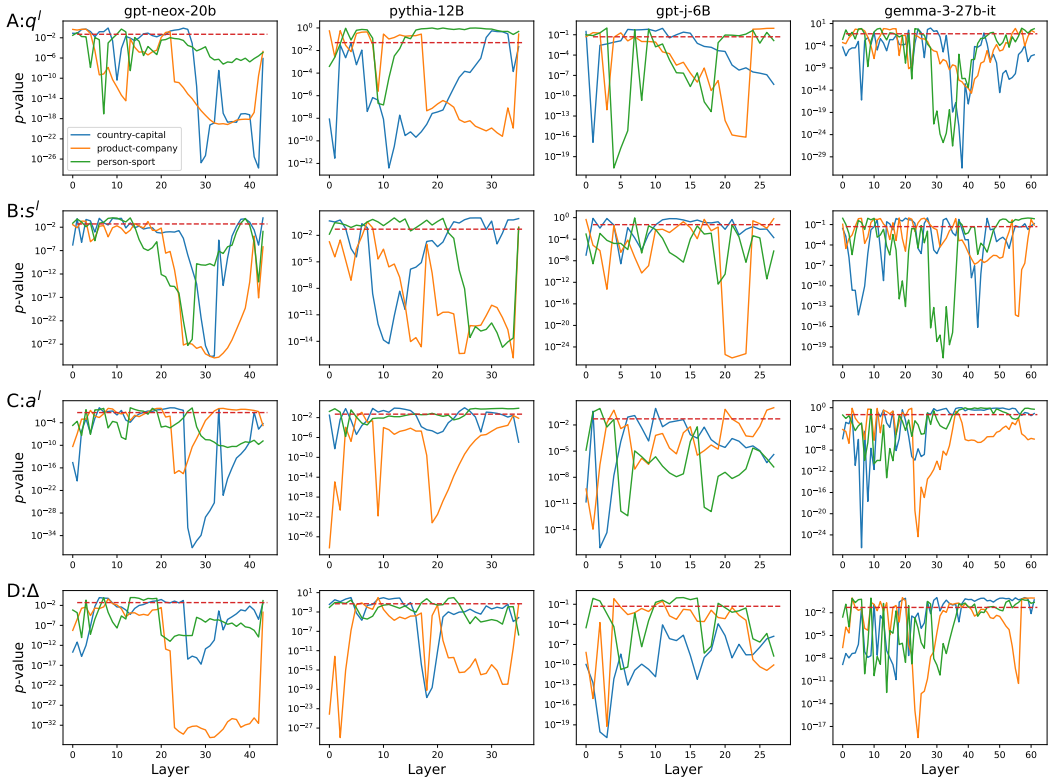
If subspaces are correlated with LLMs' decision-making, one may expect the tokens' projections to be 'significantly' different depending on the correctness of LLMs' answers. Thus, we compare all three tokens (a^l, s^l and q^l) depending on the correctness of LLMs' answers. Specifically, we choose top-1 component in each layer l with highest R^2, project the last set of query, separator and answer tokens to the component and compare them under two conditions, when LLMs make correct predictions and when they make incorrect ones. It should be noted that the last answer token is obtained by allowing LLMs to generate new tokens and that t-test is used to evaluate whether a^l, s^l and q^l are significantly different from each other depending on the correctness of LLMs' answers.

In this experiment, we test 4 models (gpt-neox-20b, pythia-12B, gpt-j-6B and gemma-3-27b-it) and 3 tasks (country-capital, product-company and person-sport) using 400 ICL prompts, different from 200 prompts used to evaluate 30 principal components. Fig. 1 shows the p-value of the t-test on 3 models engaging in 3 tasks. p-value presents the probability that the two distributions are drawn from the same distribution. The two distributions are 'significantly' different when $p \leq 0.05$.

As shown in the figure, the p-values are below 0.05 in multiple layers, suggesting that the projections of all three token types along with the top-1 component are significantly different between two conditions, when LLMs make correct predictions and when they make incorrect predictions. Conversely, the computations occurring along with the top-1 component may be correlated with LLMs' answers. Further, we also compare the error $\Delta = a^l_{j=last} - \alpha q^l - \beta s^l - \gamma^l$ depending on the accuracy of LLMs and find the same trend that Δ is also significantly (t-test, $p < 0.05$) different from one another in multiple layers (Fig. 1D).

We note that p-values vary depending on the tasks and the models, but they are consistently below 0.05 in most of the intermediate and late layers ($layer \geq 20$), suggesting that the intermediate layers can use linear algebra to infer proper predictions. Interestingly, p-value become closer to 0.05 when a layer is close to output (i.e., final) layer. Currently, it remains unclear why the late layers close to output layers become less sensitive to the correctness of LLMs' predictions. A potential reason is that the final layers may integrate evidence across examples rather than processing single examples. If this is the case, tokens (and thus, Δ) will encode information from all examples and become less sensitive to vector operations among tokens in a single example.

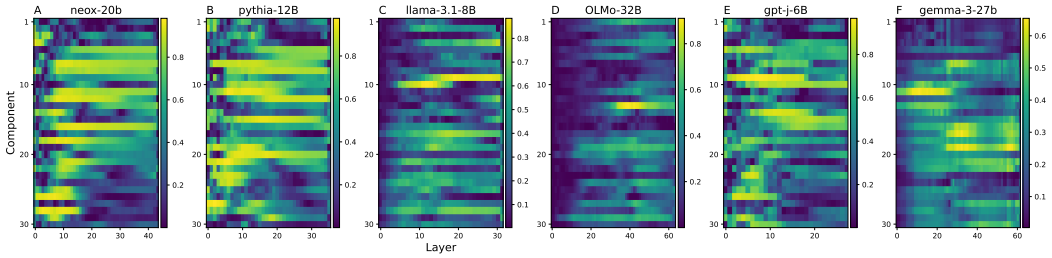
Although this analysis is based on a single component, our results raise the possibility that the subspace could be correlated with LLMs' decision-making.



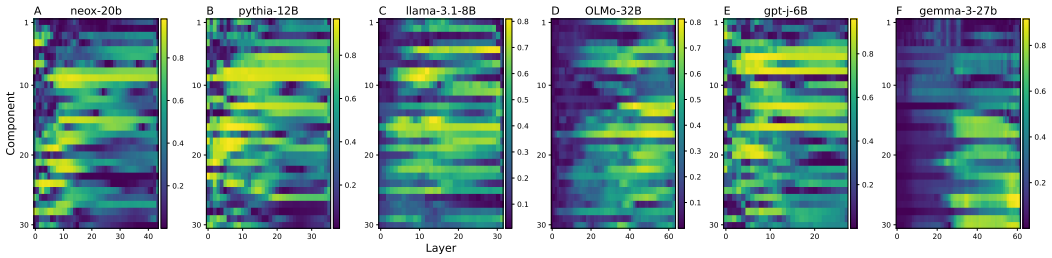
Supplementary Figure 1: Comparison of query q^l in Panel (A), separator s^l in Panel (B), answer a^l in Panel (C), and Δ in Panel (D) when LLMs make the correct predictions and when they make incorrect ones. In this experiment, we use t-test and report p-values. The blue, orange and green lines denote country-capital task, product-company task and person-sport task, respectively.

A.2 LIMITATIONS

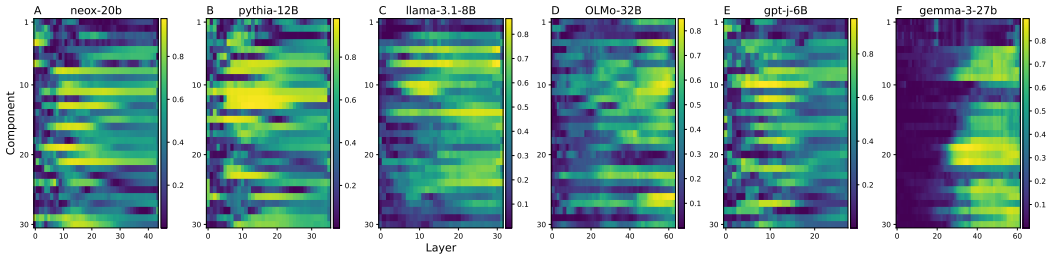
We acknowledge the following limitations of our study. First, in Section 2, we analyze the transformers' functional modules (self-attention and FFNs) under an ideal condition, which does not generally hold. We use this abstract model to gain insights into the mechanisms by which the transformers accumulate evidence from examples in ICL prompts. Second, we use ICL tasks as proxy models of language tasks. We plan to extend our approach to more general language tasks but assume that it will be challenging to identify subspaces in general tasks because tokens interact with one another in complex ways.



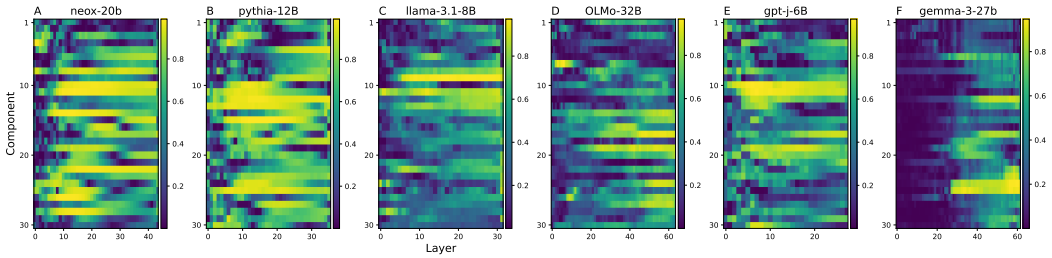
Supplementary Figure 2: The same as Fig. 3, but the task is the English-French.



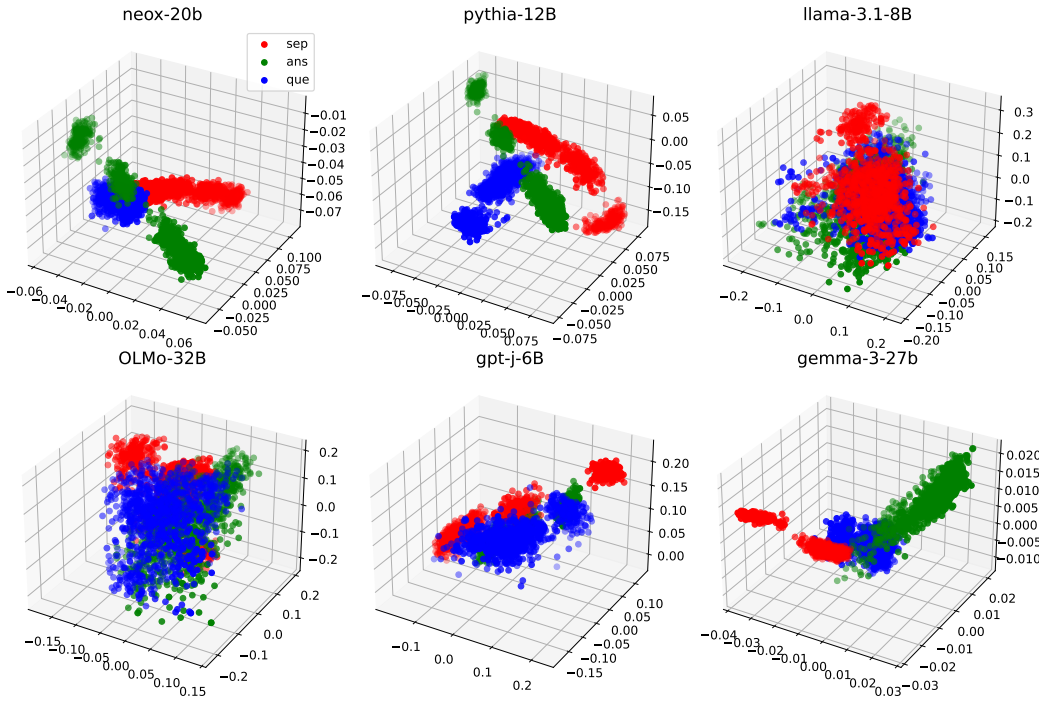
Supplementary Figure 3: The same as Fig. 3, but the task is synonym.



Supplementary Figure 4: The same as Fig. 3, but the task is Product-Company.



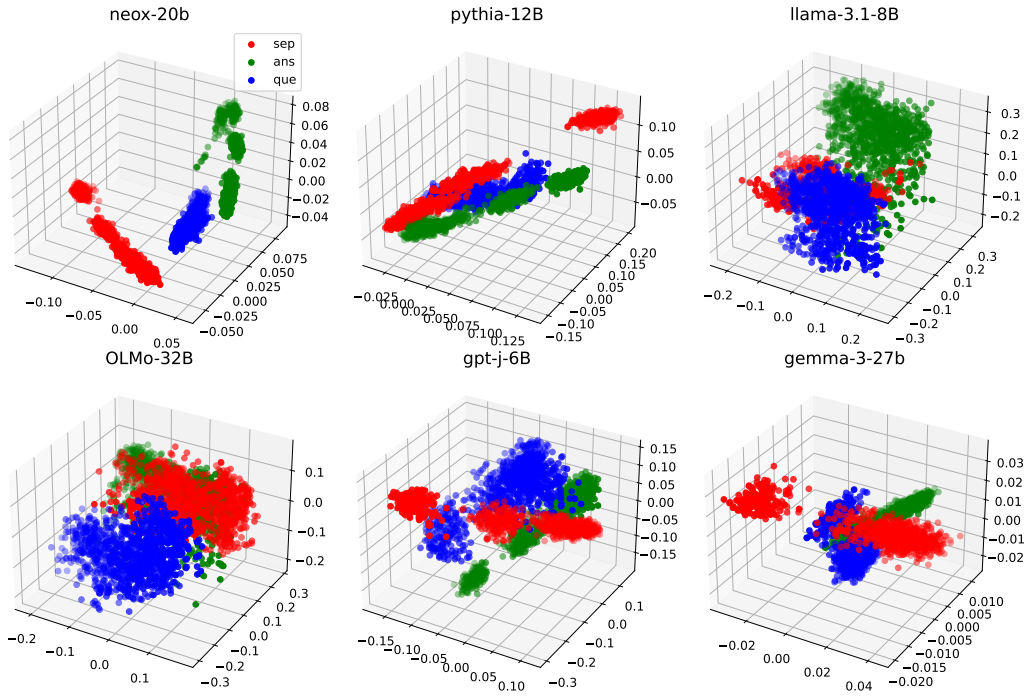
Supplementary Figure 5: The same as Fig. 3, but the task is Person-Sport.



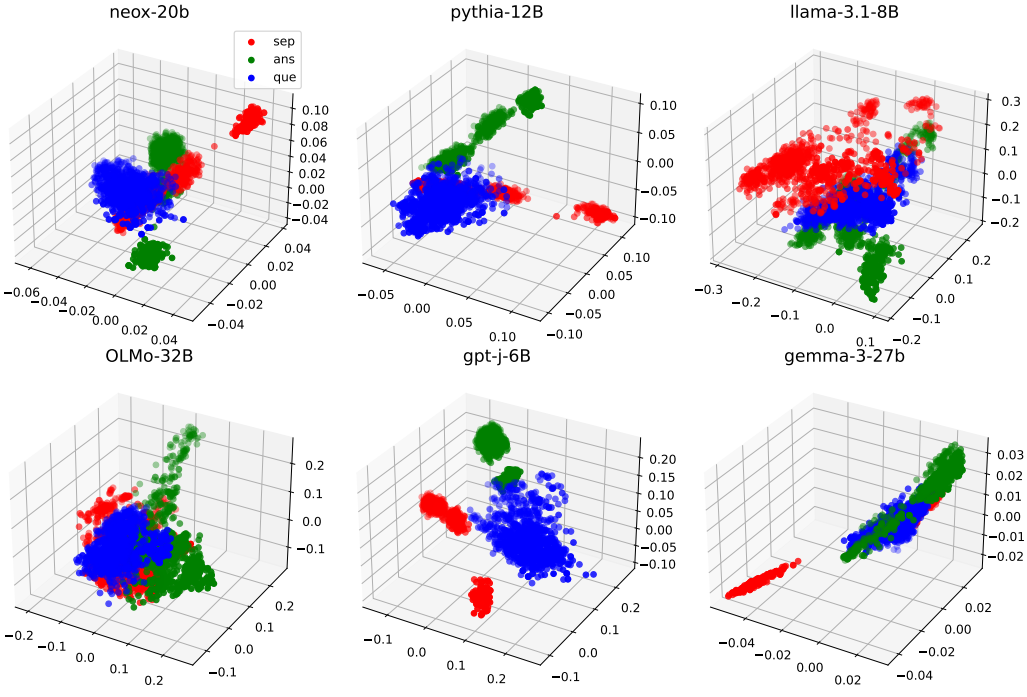
Supplementary Figure 6: The same as Fig. 5, but the task is English-French.

Supplementary Table 1: We compare the mean values (V_1) of top-3 R^2 from our original experiments and those (V_2) from the control experiments, in which an answer token is replaced with a randomly chosen neutral token. For each model and task, we display (V_1,V_2). That is, the first number in the parenthesis denotes the mean value of top-3 from the original experiments, whereas the second number denotes those from the control experiments. ANT, SYN, CC, EF, PC, PS stand for antonym, synonym, country-capital, English-French, Product-Campany and Person-Sport, respectively.Neox, Olmo, Pyt, GPT, Llama and Gamma denote GPT-neox-20B, OLMo-2-0325-32B, pythia-12B Llma-3.1-8B and gemma-3-27b-it, respectively.

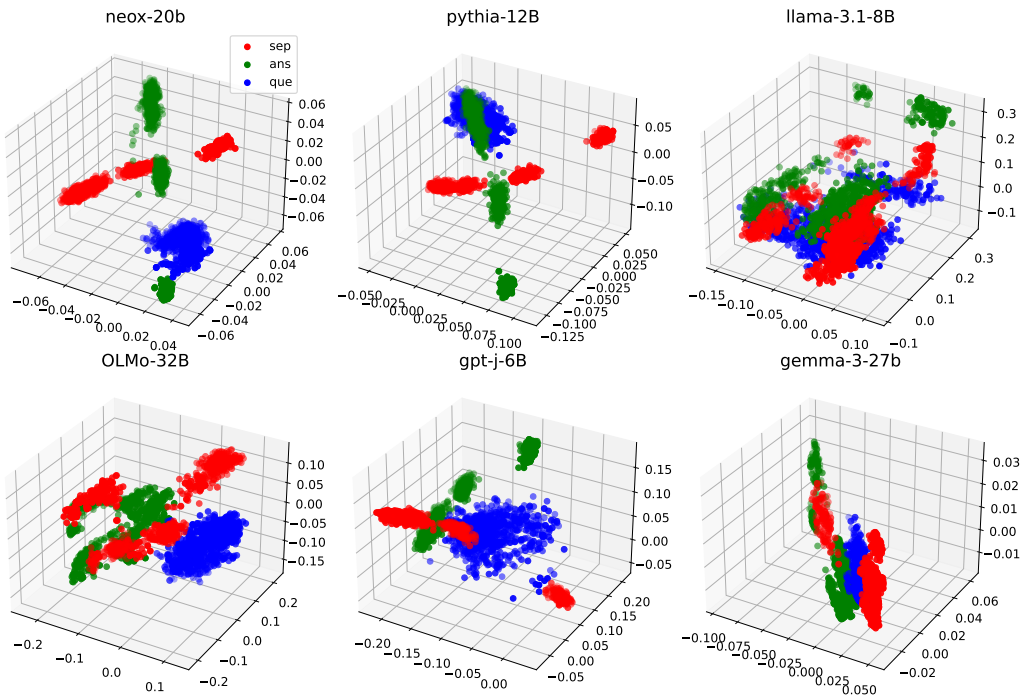
Model	ANT	SYN	CC	EF	PC	PS
Neox	(0.97, 0.04)	(0.97, 0.04)	(0.98, 0.05)	(0.97, 0.04)	(0.96, 0.08)	(0.99, 0.06)
Olmo	(0.86, 0.05)	(0.8, 0.03)	(0.89, 0.02)	(0.9, 0.08)	(0.85, 0.02)	(0.95, 0.02)
Pyt	(0.96, 0.03)	(0.97, 0.04)	(0.99, 0.03)	(0.97, 0.02)	(0.98, 0.09)	(0.99, 0.02)
GPT	(0.97, 0.06)	(0.95, 0.04)	(0.98, 0.03)	(0.96, 0.03)	(0.96, 0.09)	(0.97, 0.05)
Llama	(0.94, 0.06)	(0.81, 0.04)	(0.96, 0.02)	(0.88, 0.03)	(0.85, 0.05)	(0.97, 0.03)
Genna	(0.91, 0.03)	(0.91, 0.03)	(0.92, 0.02)	(0.65, 0.02)	(0.89, 0.04)	(0.94, 0.02)



Supplementary Figure 7: The same as Fig. 5, but the task is synonym.



Supplementary Figure 8: The same as Fig. Fig. 5, but the task is product-company.



Supplementary Figure 9: The same as Fig. Fig. 5, but the task is person-sport.