Ex 2.2.12: what the stream holds on `hue-hsv`, and a small recipe sweep

> Scouting, in two parts. Part 1 asks what the stored ex-2.2.11 models kept of *red* on the one op where the projection did not remove it. Part 2 sweeps a few changes to the recipe, including anchoring *red* to a plane rather than an axis, looking for a setup that removes cleanly on every op. The blind-spot story was wrong, the plane does not help, and nothing in the sweep qualifies, so the re-run keeps the reference recipe. Nothing here is a result; the handover re-run that follows scores the recipe at fresh seeds.

Observations

- Which lines survive on `hue-hsv` — did not hold. The split runs the other way. The lines that keep their answer are the ones whose red operand sits on the red axis, at a kept share of 0.38; the lines on either side of it lose theirs, at 0.06 and 0.07.
- Where the surviving hue is written — did not hold. Removing at the blocks alone takes as much as the full projection, keeping 0.22 against 0.24; removing at the embedding alone leaves more, 0.32. Whatever survives is re-derived inside the blocks.
- Where the op1 drift comes from — unresolved. One of the two groups the prediction compares is empty. On `mix`, an answer at the red dose needs a red operand, so no line is labelled through its answer alone.
- The sweep — no condition qualifies. The plane conditions keep 0.19–0.26 on `hue-hsv`, which is inside the reference band, and every other condition keeps at least as much. Every condition passes the task, margin, lead, contrast, and deficit gates.

The proposal: no condition qualifies as a fix for `hue-hsv`; the re-run keeps the reference recipe, and neither τ condition qualifies.

How to read this report

The measurements and the sweep's conditions were fixed before any run, at commit `007ec39`, and the promotion rule below says in advance what counts as a proposal and what happens if nothing qualifies. Everything after that commit is either observations filled into their sections or exploratory, marked as post hoc. The next preregistered experiment adopts what it needs from here and checks it on models it trains itself.

Why this experiment

The anchored-op experiments that follow will read their results through a training setup that nearly works. Every unexplained leftover in it is a confound waiting for a later experiment, and a blind spot we can name now is one we can design around.

Ex-2.2.11 put *red* on one axis of the residual stream of a small transformer,[1] taught it eleven color operations, and then projected the axis out. On ten of the eleven ops the model then lost *red*: it could no longer answer the lines whose answer needs the hue of the red operand. On `hue-hsv`, the op that takes its hue from the second operand, the model kept about a quarter of those answers, a little over the gate. The rule we froze then said the setup is not adopted.

Part 1 checks a candidate explanation. Picture hue as a clock face with red at twelve. The anchored axis measures how red a color is, and that is the same a little clockwise of red (toward orange) and a little anticlockwise (toward pink). So the one thing the axis cannot hold is which side of red a color sits on.

Every other op reads the red operand through its channels, and the red channel is what the axis holds. `hue-hsv` with red at op2 is the only case that needs the side. If the model keeps the side somewhere off the axis, that is what survives the projection. Red sitting at zero degrees is only a coincidence; a green anchor would have the same blind spot around green.

If that holds, pulling harder will not fix it, since one axis cannot hold a two-sided quantity and a plane can. So part 2 trains a small sweep on the handover setup: *red* anchored to a plane crossed against *red* anchored to an axis, at four and at six blocks, each of the two force factors raised by one step, the plane at the higher anchor weight, and two sharper settings of the pooling temperature τ. We expect the force factors to do nothing for `hue-hsv`; a null there is worth having.

The τ conditions answer a separate question. Ex-2.2.11 left the op1 alignment higher than its references and drifting before the anneal, and the τ conditions and one of the stored-checkpoint measurements are there to make progress on that.

The setup we sweep is the `handover` condition of ex-2.2.11: table A+, the stochastic corpus, the whole-line labeller, the untied readout, and the recipe from ex-2.2.3. Its twenty seeds are the reference condition, and memoization makes that condition free.

Conditions

The two parts are independent: part 2 does not wait on part 1.

Part 1: the stored checkpoints

Part 1 trains nothing. It scores the twenty `handover` checkpoints of m2/ex-2.2.11, with its probe set and its removal lines, under a few more edits than ex-2.2.11 scored. Where a measurement has a comparison, `handover-slot` and `handover-tied` are scored too.

Part 2: the sweep

With the ex-2.2.11 recipe as a base, every condition changes one or two things from the reference.

condition	seeds	what changes	τ	λ_a	anti peak	blocks	home of *red*
`handover`	20	ex-2.2.11's candidate, unchanged	0.1	0.1	2.5×	4	axis
`tau-0.03`	5	mellowmax τ 0.03	0.03	0.1	2.5×	4	axis
`tau-0.01`	5	mellowmax τ 0.01	0.01	0.1	2.5×	4	axis
`lam-0.2`	5	anchor weight doubled	0.1	0.2	2.5×	4	axis
`anti-5`	5	anti-subspace peak doubled	0.1	0.1	5×	4	axis
`L6`	5	six blocks	0.1	0.1	2.5×	6	axis
`plane`	5	red anchored to a plane	0.1	0.1	2.5×	4	plane
`L6-plane`	5	six blocks, red anchored to a plane	0.1	0.1	2.5×	6	plane
`plane-lam-0.2`	5	red anchored to a plane, anchor weight doubled	0.1	0.2	2.5×	4	plane

`handover` is ex-2.2.11's candidate at its twenty seeds, and the other conditions train at its first 5, so every condition pairs with the reference seed for seed.

`tau-0.03` and **`tau-0.01`** sharpen the pool. Mellowmax at τ over the positions of a line is roughly the best position minus τ times the log of the line length, and the pull it returns is a softmax at that τ. So with τ held fixed, a longer label span spreads a little more of the pull onto positions that are not the red operand. The whole-line label made the span longer, so a sharper pool may take $\bar{\alpha}$ at op1 back down.

`lam-0.2` and **`anti-5`** are the force factors: twice the anchor weight, and twice the anti-subspace peak. They are here so that a null can be read: if the plane fixes `hue-hsv` and these do not, the fix was the shape of the home and not its strength.

`L6`, **`plane`**, and **`L6-plane`** with the reference are a two-by-two of depth and subspace. Under the plane conditions *red* is pulled toward the span of e_1 and e_2 rather than toward e_1. Alignment with the plane is the length of a state's projection onto the pair: a cosine too,[2] since every state is unit-norm, but unsigned, so it says how much of the state lies in the plane and nothing about the direction within it. The anchor term pulls that length toward one on the labelled lines, the anti-subspace term is its square over every live position, and the removal projects the whole plane out. Six blocks instead of four is the other capacity factor, in case the HSV ops want more depth; it changes the number of slices too, so every slice-averaged measurement is also reported per slice.

`plane-lam-0.2` is the plane at the doubled anchor weight, in case the plane needs more pull than the axis did. Without it, a null on the force conditions would only say that the axis cannot be pushed harder, and a partial result on `plane` could not be told from a plane pulled too gently.

The measurements

Two things differ from ex-2.2.11 in how the sweep is scored. First, the plane conditions are scored on the plane. An unsigned two-dimensional alignment is higher than a signed one-dimensional one even for a state that has nothing to do with *red*, so every plane measurement is compared with the control checkpoints scored the same way; those are stored and cost nothing to score. Second, every op is scored on the removal lines chosen by hue, as ex-2.2.11 did, with `hue-hsv` split by slot.

Glossary

Removal lines

The red lines whose answer needs the hue of the red operand: some permutation of the channels of that operand moves the true answer far. This is the rule from ex-2.2.11, unchanged. On `hue-hsv` they are the lines with red at op2.

Kept share

How much of its clean accuracy on the removal lines a model keeps after the projection. One means the projection did nothing. Zero means every removal-line answer changed, which is what the gate takes *red* being gone to mean. The model may still land on a near neighbour of the true answer; how far the answers move is a separate question, asked by the answer-cube figures of ex-2.2.10.

Side of red

Whether the red operand sits on the red axis of the cube, leans toward orange, or leans toward pink (G = B, G > B, G < B). The part of the hue the anchored axis cannot hold.

$\bar{\alpha}$ at op1

The mean alignment with the axis over every color at the first operand position. How much the colors that are not red have drifted onto the axis.

Band

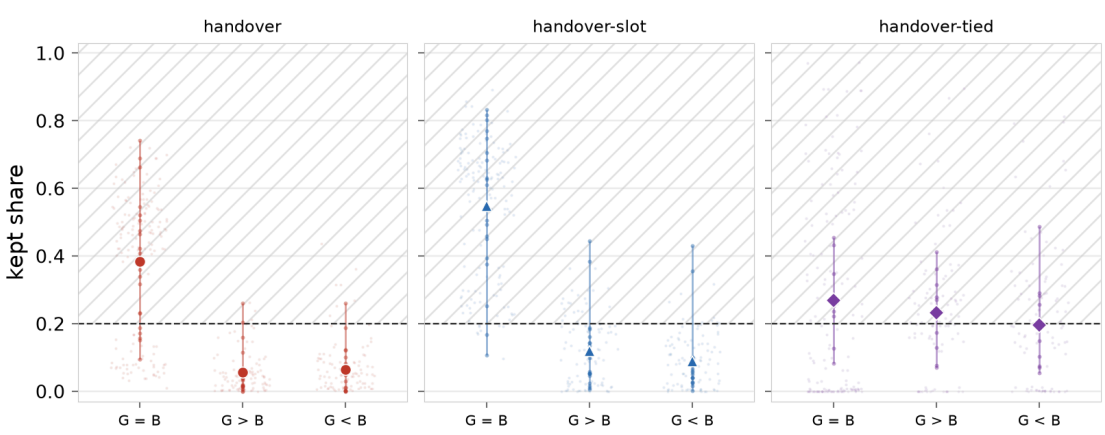
The spread a statistic showed across the twenty seeds of ex-2.2.11. A difference smaller than the band is not resolved.

Which lines survive on `hue-hsv` (part 1)

What we expect. On the `hue-hsv` removal lines, the kept share under the projection splits by the side of red: near zero where the red operand has G = B, and well above zero where G ≠ B. If the kept share is the same on all three sides, the axis is not the reason, and the plane conditions of part 2 lose their motivation; they run regardless, and would then be read as a capacity change with no hypothesis behind it.

Ex-2.2.10 already saw the two sides. Its answer-cube figure for `hue-hsv` with red at op2 shows the projected answers leaving red in two lobes, one toward orange and one toward pink. `sat-hsv` and `value-hsv` with red at op1 fan the same way, so the side survives the projection on those ops too; it just cannot rescue an answer that needs the saturation or value of red.

That figure pools every line; this measurement pairs each answer with the side of its own operand, which is what tells us whether the lobes are the side of red.



Kept share on the `hue-hsv` removal lines, by the side of red. *One panel per condition of ex-2.2.11; within each, one column per side of the red operand. The faint cloud is one dot per removal line, its kept share averaged over the condition's seeds (lines with no clean accuracy to keep are left out). The larger mark is the seed mean of the side's kept share, the ratio of group means ex-2.2.11 gates on, with one small dot per seed on the thin bar spanning the seed range. The dashed line is the 20% gate, hatched above.*

What we saw. The opposite. The lines whose red operand sits on the red axis, with G = B, are the ones that keep their answer, at a kept share of 0.38 on `handover`. The lines on either side of the axis lose theirs: 0.06 toward orange and 0.07 toward pink, both under the gate.

`handover-slot` shows the same shape more strongly, and `handover-tied` more weakly. So the surviving quarter on `hue-hsv` is not the side of red: the side is the one thing the axis cannot hold, and it is the thing the projection removes.

The exploratory section splits the on-axis lines further, by the red operand's color: two of the three on-axis reds survive and the third does not.

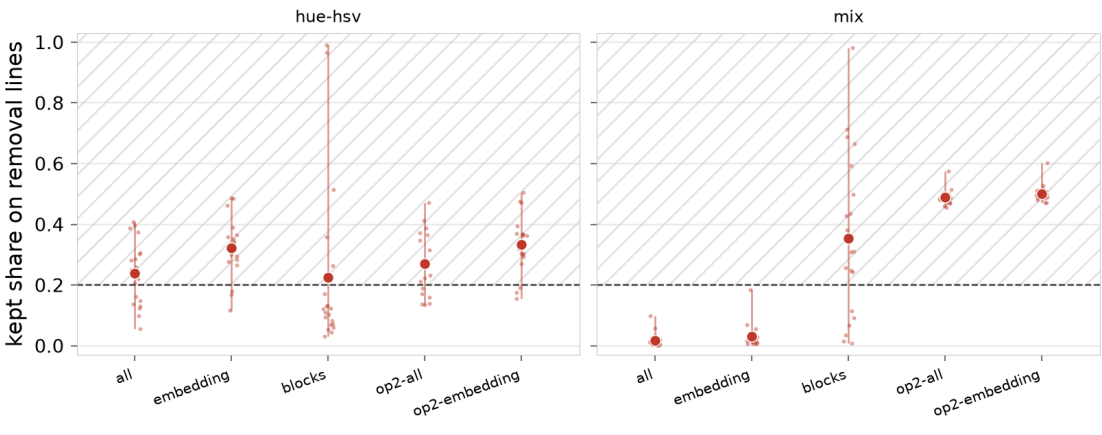
condition	G = B (n)	G > B (n)	G < B (n)	all removal lines
`handover`	0.384 (0.09–0.74) (147)	0.056 (0.00–0.26) (93)	0.065 (0.00–0.26) (93)	0.239 (0.06–0.41)
`handover-slot`	0.549 (0.11–0.83) (147)	0.119 (0.01–0.44) (93)	0.090 (0.00–0.43) (93)	0.349 (0.07–0.54)
`handover-tied`	0.270 (0.08–0.45) (147)	0.232 (0.07–0.41) (93)	0.196 (0.05–0.49) (93)	0.245 (0.08–0.43)

Kept share under `projection` on the `hue-hsv` removal lines by the side of red, seed mean and range, with the line count of each side. The last column is ex-2.2.11's measurement on the same lines.

> **Did not hold**
>
> The kept share is highest where G = B and near zero on both sides of the axis, the reverse of the prediction. The blind spot is not what survives.

Where the surviving hue is written (part 1)

What we expect. With the projection applied at the embedding only, `hue-hsv` keeps about what it keeps under the full projection: the side is written at the embedding, where the red operand's token is, and the blocks read it from there. If removal at the blocks alone takes as much as the full projection, the side is re-derived from the other channels inside the blocks, and the plane at the embedding would not be enough.



Kept share on the removal lines with the projection applied at chosen slices and positions, `handover`. *Left, `hue-hsv`; right, `mix`, the op where the full projection removes cleanly. Along the bottom, the five edits: `all` is ex-2.2.11's operator (every slice, every position), `embedding` and `blocks` split it by slice, and the two `op2` edits apply it at the second operand's position only. Each small dot is one of the twenty seeds, the larger mark the seed mean, the thin bar the seed range. The dashed line is the 20% gate, hatched above.*

What we saw. On `hue-hsv`, removing at the blocks alone keeps 0.22, about what the full projection keeps (0.24), and removing at the embedding alone keeps more, 0.32. The two second-operand edits sit with their whole-line counterparts, so where the projection is applied along the line does not matter, and where it is applied in depth does. The blocks-only edit has about the widest seed range of any measurement in this report: some seeds keep almost everything under it and some almost nothing.

`mix` behaves as ex-2.2.11 would predict, once the position of the edit is held fixed: applied over the whole line, an edit that touches the embedding removes nearly everything, and the blocks-only edit leaves about a third. Applied at the second operand alone it leaves about half, which on this op is the position mattering rather than the depth. That is the pattern the prediction had in mind for `hue-hsv` too, where it is closer to the reverse. The part of the answer that survives the full projection is not written at the embedding; or if it is, the blocks re-derive it from the other channels once the copy at the embedding is gone.

edit	`hue-hsv`	`mix`
`all`	0.239 (0.06–0.41)	0.018 (0.00–0.10)
`embedding`	0.322 (0.12–0.49)	0.032 (0.01–0.19)
`blocks`	0.225 (0.03–0.99)	0.354 (0.01–0.98)
`op2-all`	0.269 (0.13–0.47)	0.489 (0.46–0.57)
`op2-embedding`	0.333 (0.16–0.51)	0.501 (0.47–0.60)

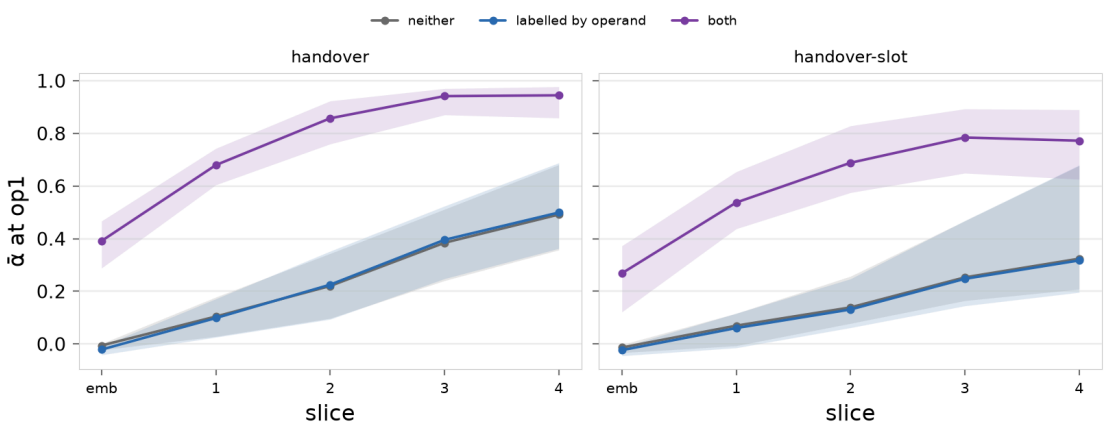
Kept share on the removal lines under each edit, `handover`, seed mean and range over twenty seeds.

> **Did not hold**
>
> Removal at the blocks alone takes as much as the full projection, and removal at the embedding alone takes less. The surviving answers are re-derived inside the blocks.

Where the op1 drift comes from (part 1)

? Unresolved

What we expect. On the red lines, ᾱ at op1 is higher on the lines the whole-line labeller labelled through their answer than on the lines it labelled through an operand. That is the candidate the containment item named for the labeller's half of the rise: a line whose answer draws is pulled at every position, op1 included. If the two groups read the same, the labeller's half of the rise has another cause.



ᾱ at op1 on the `mix` probe lines whose first operand is not red, by which other color could earn the line's label. *One panel per condition, the slices along the bottom (`emb` is the token embedding). Each line is a group's seed mean of the alignment at op1 with the anchored axis, with the seed range as a band: lines that no color labels, lines labelled through a red second operand, and lines with a red second operand and a red answer. Red is at the red dose throughout. A group with no line (labelled through the answer alone) is left out.*

What we saw. The comparison cannot be made on `mix`. The measurement was designed for an op whose answer can be at the red dose while neither operand is, and `mix` is not one: a mean of two colors reaches the red dose only when one of them is red. So the group labelled through the answer alone is empty, the group labelled through both is small (8 lines), and the figure shows two groups where the prediction needed three.

The two groups we do have are alike on both conditions. On `handover`, ᾱ at op1 is about 0.24 for the lines no color labels and 0.24 for the lines with a red second operand. The eight lines with a red second operand and a red answer sit much higher, at 0.76, and are the same on `handover-slot`, whose labeller never reads the answer.

So on this op, what raises ᾱ at op1 is having two red colors on the line rather than which of them earned the label. The containment question needs an op such as `hue-hsv` or `darken`, where the answer can be red without a red operand; the measurement is written to take any op.

group	lines	`handover`	`handover-slot`
neither	5467	0.239 (0.15–0.32)	0.154 (0.11–0.29)
labelled by operand	168	0.239 (0.15–0.32)	0.146 (0.10–0.29)
labelled by answer	0	–	–
both	8	0.763 (0.69–0.81)	0.610 (0.53–0.70)
red op1	189	0.982 (0.98–0.99)	0.968 (0.96–0.97)

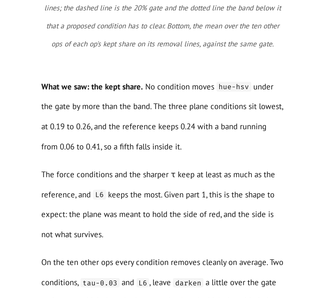
ᾱ at op1 on the `mix` lines by label group, mean over slices, seed mean and range. The last group is the lines with a red first operand, which the containment line pools with the rest.

> **Unresolved**
>
> The answer-labelled group is empty on `mix`, so the measurement cannot say what causes the half of the rise attributed to the labeller. The lines it does have put the rise down to having two red colors on the line, whichever of them earned the label.

The sweep (part 2)

What we expect. The plane conditions (`plane`, `L6-plane`, `plane-lam-0.2`) bring the `hue-hsv` kept share under the gate by more than the band, and the force conditions (`lam-0.2`, `anti-5`) do not move it. τ moves ᾱ at op1 down and nothing else. Depth on its own does little for `hue-hsv`, and the factorial says whether the plane needs it. Every condition keeps the task, the margin, the lead, and the contrast inside ex-2.11's gates; a condition that does not is reported and cannot be proposed.

In every figure of this section, each condition of the sweep is a column of 5 seed dots with its mean, the twenty reference seeds are the band across the panel, and a gate is a dashed line with the failing side hatched. The last column is the un-anchored control scored on the plane, the comparison for the plane conditions.

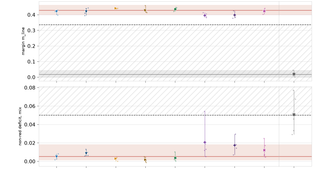


Kept share under the projection, per condition. *Top, on the `hue-hsv` removal lines; the dashed line is the 20% gate and the dotted line the band below it that a proposed condition has to clear. Bottom, the mean over the ten other ops of each op's kept share on its removal lines, against the same band.*

What we saw: the kept share. No condition moves `hue-hsv` under the gate by more than the band. The three plane conditions sit lowest, at 0.19 to 0.26, and the reference keeps 0.24 with a band running from 0.06 to 0.41, so a fifth fails inside it.

The force conditions and the sharper τ keep at least as much as the reference, and `L6` keeps the most. Given part 1, this is the shape to expect: the plane was meant to hold the side of red, and the side is not what survives.

On the ten other ops every condition removes cleanly on average. Two conditions, `tau-0.03` and `L6`, leave `darken` a little over the gate on its own, which the rule does not check and which the table below shows.

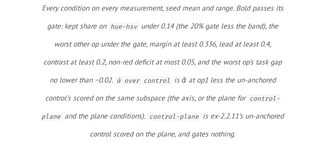


ᾱ at op1 per condition. *The mean alignment with the anchored subspace over every color at the first operand of the `mix` probe lines. The grey band is the un-anchored control scored on the axis; the plane conditions and `control-plane` are scored on the plane, where an unsigned two-dimensional alignment sits higher for every state, so they compare with `control-plane` and the others with the grey band.*

What we saw: ᾱ at op1. The sharper τ lowers it a little: `tau-0.03` reaches 0.251 against 0.264 for the reference, a drop smaller than the band, and `tau-0.01` barely moves it, at 0.254. The force conditions sit at about the same level as `tau-0.03`, which was not predicted and is also inside the band.

The plane conditions read higher, at 0.30 to 0.31, which is the scoring rather than the models: an unsigned two-dimensional alignment is higher for every state, so the comparison that counts is with the control scored on the plane. Against that, their excess is 0.19 to 0.20, below the 0.29 the reference shows over its own control.

We take that to mean the plane conditions drift no more than the axis ones, rather than that they drift less. The two excesses are over different baselines, and a state that drifts the same distance in the stream shows a smaller excess on a plane than on an axis.



The cost side, per condition. *Top, the margin m_line on the `mix` lines, gated at 0.336 (hatched below). Middle, the non-red deficit on `mix` under the projection, gated at 0.05 (hatched above). Bottom, the held-out expected exact match less the control's, on whichever op the condition is worst, gated at −0.02.*

What we saw: the cost side. Every condition passes every gate on its seed mean. The margin is lowest on `plane` and `L6-plane`, at about 0.40 against 0.43 for the reference, and still well over the gate. The non-red deficit on `mix` is under the gate on every seed mean, with one `plane` seed just over it. The task gap is within a hundredth of the control on every condition's worst op, so nothing in the sweep costs the task.

condition	changes	kept hue-hsv ↓	kept, others (worst op) ↓	ᾱ at op1 ↓	ᾱ over control	margin ↑	lead ↑	contrast ↑	non-red deficit ↓	task gap (worst op) ↑
handover	0	0.239 (0.06–0.41)	0.106 (darken 0.18)	0.264 (0.18–0.56)	0.291 (0.21–0.37)	0.429 (0.40–0.46)	0.91 (0.9–0.9)	0.84 (0.8–0.9)	0.005 (0.00–0.02)	−0.004 (value-hsv)
tau-0.03	1	0.221 (0.05–0.32)	0.134 (darken 0.21)	0.251 (0.18–0.27)	0.258 (0.21–0.30)	0.420 (0.40–0.43)	0.92 (0.9–0.9)	0.89 (0.9–0.9)	0.005 (0.00–0.01)	−0.005 (value-hsv)
tau-0.01	1	0.345 (0.19–0.49)	0.111 (sat-hsv 0.18)	0.254 (0.21–0.29)	0.281 (0.25–0.32)	0.420 (0.39–0.44)	0.92 (0.9–0.9)	0.90 (0.9–0.9)	0.008 (0.01–0.01)	−0.006 (-difference)
lam-0.2	1	0.276 (0.12–0.41)	0.083 (darken 0.14)	0.223 (0.21–0.24)	0.250 (0.24–0.27)	0.441 (0.44–0.45)	0.89 (0.9–0.9)	0.85 (0.8–0.9)	0.005 (0.00–0.01)	−0.007 (value-hsv)
anti-5	1	0.316 (0.21–0.39)	0.101 (darken 0.18)	0.213 (0.19–0.25)	0.240 (0.22–0.27)	0.430 (0.41–0.46)	0.88 (0.9–0.9)	0.86 (0.9–0.9)	0.002 (-0.00–0.01)	−0.005 (value-hsv)
L6	1	0.351 (0.21–0.43)	0.133 (darken 0.22)	0.250 (0.20–0.29)	0.277 (0.23–0.32)	0.457 (0.42–0.45)	0.90 (0.9–0.9)	0.86 (0.9–0.9)	0.004 (0.00–0.01)	−0.007 (value-hsv)
plane	1	0.256 (0.09–0.42)	0.093 (lighten 0.16)	0.302 (0.26–0.33)	0.189 (0.15–0.22)	0.396 (0.38–0.41)	0.91 (0.9–0.9)	0.85 (0.8–0.9)	0.021 (0.01–0.05)	−0.008 (-difference)
L6-plane	2	0.204 (0.09–0.35)	0.092 (lighten 0.16)	0.309 (0.27–0.57)	0.196 (0.16–0.26)	0.398 (0.38–0.43)	0.91 (0.9–0.9)	0.85 (0.8–0.9)	0.018 (0.01–0.03)	−0.007 (value-hsv)
plane-lam-0.2	2	0.192 (0.14–0.22)	0.086 (lighten 0.13)	0.305 (0.24–0.57)	0.192 (0.15–0.26)	0.424 (0.40–0.44)	0.90 (0.9–0.9)	0.84 (0.8–0.9)	0.012 (0.00–0.02)	−0.009 (value-hsv)
control-plane	–	0.989 (0.98–1.00)	0.967 (value-hsv 1.00)	0.113 (0.09–0.17)	0.000 (-0.02–0.06)	0.019 (0.00–0.04)	0.20 (0.0–0.4)	0.03 (-0.1–0.1)	0.051 (0.03–0.08)	+0.000 (mix)

Every condition on every measurement, seed mean and range. Bold passes its gate: kept share on `hue-hsv` under 0.14 (the 20% gate less the band), the worst other op under the gate, margin at least 0.336, lead at least 0.4, contrast at least 0.2, non-red deficit at most 0.05, and the worst op's task gap no lower than −0.02. ᾱ `over control` is ᾱ at op1 less the un-anchored control's scored on the same subspace (the axis, or the plane for `control-plane` and the plane conditions). `control-plane` is ex-2.2.11's un-anchored control scored on the plane, and gates nothing.

The proposal

The promotion rule, frozen before the run:

> A condition is proposed as the fix for `hue-hsv` when, at 5 seeds, its kept share on the `hue-hsv` removal lines is under the gate by at least 0.06, its task, margin, lead, contrast, and non-red deficit are inside ex-2.2.11's gates, and its ᾱ at op1 is no higher than the reference's by more than the band. Among several, the one that changes fewest factors is proposed, and ᾱ at op1 breaks a tie. If no condition qualifies, the re-run gates removal on the ten other ops and reports `hue-hsv` beside them as the op where a single axis has a known blind spot; the recipe is then the reference, with a τ change carried only if the τ conditions moved ᾱ at op1 by more than the band at no cost.
>
> Separately, a τ is proposed when its condition lowers ᾱ at op1 by more than 0.06 against the reference with task, margin, lead, and contrast inside their gates. This proposal stands on its own and can be carried with or without a fix for `hue-hsv`.

condition	kept `hue-hsv` ≤ 0.14	gates	ᾱ over control ≤ ref + 0.06	changes	qualifies
`tau-0.03`	0.221	all pass	**+0.258**	1	no
`tau-0.01`	0.343	all pass	**+0.281**	1	no
`lam-0.2`	0.276	all pass	**+0.250**	1	no
`anti-5`	0.316	all pass	**+0.240**	1	no
`L6`	0.351	all pass	**+0.277**	1	no
`plane`	0.256	all pass	**+0.189**	1	no
`L6-plane`	0.204	all pass	**+0.196**	2	no
`plane-lam-0.2`	0.192	all pass	**+0.192**	2	no

The promotion rule, clause by clause, on the seed means. The reference's ᾱ over control is +0.291.

No condition qualifies as a fix for `hue-hsv`; the re-run keeps the reference recipe, and neither τ condition qualifies. Every condition passes the gates and fails the kept clause, so the fallback in the rule applies. The re-run trains the reference recipe, `hue-hsv` stays over the gate, and the next experiment treats it as a known leftover rather than a fix in waiting. The τ conditions lower ᾱ at op1 by less than the band, so the pooling temperature stays where ex-2.2.11 set it.

Exploratory analyses

Not part of the plan. Anything we think of after seeing the data goes here, marked as post hoc.

Which red colors survive

The side split says the surviving lines have G = B, and three grid colors do. Splitting the on-axis lines by the red operand's color says which.

red operand	side	lines	kept ↓
`#ff0000`	G = B	48	0.105 (0.00–0.82)
`#ff3333`	G = B	44	0.553 (0.02–0.99)
`#cc0000`	G = B	55	0.489 (0.06–0.99)
`#ff0033`	G < B	47	0.082 (0.00–0.41)
`#ff0066`	G < B	46	0.067 (0.00–0.46)
`#ff3300`	G > B	52	0.054 (0.00–0.26)
`#ff6600`	G > B	41	0.067 (0.00–0.31)

Post hoc: kept share under `projection` on the `hue-hsv` removal lines by the color of the red operand, `handover`, mean over lines and then over seeds, with the seed range. Every grid color at or above the red dose appears once.

Nearly all of the survival comes from two colors: the darker red and the paler one. Pure red, with the same hue and the highest redness of the seven, is removed about as cleanly as the hue-shifted reds on the seed mean, though with a wider spread: one seed keeps most of its pure-red lines. So what survives is not hue zero as such. It is not the red dose as such either, since two of the hue-shifted reds have the same redness as the surviving pair.

We do not have a mechanism for the pair. A `hue-hsv` answer takes its hue from the second operand and its saturation and value from the first, so the answer of a surviving line is a hue-zero color at the saturation and value of the first operand.

One candidate is that the model has learnt to answer those lines from how *achromatic* the second operand looks, meaning its G = B.[3] The axis does not hold that, the blocks can read it off the other channels, and the bypass measurement is consistent with it. A follow-up could test it by projecting at the blocks with the second operand replaced by a gray of the same value.

The side split on the other ops

The same split on every op, for `handover`. The HSV ops that read the red operand's saturation or value show the same shape as `hue-hsv`, with the on-axis lines keeping more, and the RGB ops are flat across the three sides, each at its own level under the gate. The rule that picks removal lines leaves some sides empty on some ops.

op	G = B	G > B	G < B	all
`mix`	0.012 (159)	0.022 (103)	0.025 (103)	0.018
`screen`	0.131 (168)	0.129 (103)	0.160 (116)	0.139
`multiply`	0.143 (169)	0.152 (104)	0.158 (111)	0.150
`lighten`	0.146 (164)	0.121 (106)	0.156 (116)	0.142
`darken`	0.181 (164)	0.167 (105)	0.195 (113)	0.181
`difference`	0.071 (173)	0.047 (112)	0.071 (118)	0.064
`exclusion`	0.112 (169)	0.123 (109)	0.143 (112)	0.124
`hsvmix`	0.056 (173)	0.045 (110)	0.052 (118)	0.052
`hue-hsv`	0.384 (147)	0.056 (93)	0.065 (93)	0.239
`sat-hsv`	0.172 (163)	0.037 (102)	0.050 (112)	0.102
`value-hsv`	0.150 (170)	0.019 (106)	0.026 (113)	0.083

Post hoc: the side split on every op's removal lines, `handover`, seed means with line counts. The rule that picks removal lines leaves some sides empty on some ops.

Discussion

The blind-spot story was a good story and it is wrong. It predicted which lines would survive on `hue-hsv`, and the lines that survive are the ones it said could not. That settles the plane: it was built to hold the one quantity the axis cannot, and that quantity is already removed. The sweep agrees, since every plane condition sits inside the reference band on `hue-hsv`.

The two force conditions and the two τ conditions give the nulls we wanted, and they are nulls on every measurement. At five seeds against ex-2.2.11's bands, that says a doubling of either force factor, or a sharper pool, moves nothing we can resolve.

What survives is narrower than we thought: two of the seven red colors, both on the red axis, on one op, re-derived inside the blocks. That makes the leftover easier to characterise for the next experiment, and harder to remove by anchoring, because the model has a route to those answers that does not run through the axis, either at the embedding or after it. Whether that route matters depends on what the anchored-op experiments will ask of the removal.

The containment question is still open, because of the op we chose. The measurement is in the code and takes any op; the next round that looks at op1 drift should run it on an op whose answer can be red on its own.

Method

The side of red

A grid color's side is the sign of G − B. The seven grid colors at or above the red dose split 3 / 2 / 2 across G = B, G > B, and G < B, and the removal lines of `hue-hsv` carry those counts in the first figure's table.

The plane

The home of *red* under the plane conditions is axes 0 and 1 of the stream, e_1 and e_2 together. Alignment is the length of the projection of a state onto the pair. The anchor and anti-subspace terms, the trajectory measurements, and the projection operator all take the pair where they took the axis. Every plane measurement is compared with the control checkpoints scored on the same pair. The concept holds two coordinates of sixty-four rather than one, and its variance share counts both.

The label groups

The containment split groups each `mix` probe line by which of its other colors could have earned it the whole-line label: none, the second operand at the red dose, the answer at the red dose, or both. Lines whose first operand is itself red form a fifth group, reported in the table and left out of the figure. The labeller draws are stochastic, so a line is assigned to the group it could belong to under the labeller, rather than to whatever a particular epoch drew.

Budget

40 runs at d64, each as long as an ex-2.2.11 run (the two `L6` conditions about half again as long), at about three minutes a run on an L4. Scoring adds the part 1 edits on the checkpoints of three conditions, and the sweep conditions under the operators from ex-2.2.11. The reference condition is memoized from ex-2.2.11. The run cost about six dollars on Modal.

---

1. The *residual stream* is the running vector of activations that each layer of a transformer reads from and writes back to. ↩

2. The *cosine similarity* of two vectors is the cosine of the angle between them: one when they point the same way, zero when they are at right angles. ↩

3. An *achromatic* color is one with no hue: a gray, where the three channels are equal or nearly so. ↩