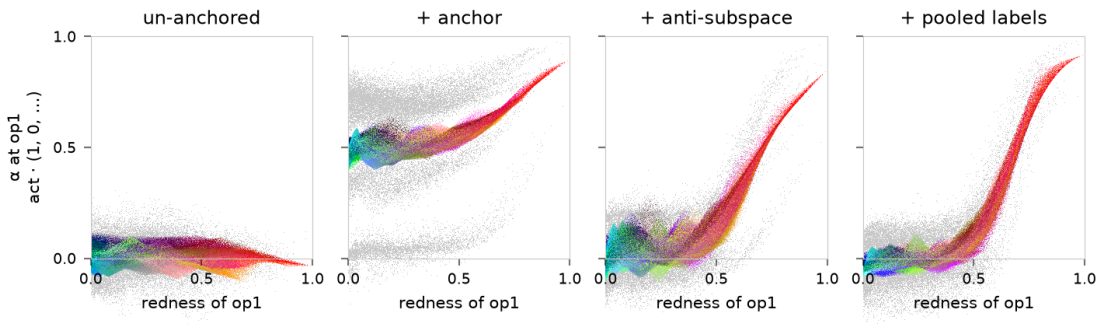
D2.1: figures for the anchoring post

Grading-cloud figures for the LessWrong post on D2.1, *Anchoring concepts in transformers*. This page is not an experiment. Every panel is drawn from the published results of ex-2.1.6, ex-2.1.8, and ex-2.1.10, which hold the preregistered claims and their evidence. The post uploads the light `_assets/*.png` files from the bundle of this page.

The recipe, one piece at a time

Each panel below adds one piece of the anchoring recipe. The quantity plotted is the α response at the op1 token, drawn against the redness of the first operand.[1] Each chart is a *grading cloud*: the response is measured at the 216 vocabulary colors, interpolated across the whole RGB cube, and drawn as a dither of the colors themselves, so the vertical spread at each redness is the range of responses among equally red colors. The grey band behind each cloud drops the averaging: it redraws the response separately for every residual slice and every seed, flattened to grey so it reads as an envelope rather than data.



The anchoring recipe, one piece at a time. *Each panel is a grading cloud of the α response at op1 (mean over the five residual slices) against the redness of the first operand, drawn over the whole RGB cube; the grey envelope behind it redraws the response separately for every residual slice and seed, so the cloud reads against the spread it summarizes. Left to right: the un-anchored control and the bare anchor at λ = 0.1 (ex-2.1.6); the anchor plus the anti-subspace term at its tuned operating point, with labels still keyed to op1 (ex-2.1.8,* `end90-hold30`*); and the full recipe, where sequence-level labels name either operand and mellowmax pooling finds the red token on its own (the primary condition of ex-2.1.10,* `either-t100`*).*

The bare anchor aligns everything at once: the response rises for every color, and even the least red colors sit at $\alpha \approx 0.39$. Adding the anti-subspace term is what makes the alignment graded, pulling non-red colors back to zero while pure red stays anchored. That grading survives the move to labels that no longer point at a particular token: in the primary condition, the slice-mean response at op1 tracks the $\text{sim}^{1.5}$ affinity shape with $r^2 \approx 0.82$.

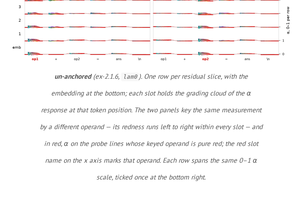
Where the response sits, per (slice, position)

The clouds above average over the residual slices and look only at op1. Splitting each condition by slice and by token position shows where in the network, and where in the sequence, its response sits.

Each condition gets a pair of these, side by side, because the measurement affords two views. The probe set runs every color as op1 against each of the 27 partners it mixes cleanly with, and mixing is symmetric, so every color also sits at op2 on 27 lines. Averaging each line over op2 keys the α map onto the first operand's color; averaging over op1 keys it onto the second. Both are margins of one per-line array, in the statistical sense — each drops an operand by averaging over it — so the colors are re-sorted rather than re-measured, and either way a slot's response is a mean over 27 lines.[2]

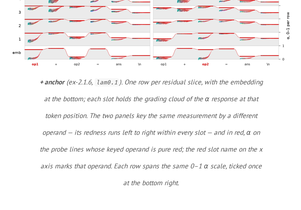
Read the off-key slot with care. Wherever a condition responds at all, the slot that is *not* the key tilts upward too, and part of that tilt is the shape of the probe set rather than anything the model does. A pair is on the probe set only if the mix lands back on the color grid, and that closure rule correlates a color's redness with its partners': pure red's partners average 0.36 redness against 0.25 for the palette as a whole. So an off-key slot shows what a color's *partners* do, and only the on-key slot is a response to the key. The embedding is the clean demonstration — there every state is a function of its own token, so whatever tilt survives in the off-key slot at `emb` is all bookkeeping. The op1 slot repeats that at every slice, because attention is causal: the state above the first token never sees the second operand, so its tilt in the op2-keyed view is partner composition all the way up the stack. The tilt scales with how much response there is to compose from, which is why the un-anchored panels show none of it. Fitted channel probes make the same point without the anchor axis: ex-2.1.12 probes each condition's own color readout and finds no usable op2 information before op2 in any of them, anchored or not.

un-anchored



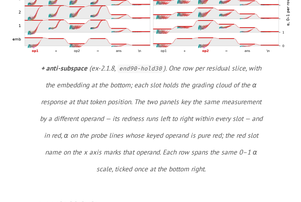
un-anchored *(ex-2.1.6, `1am0`). One row per residual slice, with the embedding at the bottom; each slot holds the grading cloud of the α response at that token position. The two panels key the same measurement by a different operand — its redness runs left to right within every slot — and in red, α on the probe lines whose keyed operand is pure red; the red slot name on the x axis marks that operand. Each row spans the same 0–1 α scale, ticked once at the bottom right.*

+ anchor



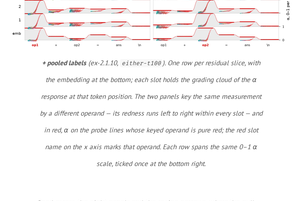
+ anchor *(ex-2.1.6, `1am0_1`). One row per residual slice, with the embedding at the bottom; each slot holds the grading cloud of the α response at that token position. The two panels key the same measurement by a different operand — its redness runs left to right within every slot — and in red, α on the probe lines whose keyed operand is pure red; the red slot name on the x axis marks that operand. Each row spans the same 0–1 α scale, ticked once at the bottom right.*

+ anti-subspace



+ anti-subspace *(ex-2.1.8, `end9-hl430`). One row per residual slice, with the embedding at the bottom; each slot holds the grading cloud of the α response at that token position. The two panels key the same measurement by a different operand — its redness runs left to right within every slot — and in red, α on the probe lines whose keyed operand is pure red; the red slot name on the x axis marks that operand. Each row spans the same 0–1 α scale, ticked once at the bottom right.*

+ pooled labels



+ pooled labels *(ex-2.1.10, `either-t100`). One row per residual slice, with the embedding at the bottom; each slot holds the grading cloud of the α response at that token position. The two panels key the same measurement by a different operand — its redness runs left to right within every slot — and in red, α on the probe lines whose keyed operand is pure red; the red slot name on the x axis marks that operand. Each row spans the same 0–1 α scale, ticked once at the bottom right.*

Read across the eight panels and the recipe narrows *where* the pull lands, in three steps.

Un-anchored there is nothing to place. No slot in either keying leaves the noise, and the slice-mean response at op1 scores $r^2 \approx 0.02$ against the sin$^{1.3}$ shape.

The bare anchor lifts the whole sequence. With a pure-red first operand, mid-stack α runs 0.96 at op1, 0.97 at the `+` slot, 0.84 at op2 and 0.96 at the `=` slot — and it does not stop at the anchored span, reaching 0.71 at the answer and 0.54 at the newline. Both keyings show the same indiscriminate lift.

The anti-subspace term confines it to the span. Outside the span the response falls away (0.33 at the answer, 0.08 at the newline), and the drift over all colors at the last slice drops from 0.72 to 0.21 — the containment ex-2.1.8 was tuned for. Inside the span the pull is still shared: a pure-red op1 brings the op2 slot to 0.83 and the `=` slot to 0.95, and the op2 slot grades against *op1*'s redness at $r^2 = 0.64$. That is what op1-keyed labels buy: a red token pulls the span it sits in, without the span having to say which token it was.

Pooled labels narrow it to the token. On the primary a pure-red op1 leaves the rest of the span between 0.23 and 0.55, and a pure-red op2 leaves the op1 slot at 0.22. Each operand slot answers to its own color — $r^2 = 0.82$ at op1 and 0.88 at op2, reaching α ≈ 0.97 and 0.99 at pure red. So the pull finds the red token in either slot, grades it the same way there, and leaves its neighbours alone, which is the claim the pooled labels were meant to buy.

Two asymmetries survive on the primary. Op2 carries a little more of the drift at the last slice (0.20 against 0.15 at op1), matching ex-2.1.10's reading that op1 hosts less of it once half the pull budget goes to op2. The answer slot barely moves between the two keyings (r^2 0.72 against 0.73): the answer is the mix, so its redness is a function of both operands and neither keying has much claim on it.

---

1. α is the cosine between the residual stream (the running vector that each layer reads from and writes back to) and the anchor axis, the direction we train the concept toward. The model keeps every residual state unit-norm (it is an nGPT variant), which is why the axis label can write that cosine as a plain dot product. Here it is averaged over the five residual slices, the five points in the network where we read that vector. ↩

2. Ex-2.1.10 published its per-line array, so both of its margins are contractions of what it stored. Ex-2.1.6 and ex-2.1.8 averaged theirs over partners before storing, which is the step that drops op2's identity, so their second margin is recomputed from their published checkpoints by `experiment.py` beside this notebook. It checks the op1 margin it recovers against the one they published before publishing anything. ↩