Three recipes side by side

> We re-scored the recipe of record and two soft-start recipes. The soft start on the step stayed close to the recipe of record on every measurement. The pool over slices 1 to 3 removed about as much and cost the task least, but its edit still spilled past the criterion on some seeds. That spill° enters with the edit at block 1, the one slice the pool pulls and the step leaves alone.

Spill. The side effect of the edit on the other ops…

The recipe of record, `falloff`, was chosen in split-anti-seeds before the pooling temperature° could start soft. Two recipes from the soft-pool experiment have since looked close to it. `step, soft τ` is the same as `falloff`, but with a soft start. It kept the anchor on every seed of the loss-rate screen, and against the median baseline° of reference-baseline its spill is within the criterion on every run.

Pooling temperature. How sharply the anchor pull picks its positions….

Median baseline. What an arbitrary direction costs…

`pool 1–3, soft τ` pools the pull over slices 1 to 3 as well as over positions. So the pull chooses the slice within that range, where the other recipes name the blocks. That makes it the easier recipe to explain and to carry to another model. It is worth having even if it scores a little lower, as long as we know how much lower.

So how do the three compare when every measurement the D2.2 design asks for is taken the same way on each? This report trains nothing. It scores the stored runs of all three at seeds 704 to 711, with two small new passes: the write° bound° on the two soft conditions, and the median baseline on the seven loss-rate runs that were trained to the end.

Write. How much e_1 the blocks after an edit put back on the states of the other ops, as an angle…

Bound. The write allowed…

Observations

- Removal, spill and task (E1): the three remove about the same share of the op, and the two soft recipes remove a couple of hundredths less than `falloff` on most seeds. `falloff` and `step, soft τ` stay within the spill° criterion on every run, and the pool goes over it on two. All three pass the task gate° on the mean; the pool costs the least, and only `falloff` has an op past the per-op line, on one seed.

  Spill. The side effect of the edit on the other ops…

  Task gate. The task criterion of the D2.2 design…

- What the edit does (E2): past the embedding, the write° stays near the bound° in all three. The pool goes a little over it at slice 1, on positions where the bound is a few degrees. The grading° and the landing° barely differ. Under the pool the other ops sit closer to e_1 at the embedding and at slice 1.

  Write. How much e_1 the blocks after an edit put back on the states of the other ops, as an angle…

  Bound. The write allowed…

  Grading. Whether the edit takes most from the contexts that needed the op most…

  Landing. How far the answers move toward the target null…

- Where the evidence sits (E3): under the pool the evidence for the op peaks at slice 3 at every seed, with the seeds nearly on top of one another, and under the step at slice 4. Under every recipe, the edit at blocks 2 to 4 removes the op and spills nothing, and the edit at the embedding alone spills nothing either. What spill there is enters with the edit at block 1, and most of it comes under the pool. The e_1 components of the color embeddings follow lightness more closely under the pool than under the step, but that lean is not what the edit takes away.

Scope and method

This is an exploratory re-analysis, with no preregistration and no gate. It covers three conditions at model seeds 704 to 711, 400 epochs each, all with the `falloff` anti weight:

- `falloff`, the recipe of record: the step pull on slices 2 to 4, pooled over positions at a fixed temperature τ_p of 0.1 (the runs of split-anti-seeds).
- `step, soft τ`: the same step, with τ_p starting at 1 and sharpening to 0.1 by the end of the pull warm-up at epoch 40 (soft-pool).
- `pool 1–3, soft τ`: one pull pooled over the slice and position pairs of slices 1 to 3, with both temperatures starting soft (τ_p 1 → 0.1, and the slice temperature τ_ℓ 4 → 0.4) and nothing pulling slice 4 (soft-pool).

The runs at a seed share their initialization, batches, and label draws, and the control of ex-2.2.23 at that seed (trained without the anchor) shares them too. So each comparison is paired by seed. A difference between two conditions is set against the spread of the paired differences over the eight seeds, rather than against the seed range of either condition. We report the mean paired difference from `falloff`, with its range and the count of seeds on each side of zero.

The comparison is of recipes, not of factors. `pool 1–3, soft τ` differs from `step, soft τ` in which slices it pulls as well as in pooling over them. So a gap between those two cannot be put down to either change alone. Soft-pool lists what each change moves.

Eight seeds are too few to compare how often a recipe loses the anchor. None of the 24 runs here lost it. The loss-rate screen counted losses at 40 seeds each for `falloff` (one loss) and `step, soft τ` (none), and bounded the rates at about one in nine and one in fourteen. The pool has no count beyond these eight. This report quotes those counts and does not add to them.

The loss-rate screen trained seven of its runs to 400 epochs: the four `falloff` runs it did not count as kept (one of them the lost run), and three `step, soft τ` runs it finished as a check. They have no control at their seeds, so removal° and spill° are scored against the median baseline° alone. They were picked as slow or doubtful seeds, so they are a check on the slow end and not a sample of either condition.

Removal. How much of `difference` the full edit takes out…

Spill. The side effect of the edit on the other ops…

Median baseline. What an arbitrary direction costs…

Their task change° is taken against the mean of the eight controls at 704 to 711, as loss-rate took it. The eight controls differ from one another by less than a fiftieth on the mean score, which is about how far the choice of control could move it.

Task change. What the anchor costs the task…

Every edit is scored on the 14,000 held-out contexts of ex-2.2.21, 2,000 per op. The edit takes a share of the e_1 component out of the state at every position and every slice. That share is the dose: a quarter, a half, three quarters, or all of it. The measurements are defined in the glossary at the end.

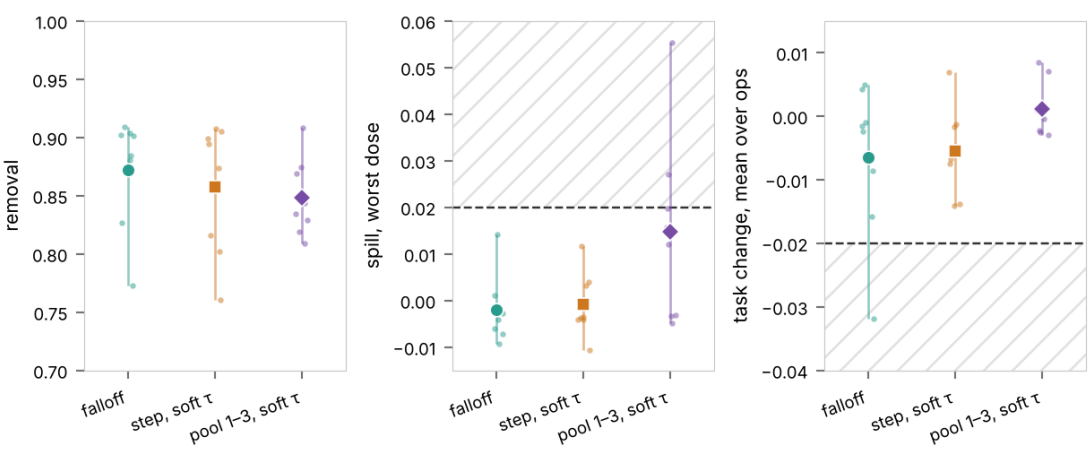
Removal, spill and task (E1)

Do the three recipes remove as much, spill° as little, and cost the task as little, at each seed? The main measure is removal° at the full dose, with the spill at the worst dose beside it and the task change° after that. The paired difference from `falloff` at each seed says whether a gap is larger than the variation between seeds would make by itself. The seven finished loss-rate runs come last.

Spill. The side effect of the edit on the other ops…

Removal. How much of `difference` the full edit takes out…

Task change. What the anchor costs the task…



Removal, spill and task, by condition. *One column per condition, the 8 seeds as faded dots over a thin bar for their range, and the seed mean on top in the marker of the condition.* ***Left:*** *removal at the full dose, net of the median over the other axes of the run.* ***Middle:*** *spill, the largest net drop on another op at any dose; the dashed rule is the selectivity criterion, with the side that misses it hatched.* ***Right:*** *the task change, the mean clean score over ops less that of the control at the seed; the dashed rule is the task gate on that mean.*

condition	removal ↑	removal, nothing subtracted ↑	removal less `falloff`, paired	spill ↓	over criterion ↓	over, control median ↓	task change ↑	worst op change ↑
`falloff`	0.87 (0.77 to 0.91)	0.96 (0.87 to 0.99)	—	−0.002 (−0.009 to 0.014)	0 of 8	0 of 8	−0.007 (−0.032 to 0.005)	−0.066 (`hue-hsv`, 708)
`step, soft τ`	0.86 (0.76 to 0.91)	0.94 (0.84 to 0.99)	−0.02 (−0.07 to 0.00); 1 up, 7 down	−0.001 (−0.011 to 0.012)	0 of 8	0 of 8	−0.005 (−0.014 to 0.007)	−0.036 (`hue-hsv`, 705)
`pool 1–3, soft τ`	0.85 (0.81 to 0.91)	0.93 (0.90 to 0.98)	−0.02 (−0.08 to 0.06); 1 up, 7 down	0.015 (−0.005 to 0.055)	2 of 8	3 of 8	0.001 (−0.003 to 0.008)	−0.015 (`hue-hsv`, 708)

Removal, spill and task*, seed mean and range over 8 runs. Removal and spill are net of the median drop over the other 63 coordinate axes of the run, and "over, control median" counts the runs over the criterion with the median taken in the control at the seed instead. The paired gap is the removal of the condition less that of* `falloff` *at the same seed, with the count of seeds on each side of zero. The task change is the mean over ops of the clean score less that of the control at the seed; the worst op change is the largest loss on one op at any seed.*

The three remove about the same share of the op. The two soft recipes come in under `falloff` on seven seeds of eight, by a couple of hundredths on the mean, a gap that is small next to the spread between seeds (the paired column of the table). With nothing subtracted the order is the same, so the gap does not come from the baseline.

On spill, the recipes differ. `falloff` and `step, soft τ` stay well within the criterion on every run. The pool goes over it on 2 seeds: 707, and 711 by more than twice the criterion. On a third seed (705) it sits at the criterion, and which side it falls on depends on whose median is subtracted.

Under the pool, the spill lands on `value-hsv` on three seeds. That is the op on which the spill-timing experiment found the late spill of this condition. On the two seeds over the criterion it lands on `mix` and `darken`.

On the task, the order reverses. All three pass the gate on the mean over ops, and the pool costs the least, with the narrowest spread. `falloff` is the one with an op past the per-op line: `hue-hsv` at seed 708, a little past it, which split-anti-seeds already noted. The worst op under the two soft recipes is `hue-hsv` too, at well under the line.

condition	measure	704	705	706	707	708	709	710	711
`falloff`	removal	0.90	0.91	0.83	0.90	0.77	0.90	0.88	0.88
	spill	0.014	0.001	−0.003	−0.007	−0.001	−0.004	−0.009	−0.006
`step, soft τ`	removal	0.91	0.91	0.80	0.90	0.76	0.89	0.87	0.82
	spill	−0.004	0.003	−0.004	−0.004	−0.004	0.004	−0.011	0.012
`pool 1–3, soft τ`	removal	0.83	0.91	0.82	0.87	0.83	0.87	0.84	0.81
	spill	−0.003	0.020	−0.005	**0.027**	0.012	−0.003	0.016	**0.055**

Removal and spill by seed*, as in the table above.*

Bold marks a run over the selectivity criterion.

condition	seed	removal ↑	nothing subtracted ↑	spill ↓	task change ↑	worst op change ↑
`falloff`	714	−0.03	0.07	0.015	−0.022	−0.035 (`hue-hsv`)
`falloff`	724	0.84	0.92	−0.001	−0.009	−0.030 (`sat-hsv`)
`falloff`	725	0.89	0.96	−0.004	0.004	0.000 (`mix`)
`falloff`	735	0.85	0.96	−0.007	−0.007	−0.027 (`difference`)
`step, soft τ`	724	0.88	0.98	−0.007	−0.005	−0.020 (`sat-hsv`)
`step, soft τ`	732	0.82	0.90	−0.002	−0.052	−0.089 (`hue-hsv`)
`step, soft τ`	734	0.86	0.97	−0.006	0.000	−0.006 (`hue-hsv`)

The loss-rate runs trained to 400 epochs. *Removal and spill net of the median over the other axes of the run, as above. These seeds have no control, so the task change is against the mean of the 8 controls at 704 to 711, whose mean scores span 0.017 from the lowest to the highest.*

The runs the loss-rate screen trained to the end tell the same story at other seeds. The lost `falloff` run (714) removes nothing with the median subtracted either. It sits just under the mean line of the task gate°, by less than the controls differ among themselves. Every kept run removes about as much as the runs above and spills nothing.

Task gate. The task criterion of the D2.2 design…

One `step, soft τ` run (732) scores below both lines of the task gate against the mean of the controls, by more than the spread of the controls could account for. Loss-rate saw the same and had no control at that seed to settle it, and neither do we. Counting the eight runs at 704 to 711, the soft start has one run below the task gate in eleven, and `falloff` has one past the per-op line in twelve, besides the lost run.

What the edit does (E2)

The measurements of E1 are means over all the contexts of an op. Edit-effects showed on `falloff` that the edit also behaves well context by context. The blocks downstream put back less e_1 on the other ops than the clean model has. The drop on each `difference` context follows how much that context needed the op. And the edited answers land about half of the way to the target null°.

Do the soft start and the pool keep those properties? The main measure is the write° as a ratio of the bound°, with the grading° and the landing° beside it.

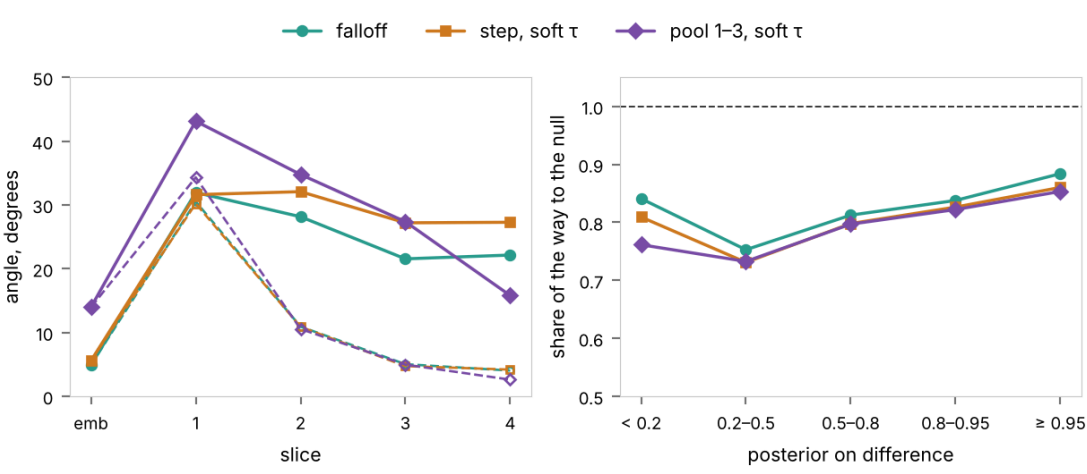
Target null. Where a removal should land…

Write. How much e_1 the blocks after an edit put back on the states of the other ops, as an angle…

Bound. The write allowed…

Grading. Whether the edit takes most from the contexts that needed the op most…

Landing. How far the answers move toward the target null…



What the edit does, by condition. Left: *the bound and the write at the example answers of the other ops that do not fit* `difference`*, by slice: arcsin |α| at the 99% quantile, clean (solid, the bound) and as it arrives under the full edit (dashed, the write), seed mean.* ***Right:*** *the share of the way to the target null the full edit takes on* `difference` *contexts, by bin of the posterior on* `difference`*, net of the control, seed mean; the dashed rule at 1 is the target null.*

condition	largest write over bound ↓	sites over 1.25×, all runs ↓	bound at example answers, emb and slice 1 ↓	share removed ↑	share, not net of the control ↑	landing ↑
`falloff`	1.08 (slice 1, *?*, 0.3°)	28 (≤ 4°)	5°, 32°	0.85 (0.79 to 0.90)	0.96 (0.87 to 0.99)	0.52 (0.49 to 0.56)
`step, soft τ`	1.15 (slice 1, *a*, 3.6°)	23 (≤ 6°)	6°, 32°	0.83 (0.77 to 0.90)	0.94 (0.84 to 0.99)	0.49 (0.47 to 0.53)
`pool 1–3, soft τ`	1.33 (slice 1, *a*, 4.9°)	48 (≤ 5°)	14°, 43°	0.83 (0.79 to 0.88)	0.93 (0.90 to 0.98)	0.51 (0.47 to 0.55)

The write, the grading and the landing. *The largest ratio of write to bound in angle over the slices past the embedding and every position, from the seed-mean quantiles as in edit-effects, with where it falls and the bound there. The sites over 1.25× count slice and position pairs over all 8 runs, each run taken on its own, of 768, with the largest bound among them. The bound at the example answers is over those that do not fit* `difference`*. The share removed is over every* `difference` *context, and the landing is that of split-anti-seeds and soft-pool; seed mean and range.*

Context by context, the three behave alike. Past the embedding the write stays close to the bound under every recipe, and from slice 2 on the other ops arrive with much less e_1 than the clean model gives them, the same in all three. The grading has the same shape under each: about the same share removed in every bin of the posterior, with the pool lower in the bin of contexts that needed the op least. The landing is about half of the way to the target null in all three.

The pool is where they differ, at the first two slices. There the other ops sit closer to e_1 under the pool: the bound at their example answers is about 3× as wide at the embedding as under `falloff`, and about a third wider at slice 1.

At slice 1 under the pool, the write goes a little over the bound at the example operands, where the bound is a few degrees. That is the largest ratio of any recipe, and the only one past the tolerance of ex-2.2.1 at the seed mean, but in degrees it is small. More single sites go over the tolerance under the pool than under the step, all where the bound is 6° or less.

So the pool does not change what the edit does downstream. It changes how close the other ops sit to e_1 at the slices before the model has read the context. The step deliberately puts no pull there.

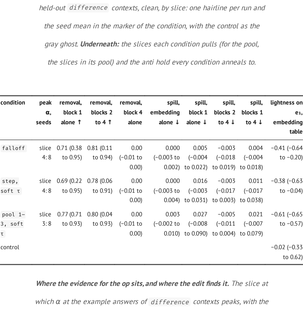
Where the evidence sits (E3)

The step puts the pull on slices 2 to 4, while the pool lets the pull choose among slices 1 to 3. In slice-terms, a pull pooled over blocks 1 to 4 at a fixed temperature settled on block 4 on most seeds and on the query answer on one, where the op could not be removed. Soft-pool found the pool over slices 1 to 3 peaking at slice 3 on average. Does it settle on the same slice at every seed, and does the edit find it there?

What the pull lands on is the evidence for the op. In example-evidence, α^{*} at an example answer followed how well that one example fits `difference`, and in equals-evidence the anchor was nearly absent at the `=` sites, where the model infers the op itself. So this section follows where the evidence sits. The main measure is α at the example answers by slice, per seed, with the removal[*] by the edit restricted to a set of slices beside it.

Alignment. How far a style points along ex…

Removal. How much of `difference` the full edit takes out…



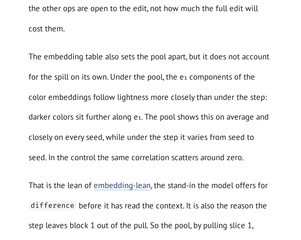
Where the anchor sits by depth. Top: *α at the example answers of the held-out* `difference` *contexts, clean, by slice: one hairline per run and the seed mean in the marker of the condition, with the control as the gray ghost.* ***Underneath:*** *the slices each condition pulls (for the pool, the slices in its pool) and the anti hold every condition annexis to.*

condition	peak α, seeds	removal, block 1 alone ↑	removal, blocks 2 to 4 ↑	removal, block 4 alone	spill, embedding alone ↓	spill, block 1 alone ↓	spill, blocks 2 to 4 ↓	spill, blocks 1 to 4 ↓	lightness on e_1, embedding table
`falloff`	slice 4: 8	0.71 (0.38 to 0.95)	0.81 (0.11 to 0.94)	0.00 (−0.01 to 0.00)	0.000 (−0.003 to 0.002)	0.005 (−0.004 to 0.022)	−0.003 (−0.018 to 0.019)	0.004 (−0.004 to 0.018)	−0.41 (−0.64 to −0.20)
`step, soft τ`	slice 4: 8	0.69 (0.22 to 0.95)	0.78 (0.06 to 0.91)	0.00 (−0.01 to 0.00)	0.000 (−0.003 to 0.004)	0.016 (−0.003 to 0.051)	−0.003 (−0.017 to 0.003)	0.011 (−0.017 to 0.038)	−0.38 (−0.63 to −0.04)
`pool 1–3, soft τ`	slice 3: 8	0.77 (0.71 to 0.93)	0.80 (0.04 to 0.93)	0.00 (−0.01 to 0.00)	0.003 (−0.002 to 0.010)	0.027 (−0.008 to 0.090)	−0.005 (−0.011 to 0.004)	0.021 (−0.007 to 0.079)	−0.61 (−0.65 to −0.57)
control									−0.02 (−0.33 to 0.62)

Where the evidence for the op sits, and where the edit finds it. *The slice at which α at the example answers of* `difference` *contexts peaks, with the count of seeds. Removal and spill under the full edit restricted to a set of slices, at every position, net of the same edit on the control at the seed; seed mean and range. The last column is the correlation, over the color embeddings, of the e_1 component with lightness (the lean of embedding-lean; negative means darker colors sit further along e_1).*

Under each recipe the evidence settles in one place on every seed. Under the step, α at the example answers climbs to slice 4 and stays there, and the soft start changes nothing in that shape. The pool climbs higher at slice 1, peaks at slice 3 on every seed, and falls back at slice 4, where nothing pulls it. Its eight seeds lie nearly on top of one another, and none of them settled on the query answer.

The edit finds the op wherever its evidence settles. The edit at blocks 2 to 4 alone removes most of it under every recipe, except at seed 710. There it falls short under all three, as it did in every arm of split-anti-seeds, and the edit at block 1 alone removes nearly all of it. The edit at block 4 alone removes nothing in any recipe, the pool included. Under the pool the edit at block 1 alone removes more, and more evenly across seeds, than under the step, which fits the pool pulling there.



Spill under the edit restricted to a set of slices, *by condition: the largest drop on another op under the full edit at every position of those slices only, net of the same edit on the control at the seed. One panel per slice set, the 8 seeds as faded dots over their range and the seed mean in the marker of the condition; the dashed rule is the selectivity criterion, with the side that misses it hatched. The restricted edits are not the full edit taken apart, so their spill does not add up to it. The baseline here is the control, not the median of E1, and against the control the full edit puts two* `step, soft τ` *runs over the criterion too (E4 of reference-baseline), so compare the panels with one another rather than with E1.*

condition	measure	704	705	706	707	708	709	710	711
`falloff`	removal, block 1 alone	0.70	0.76	0.38	0.73	0.49	0.77	0.95	0.92
	removal, blocks 2 to 4	0.91	0.88	0.95	0.93	0.88	0.94	0.11	0.93
	spill, block 1 alone	0.022	0.007	0.006	−0.004	0.011	0.004	0.001	−0.004
`step, soft τ`	removal, block 1 alone	0.72	0.90	0.22	0.68	0.45	0.90	0.95	0.73
	removal, blocks 2 to 4	0.91	0.87	0.91	0.91	0.76	0.87	0.06	0.90
	spill, block 1 alone	0.015	0.031	0.028	−0.005	0.011	0.020	−0.002	0.029
`pool 1–3, soft τ`	removal, block 1 alone	0.75	0.86	0.71	0.73	0.71	0.75	0.91	0.72
	removal, blocks 2 to 4	0.88	0.87	0.91	0.91	0.93	0.92	0.04	0.91
	spill, block 1 alone	0.011	0.031	0.004	0.042	0.041	−0.008	0.009	0.090

The restricted edits by seed, *as in the table above.*

The same restricted edits[*] show where the spill[*] of E1 comes in. The edit at blocks 2 to 4 costs the other ops nothing under any recipe. The edit at block 1 alone costs them a little under `falloff`, more under `step, soft τ`, and most under the pool. Under the pool, the block-1 edit costs most at 707 and 711, the two seeds over the criterion in E1, and at 708. The edit at the embedding alone costs the other ops nothing under any recipe, and the edit at blocks 1 to 4 costs what the edit at every slice does. So under the pool the spill enters with block 1.

Restricted edit. The full edit made at a set of slices, with the rest of the ring held. E.g. blocks 2 to 4, block 4, or blocks 1 to 4, at every position…

Spill. The side effect of the edit on the other ops…

The restricted edits are not the full edit taken apart, since each later block reads what the earlier ones were given. And the block-1 edit on its own overstates the cost: under `step, soft τ` it goes over the criterion on 3 seeds whose full edit spills nothing. So it says where the other ops are open to the edit, not how much the full edit will cost them.

The embedding table also sets the pool apart, but it does not account for the spill on its own. Under the pool, the e_1 components of the color embeddings follow lightness more closely than under the step: darker colors sit further along e_1. The pool shows this on average and closely on every seed, while under the step it varies from seed to seed. In the control the same correlation scatters around zero.

That is the lean of embedding-lean, the stand-in the model offers for `difference` before it has read the context. It is also the reason the step leaves block 1 out of the pull. So the pool, by pulling slice 1, seems to bring back part of the lean.

But the edit at the embedding alone, which takes that lean out, spills nothing. And in the states at slice 1, the lean has faded to about the same level under all three recipes (−0.21 under the pool against −0.18 under `step, soft τ`). So what the block-1 edit removes is what block 1 writes[*] onto e_1 for the other ops, which these measurements do not resolve. The lean in the embedding table is a sign of the pull reaching slice 1, rather than the thing the edit takes away. Within the pool the lean is about the same on every seed, so it does not say why 707 and 711 spill and 704 does not, either.

Write. How much is the blocks after an edit put back on the states of the other ops, as an angle…

Discussion

On these eight seeds `step, soft τ` is about as good a recipe as `falloff`. It removes a couple of hundredths less, spills° no more, and costs the task about as little. The edit also acts the same way context by context. Its one run below the task gate° is a loss-rate run with no control at its seed. And `falloff` has a run of its own past the per-op line, so on the task the two look about level.

Spill. The side effect of the edit on the other ops…

Task gate. The task criterion of the D2.2 design…

The measure that could still set them apart is how often each loses the anchor, which these runs cannot add to. Loss-rate found no losses in 40 for the soft start and one for `falloff`, a difference too small at those counts to rest on.

`pool 1–3, soft τ` is close behind on removal° and ahead on the task, and the evidence settles in the same place on every seed, which is what one would want of a recipe that chooses the slice for itself. Its shortfall is spill.

Removal. How much of `difference` the full edit takes out…

The pool still has a range, slices 1 to 3, but the range follows from the architecture rather than from what we know of the latent representations. The embedding holds no context, so the inferred op cannot exist there. And at slice 4 no block is left to carry what sits at an example answer to the query `=`, so the op there cannot reach the prediction. Slices 1 to 3 are the only ones where it can do both, which carries to another model by its depth alone.

The edit at block 1 is where the spill enters, and block 1 is the one slice the pool pulls and the step leaves alone. What block 1 writes° onto e_1 for the other ops is not resolved here. But E2 shows the other ops sitting closer to e_1 at the first two slices under the pool. How hard and how early the pull acts at slice 1 is set by the schedules of the anchor weight λ_a and the slice temperature τ_ℓ, and this pool has not been tuned on either.

Write. How much e_1 the blocks after an edit put back on the states of the other ops, as an angle…

Glossary

Removal

How much of `difference` the full edit takes out: the drop in its score, as a share of the way from the clean score to the target null, net of the median baseline. Higher is better. Removal with nothing subtracted is shown beside it.

Spill

The side effect of the edit on the other ops: the largest drop in the score of any other op at any dose, net of the median baseline. A run is within the selectivity criterion when it is at most 0.02.

Median baseline

What an arbitrary direction costs: the median drop when the same edit is made along each of the other 63 coordinate axes of the same run, per op and dose (the rule of the D2.2 design).

Task change

What the anchor costs the task: the clean score per op, less that of the control at the same seed.

Task gate

The task criterion of the D2.2 design: it fails if the task change falls below −0.02 on the mean over ops, or below −0.05 on any one op.

Write

How much e_1 the blocks after an edit put back on the states of the other ops, as an angle: arcsin $|\alpha|$ at the 99th percentile, as the states arrive under the full edit (as in edit-effects).

Bound

The write allowed: the same angle on the clean pass, before any edit. At the example answers it counts only those that do not fit `difference`. Ex-2.2.1 allowed the write a quarter over the bound as a tolerance.

Grading

Share removed

Whether the edit takes most from the contexts that needed the op most: the share of the way to the target null that the full edit takes on `difference` contexts, by bin of the posterior on `difference`. It is measured context by context, and the median baseline exists only per op, so it is net of the control at the same seed instead, as in edit-effects. The share before netting is shown beside it.

Landing

How far the answers move toward the target null: the share of the distance to the target null that the full edit covers, in total variation over whole answer distributions, on `difference` contexts.

Restricted edit

The full edit made at a set of slices only (the embedding, block 1, blocks 2 to 4, block 4, or blocks 1 to 4), at every position. The median baseline exists only for the edit at every slice, so a restricted edit is net of the same restricted edit on the control at that seed, as in split-anti-seeds.