Optimizers with no privileged basis

> Adam scales its update per coordinate, which makes the coordinate axes of the residual stream a privileged basis°, and the model parks its syntax tokens one-hot on them. This scout trains the un-anchored control under three optimizers whose update is the same in every orthonormal basis of the residual coordinates. Under all three the axes became ordinary directions, but the stream was no more isotropic° for it. Muon learned the task as well as Adam once its rate was raised.

Privileged basis. Directions that training treats differently from the rest....

Isotropic. The same in every direction....

Along the principal directions° of the stream there is still one large direction, a spike that holds about two fifths or more of the energy at the last slice, as under Adam, and in most runs the separator tokens sit on it (E3 has the cosines between them). Pooled Adam also matched Adam on the task within the seed range, though less closely, and SGD was never tried at a rate high enough to judge.

Principal directions. Where the energy of the states sits...

The question comes from the spill-timing experiment (PR #285): there, reference edits along random directions cost every op far more than the edit along e_1, which led us to check whether the coordinate axes are special. Our prior is that an isotropic stream would be easier to reason about; the open question is what it costs. There are no anchored runs. An optimizer that keeps the stream free of a privileged basis, and learns the task as well, is a candidate for an anchored arm later, with its own tuning of the term weights.

We find that the spike looks like a property of the model and the task, with the optimizer deciding whether it sits on a coordinate axis; its size moves with the learning rate, though. That matters for reference edits, which project out a typical direction rather than the anchored one: under the two new optimizers that learned the task, projecting out a typical coordinate axis costs more than it does under Adam.

Observations

- Does the task suffer? (E1): Muon at three times the finder's rate matches Adam within the seed range. Pooled Adam at that rate matches too by the same rule, though less closely: it is a hundredth behind at both seeds. Both did best at the top rung (the highest of the three learning rates) on both seeds, so the bracket may fall short of their best rates. SGD trails by most of the score at every rate, and was still improving with the rate at the top of the bracket.
- Is the stream isotropic? (E2): no coordinate holds much more of the energy than the largest direction of a random basis does, and the kurtosis° and the removal cost are about the same along the axes as along random directions, so the axes are ordinary directions. Along the principal directions° of the stream, though, the top direction holds 0.38 to 0.67 of the energy at the last slice, about as much as under Adam (0.47 to 0.48). The spike has moved off the axes, and the stream is not isotropic°. Removing a typical axis now costs what a random direction does, which under Muon at the top rung is two to three times what a typical axis costs under Adam.

  Kurtosis. How heavy-tailed the values along a direction are....

  Principal directions. Where the energy of the states sits...

  Isotropic. The same in every direction....
- Where the syntax tokens go (E3): the syntax tokens leave the axes; their largest cosine with any axis is what a random unit vector gives. The separators `,` and ⏎ still share a direction, at a cosine of 0.65 to 1.00 in every run. At the top rung it is the top principal direction, but at the lower rungs of pooled Adam the separators drift off it. Under Muon `=` leans on it too; under Adam and pooled Adam `=` sits apart.
- How the colors spread (E4): under pooled Adam and Muon the color embeddings spread over about as many directions as under Adam, and keep about as much of the RGB order. Nothing here suggests colors are graded better under Muon.

Scope and method

This is an exploratory scout, with no preregistration and no gate. We train with three optimizers, each on the control at model seeds 704 and 705 for the full 400-epoch schedule. Each runs at three learning rates, which we call rungs: the rate the finder of ex-2.2.17 suggests for that optimizer at seed 704, three times it, and a third of it.

The bracket reaches both ways because the finder could be off in either direction. Its 300-step suggestion sat above the rate the learning-rate ladders of ex-2.2.21 chose for Adam, and for SGD it may sit below the rate a long run wants (short SGD runs we tried earlier, on the setup of the stored checkpoints, used rates of 0.3 to 1.0). The three rungs also say how sensitive each optimizer is to the rate.

That makes eighteen runs in all, plus three learning-rate finder runs. They are compared with the two Adam controls of ex-2.2.23 at the same seeds, which share the initialization, the batches, and the model. With two seeds, we can tell an optimizer from Adam only where the difference is larger than the seed range of the stored controls (about 0.02 in the task score; see the control columns of split-anti-seeds). So the scout can rule optimizers in or out, but it cannot rank close ones.

The three optimizers:

- Pooled Adam: Adam, with its second moment[2] pooled over the residual axis of each matrix that reads or writes the stream, so the scaling is one number per row of a reading matrix, or per column of a writing matrix, and it is the same in any orthonormal basis of the residual coordinates. The gains, and the other axis of each matrix, are scaled per coordinate as in Adam.
- Muon: before each step, the momentum of each matrix is orthogonalized, by a Newton–Schulz iteration toward the nearest orthogonal matrix. This is equivariant° to a rotation on either side of the matrix. Every matrix gets this, the embedding and readout tables included; the usual Muon recipe leaves the tables to Adam. The gains are updated with Adam at the learning rate of the Adam recipe, following the same schedule.
- SGD with momentum 0.9, the plain reference, with no per-coordinate scaling at all.

A test in the repo checks that training in a rotated basis changes nothing for these three, and does change Adam. None of them makes training much dearer here: in a quick timing of one update against one forward and backward pass of this model, the Muon update took a few percent of the step, and pooled Adam took about what Adam takes.

2 Second moment…

Equivariant. Rotating the input rotates the output the same way.…

The measurements

- The task: the expected exact match° of each op before any edit, as the basis measurement takes it on held-out contexts, compared with the Adam control at the same seed. Also the validation loss along training.

Expected exact match. The task score, as the chance that a drawn answer agrees with a true one....

- Isotropy°, from the basis measurement of rotated-basis: the share of the stream energy that each coordinate axis holds, sorted, against the same along a random orthonormal basis; the mean kurtosis° along each basis, higher along the axes than along random directions for a privileged basis°; and the cost of projecting out one direction at full dose, for each coordinate axis and for sixteen random directions. On the Adam controls, measured here, one coordinate holds 0.45 to 0.47 of the energy, the kurtosis is 4.3 along the axes against 3.0 to 3.1 along random directions, and a typical axis costs 0.03 against 0.07 to 0.12 for a random direction.

Isotropic. The same in every direction....

Kurtosis. How heavy-tailed the values along a direction are....

Privileged basis. Directions that training treats differently from the rest....

- Where the tokens sit: the largest cosine of each token embedding with a coordinate axis, and which axis that is. Under Adam the syntax tokens `?`, `,` and ⏎ sit at a cosine near 1 on one coordinate and `=` on another.
- How the color embeddings spread: over how many directions the 216 color embeddings spread (their participation ratio[1]), how much of their spread a linear map from the RGB channels explains, the rank correlation between their distances and the RGB distances, and their mean cosine with each other and with the separator direction (the mean of the `,` and ⏎ embeddings).

1 Participation ratio...

- The spectrum along the principal directions° of the stream: the eigenvalues of the second-moment matrix of the states at each slice, over the positions of 1000 held-out contexts, as shares of the total. A spike shows here whichever direction it sits on. Alongside it, we take the cosine of each syntax token embedding with the top direction at the last slice, and the pairwise cosines among the syntax embeddings.

Principal directions. Where the energy of the states sits...

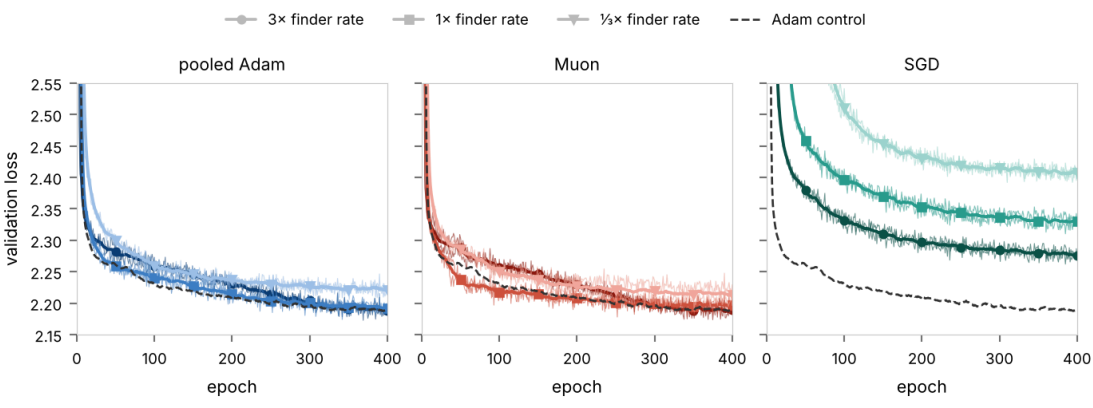
Does the task suffer? (E1)

Is the task learned as well without a privileged basis°, at any of the three rates? An optimizer that trails Adam by more than the seed range at every rate is out for now; one that matches at one rate is a candidate. If an end rung does best at both seeds, though, the best rate may lie outside the bracket, and the shortfall with it. The finder history (whether the loss was still falling at the end of its range) bears on that.

Privileged basis. Directions that training treats differently from the rest....

The main measure is the task change: the expected exact match° averaged over the ops, less that of the Adam control at the same seed; the table below has it per run. The validation loss along training, in the figure, shows the same order, and also how each run gets there.

Expected exact match. The task score, as the chance that a drawn answer agrees with a true one....



***Validation loss along training**, one panel per optimizer. Each rate is a shade of the optimizer hue, strongest for three times the finder's rate. The two seeds are hairlines; their mean, smoothed over nine epochs, is drawn over them with a marker every 50 epochs. The dashed line is the seed mean of the Adam control at the recipe's rate, smoothed the same way.*

optimizer	rate	learning rate	704: mean ↑	704: worst op ↑	705: mean ↑	705: worst op ↑	val loss ↓
pooled Adam	3×	0.005	−0.009	−0.019	−0.010	−0.020	2.191
	1×	0.0017	−0.039	−0.055	−0.032	−0.052	2.193
	⅓×	0.00055	−0.225	−0.407	−0.190	−0.384	2.222
Muon	3×	0.017	−0.004	−0.010	+0.006	−0.015	2.189
	1×	0.0058	−0.010	−0.034	−0.001	−0.017	2.198
	⅓×	0.0019	−0.148	−0.254	−0.050	−0.073	2.213
SGD	3×	0.029	−0.408	−0.634	−0.419	−0.634	2.274
	1×	0.0097	−0.462	−0.697	−0.468	−0.694	2.324
	⅓×	0.0032	−0.496	−0.724	−0.492	−0.711	2.399
Adam (control)		0.0032	·	·	·	·	2.193

***The task change against Adam, per run.** The rate is a multiple of the finder's suggestion for that optimizer, and the learning rate is the peak rate it gives. For each seed, "mean" is the expected exact match averaged over the 7 ops less that of the Adam control at the same seed, and "worst op" is the largest drop on any one op. The last column is the validation loss at the end of training, seed mean. Arrows mark which direction is better.*

Muon at three times the finder's rate matches Adam: the change is within a hundredth at both seeds, and no op falls by more than the seed range. It also ends with the lowest validation loss of any run here, though only by a few thousandths. At the finder's rate it is nearly as close, and at a third of it, it trails clearly. The finder's rate leads for most of training, though: its validation loss is up to 0.05 below that of the top rung under Muon and up to 0.02 under pooled Adam, and the top rung passes it only after epoch 301 and 317 respectively, as the rate decays.

Pooled Adam trails a little at the top rung, by about a hundredth on the mean and twice that on its worst op, more at the finder's rate, and far more at a third of it. By the rule above it is a candidate, since a hundredth is inside the seed range, though a weaker one than Muon. SGD trails by most of the score at every rate.

For all three optimizers the top rung did best at both seeds, so by the rule above the bracket may have missed the best rate. The finder fits that: in its last zoom the loss was still falling at the top of its range for every optimizer, and its suggestion is the rate where the loss fell fastest, which is below the rate where it was lowest.

But a finder ramps the rate up over one short run, so the later, higher rates have also had more steps. That alone could put the lowest loss at the top, so the finder cannot place the best rate by itself. It does say that the suggestion was low for every optimizer here; for pooled Adam it was 0.0017, against the 0.0032 the recipe uses for plain Adam.

For pooled Adam and Muon this changes little in a scout. Muon already matches Adam, and pooled Adam is close enough that a higher rate might make up the difference, which these runs do not test. With rungs a factor of three apart, the best rate could also sit between the finder's rate and the top rung; a finer ladder would place it, and is worth running only once an optimizer is chosen for an anchored arm.

For SGD it matters more. The validation loss falls steadily as the rate rises (see the table), and the top rung, at 0.029, is still about a tenth of the lowest rate of those earlier short SGD runs. So SGD was not tested at a useful rate, and these runs say nothing against it. They do mean that the SGD runs in E2 and E3 are models that learned little of the task.

Is the stream isotropic? (E2)

Does the energy spectrum flatten, and does the kurtosis[*] along the axes fall to that along random directions? The equivariance[*] means the optimizer has no privileged basis[*]. It does not make the stream isotropic[*], since the model may still build a spike along some direction. A spike off the axes spreads its energy over the axes and over a random basis alike, so the spectrum along either basis looks nearly even, and these measurements would miss it.

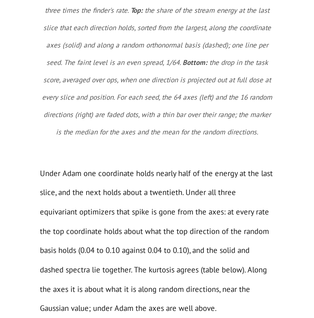
Kurtosis. How heavy-tailed the values along a direction are.

Equivariant. Rotating the input rotates the update the same way.

Privileged basis. Directions that training treats differently from the rest.

Isotropic. The same in every direction.

The main measure is the share of the stream energy that the largest coordinate holds at the last slice, beside the share that the largest direction of a random orthonormal basis holds. If the axes are a privileged basis the first is much larger; if they are ordinary directions the two are about the same.



The energy spectrum and the cost of a direction. *Adam and each optimizer at three times the finder's rate. **Top:** the share of the stream energy at the last slice that each direction holds, sorted from the largest, along the coordinate axes (solid) and along a random orthonormal basis (dashed); one line per seed. The faint level is an even spread, 1/64. **Bottom:** the drop in the task score, averaged over ops, when one direction is projected out at full dose at every slice and position. For each seed, the 64 axes (left) and the 16 random directions (right) are faded dots, with a thin bar over their range; the marker is the median for the axes and the mean for the random directions.*

Under Adam one coordinate holds nearly half of the energy at the last slice, and the next holds about a twentieth. Under all three equivariant optimizers that spike is gone from the axes: at every rate the top coordinate holds about what the top direction of the random basis holds (0.04 to 0.10 against 0.04 to 0.10), and the solid and dashed spectra lie together. The kurtosis agrees (table below). Along the axes it is about what it is along random ones, near the Gaussian value; under Adam the axes are well above.

optimizer	rate	top share, axes	top share, random basis	top share, principal	kurtosis, axes	kurtosis, random basis	cost, median axis	cost, random direction
Adam		0.46 (0.45 to 0.47)	0.06 (0.04 to 0.07)	0.47 (0.47 to 0.48)	4.31 (4.30 to 4.32)	3.05 (3.01 to 3.09)	0.03 (0.03 to 0.03)	0.09 (0.07 to 0.12)
pooled Adam	3×	0.07 (0.05 to 0.10)	0.06 (0.05 to 0.06)	0.42 (0.40 to 0.44)	2.85 (2.80 to 2.91)	2.97 (2.93 to 3.02)	0.05 (0.05 to 0.05)	0.06 (0.05 to 0.06)
	1×	0.08 (0.06 to 0.10)	0.06 (0.05 to 0.06)	0.45 (0.45 to 0.46)	2.80 (2.78 to 2.81)	2.85 (2.83 to 2.88)	0.07 (0.07 to 0.07)	0.07 (0.07 to 0.08)
	⅓×	0.08 (0.08 to 0.09)	0.05 (0.05 to 0.06)	0.44 (0.42 to 0.45)	2.73 (2.68 to 2.78)	2.75 (2.74 to 2.75)	0.09 (0.08 to 0.10)	0.11 (0.10 to 0.11)
Muon	3×	0.04 (0.04 to 0.05)	0.04 (0.04 to 0.05)	0.40 (0.40 to 0.40)	3.01 (2.90 to 3.12)	3.02 (2.94 to 3.10)	0.08 (0.06 to 0.10)	0.09 (0.07 to 0.11)
	1×	0.07 (0.06 to 0.07)	0.06 (0.05 to 0.08)	0.45 (0.44 to 0.46)	3.16 (3.12 to 3.20)	3.08 (3.08 to 3.08)	0.12 (0.10 to 0.14)	0.16 (0.15 to 0.19)
	⅓×	0.08 (0.07 to 0.08)	0.07 (0.06 to 0.08)	0.64 (0.62 to 0.67)	3.36 (3.33 to 3.39)	3.34 (3.29 to 3.40)	0.19 (0.18 to 0.21)	0.19 (0.15 to 0.23)
SGD	3×	0.06 (0.04 to 0.08)	0.07 (0.06 to 0.07)	0.40 (0.38 to 0.42)	3.06 (3.01 to 3.10)	3.24 (3.20 to 3.28)	0.02 (0.02 to 0.03)	0.03 (0.03 to 0.03)
	1×	0.06 (0.06 to 0.07)	0.08 (0.07 to 0.09)	0.48 (0.47 to 0.49)	3.02 (3.01 to 3.02)	3.31 (3.26 to 3.36)	0.02 (0.02 to 0.03)	0.03 (0.03 to 0.03)
	⅓×	0.07 (0.05 to 0.08)	0.08 (0.07 to 0.10)	0.49 (0.46 to 0.51)	2.91 (2.85 to 2.96)	3.28 (3.22 to 3.34)	0.03 (0.03 to 0.03)	0.03 (0.03 to 0.04)

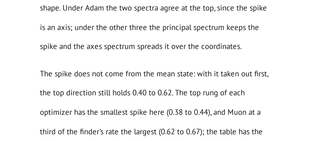
Energy, kurtosis and cost along the axes and along random directions. *seed mean and range over the two seeds. The top share is the share of the stream energy that the largest direction holds at the last slice, along the coordinate axes, along a random orthonormal basis, and along the stream's own principal directions. The kurtosis is the mean over the directions of a basis, averaged over slices; a Gaussian has 3. The cost is the drop in the task score, averaged over ops, from projecting out one direction at full dose: the median over the 64 axes, and the mean over the 16 random directions.*

The cost of projecting out one direction says the same. Under Adam a typical axis is cheap, and a random direction costs a few times more because every random direction touches the spike. Under pooled Adam and Muon a typical axis costs about what a random direction costs. The axis costs spread wider under Muon than under pooled Adam at the top rung: the middle half of them spans 0.07 to 0.14 against 0.03 to 0.04. Under SGD both are cheap, but SGD learned little, so there is little to lose. So the axes are ordinary directions under all three optimizers.

Ordinary axes do not make the stream isotropic, though. Seen from the random basis, which it is not aligned with, the large spike of the Adam controls looks like the new spectra: the top direction holds 0.04 to 0.07 under Adam, in the same range as the new runs along either basis. So the measurements above cannot separate an even stream from a spike that has moved off the axes.

The spectrum along the principal directions[*] of the stream can. They are the directions where the energy sits, so a spike shows there wherever it points.

Principal directions. Where the energy of the matrix sits.



The energy spectrum along the principal directions of the stream, *Adam and each optimizer at three times the finder's rate. The share of the stream energy at the last slice that each direction holds, sorted from the largest, along the principal directions (solid) and, for comparison, along the coordinate axes (dashed, as in the figure above); one line per seed. The faint level is an even spread, 1/64.*

The spike is still there. At the last slice the top principal direction holds 0.38 to 0.67 of the energy under the new optimizers, against 0.47 to 0.48 under Adam, and the rest of the spectrum has the same shape. Under Adam the two spectra agree at the top, since the spike is an axis; under the other three the principal spectrum keeps the spike and the axes spectrum spreads it over the coordinates.

The spike does not come from the mean state: with it taken out first, the top direction still holds 0.40 to 0.62. The top rung of each optimizer has the smallest spike here (0.38 to 0.44), and Muon at a third of the finder's rate the largest (0.62 to 0.67); the table has the rest.

So the stream is no more isotropic than under Adam; its spike has moved off the axes. The spike is in the SGD runs too, which learned little of the task, so it does not wait on the task being solved. That fits a spike that is a property of the model and the task, with the optimizer deciding whether it lands on a coordinate axis. Its size does move with the rate, though, and two seeds at three rates do not say what sets it.

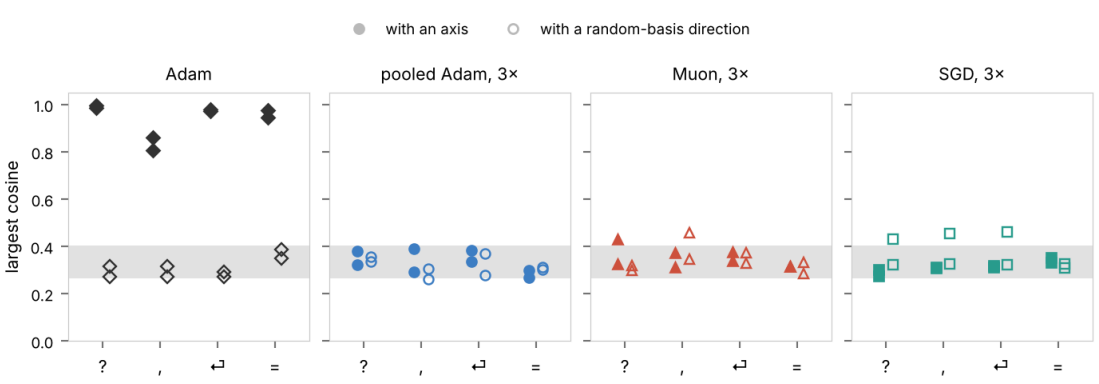
The largest coordinate hints at the same spike. In 10 of 12 pooled Adam and Muon runs, the largest coordinate at the last slice is the axis that one of the syntax token embeddings leans on most, where chance would give about one run in 16. Adam puts its spike on such an axis at both seeds. Among the SGD runs it holds in none, though their separators sit on the top principal direction just as firmly (E3), so the hint is a weak one.

With no privileged basis, a typical axis is no longer cheap to project out. Under Muon at the top rung the median axis costs 0.06 to 0.10 of the task score, against 0.03 for a typical axis under Adam, and at the lower rates more still. The cheapest Muon axes cost about what a typical Adam axis does (0.02), so some quiet axes remain, just not a typical one.

Where the syntax tokens go (E3)

With no cheap one-hot coordinates, where do the separators and `=` sit? They could be: **(1)** spread over many coordinates, **(2)** on a direction of their own that is not an axis, or **(3)** still one-hot on an axis, for a reason the optimizer does not supply.

The measure is the largest cosine of each syntax token embedding with any coordinate axis: near 1 for a token one-hot on an axis, and about 0.3 for a unit vector pointing nowhere in particular among 64 dimensions.



***Where the syntax tokens sit**, Adam and each optimizer at three times the finder's rate. For each syntax token, the largest absolute cosine of its embedding with any coordinate axis (filled) and with any direction of the random basis (open), one marker per seed. The shaded band holds 90% of the same cosine for a random unit vector in 64 dimensions.*

The syntax tokens leave the axes under all three optimizers. At every rate their largest cosine with any axis falls between 0.25 and 0.45, about what a random unit vector gives, and their largest cosine with a direction of the random basis is the same size (0.26 to 0.47). So they sit on no particular direction of either basis, and option 3 is out.

The largest cosine cannot tell options 1 and 2 apart: a token spread over many coordinates and a token on a direction of its own both give a cosine of random size with every axis. The axis where each token leans most gives a hint. In 12 of 18 runs, at least two of `?`, `,` and ↵ lean most on the same axis (2 under pooled Adam, 4 under Muon, 6 under SGD), where three tokens pointing independently would do so in about one run in 22. Under Adam all three share one axis at both seeds.

The cosines between the tokens themselves settle it: the separators keep a direction of their own. In every run `,` and ↵ have a cosine of 0.65 to 1.00 with each other (under Adam, 0.91 to 0.92), and that direction is the spike of E2. At the top rung their cosine with the top principal direction° at the last slice is 0.58 to 0.84, against 0.77 to 0.95 under Adam.

Principal directions. Where the energy of the states sits…

That holds at every rate under Muon and SGD too (0.53 to 0.84). But at the lower rungs of pooled Adam the separators drift off the top direction (0.11 to 0.72), so what the spike holds there is open. `?` joins them under Muon and SGD, and less so under pooled Adam.

The token `=` is the one that moves. Under Adam it sits apart, on a coordinate of its own, at a cosine of 0.13 to 0.18 with the spike and 0.08 to 0.18 with `?`. Under Muon it leans on the spike too, at 0.49 to 0.88 with the top principal direction and 0.33 to 0.59 with `?`. Under SGD it sits between, and under pooled Adam it sits apart as under Adam (0.04 to 0.29 with the spike).

So option 2 holds: the syntax tokens sit on a direction of their own, which is where the stream energy sits. What Adam added was to make that direction an axis, and to give `=` an axis apart.

How the colors spread (E4)

Privileged basis. Directions that training treats differently from the rest.…

Does an optimizer with no privileged basis° lay the colors out differently? If it spreads them over more directions, or orders them more like RGB, an anchored color concept might grade more smoothly under it. These are the color token embeddings only, which the edits do not touch, so this says how the vocabulary is laid out, not how the stream carries a color.

optimizer	rate	participation ratio	RGB, share explained ↑	RGB, rank correlation ↑	mean cosine	cosine with separators
Adam		24.3 (23.0 to 25.5)	0.23 (0.22 to 0.24)	0.67 (0.65 to 0.68)	0.23 (0.22 to 0.24)	0.05 (0.04 to 0.05)
pooled Adam	3×	22.6 (21.0 to 24.3)	0.25 (0.22 to 0.27)	0.69 (0.64 to 0.74)	0.23 (0.19 to 0.26)	0.04 (0.03 to 0.04)
	1×	22.5 (21.3 to 23.7)	0.24 (0.22 to 0.25)	0.64 (0.60 to 0.67)	0.21 (0.21 to 0.21)	0.06 (0.06 to 0.06)
	⅓×	21.2 (21.2 to 21.3)	0.26 (0.25 to 0.26)	0.71 (0.69 to 0.73)	0.31 (0.29 to 0.33)	0.05 (0.05 to 0.06)
Muon	3×	22.1 (22.1 to 22.1)	0.22 (0.20 to 0.25)	0.61 (0.54 to 0.68)	0.31 (0.15 to 0.47)	0.18 (0.09 to 0.26)
	1×	27.0 (24.2 to 29.8)	0.19 (0.19 to 0.20)	0.58 (0.55 to 0.61)	0.26 (0.14 to 0.38)	0.10 (0.06 to 0.13)
	⅓×	31.4 (29.2 to 33.6)	0.15 (0.14 to 0.17)	0.49 (0.46 to 0.53)	0.19 (0.14 to 0.23)	0.14 (0.10 to 0.18)
SGD	3×	38.3 (37.8 to 38.9)	0.14 (0.14 to 0.15)	0.49 (0.48 to 0.51)	0.07 (0.05 to 0.08)	0.06 (0.05 to 0.07)
	1×	46.0 (45.8 to 46.2)	0.09 (0.08 to 0.09)	0.31 (0.29 to 0.33)	0.04 (0.04 to 0.05)	0.08 (0.07 to 0.09)
	⅓×	47.6 (47.3 to 48.0)	0.05 (0.05 to 0.06)	0.18 (0.17 to 0.19)	0.02 (0.02 to 0.03)	0.08 (0.07 to 0.08)

***How the color embeddings spread**, seed mean and range over the two seeds.*

The participation ratio is the number of directions the 216 unit-normalized color embeddings effectively spread over, after taking out their mean. The share explained is the share of that spread a linear map from the three RGB channels accounts for, and the rank correlation is the Spearman correlation between the cosine distance of two colors and their distance in RGB. The last two columns are the mean cosine between colors, and their mean absolute cosine with the separator direction.

Under pooled Adam and Muon the colors spread over about as many directions as under Adam (21 to 30 against 23 to 26), and a linear map from RGB explains about as much of their spread (0.19 to 0.27 against 0.22 to 0.24). The rank correlation with RGB distance is about the same too, a little lower for Muon at seed 705. SGD spreads them over more directions and keeps less of the RGB order, as fits a model that learned little of the task.

One run stands apart: under Muon at seed 705 the colors lean toward the separator direction, at a mean cosine of 0.13 to 0.26 against 0.03 to 0.06 under Adam and pooled Adam, and they sit closer to each other. That is one seed of two, at both rates, so it may be chance, and it says nothing yet about grading.

So on this measure the colors are laid out under Muon much as under Adam. Whether an anchored color, such as red, grades better under Muon is a question for anchored runs.

Discussion

Muon is a candidate for an anchored arm. At three times the finder's rate it learns the task as well as Adam, at both seeds, and its coordinate axes are ordinary directions. Pooled Adam also matches within the seed range, a hundredth behind at both seeds, and since its best rung was also the top one, it may match Adam at a higher rate than this bracket reached. SGD was not tested at a useful rate, so the scout says nothing against it.

All three optimizers show that the privileged basis° under Adam comes from the optimizer, as the stored runs suggested. The spike does not: along the principal directions° of the stream it is about as large under every optimizer here as under Adam, and in most runs the separators sit on it. That fits a spike that belongs to the model and the task, where the part Adam plays is to turn it onto an axis.

Privileged basis. Directions that training treats differently from the rest....

Principal directions. Where the energy of the states sits...

The stream is not isotropic° under any of these optimizers, so an optimizer is not the lever for making it so. What makes the model put the separators on one large direction (the syntax of the task, the unit-norm stream, the width) is a question for a different experiment. So is whether the optimizer matters more in a model that holds many more concepts, where quiet directions may be scarcer.

Isotropic. The same in every direction....

The cost of a direction rises when no direction is privileged. Under Adam most axes are quiet, and an anchor on e_1 takes one of them at small cost to the task. Under Muon a typical axis costs two to three times as much, which fits each axis touching the spike, though the cheapest Muon axes are about as quiet as a typical Adam one.

If that holds for anchored runs, an anchor on e_1 under Muon would have to make room for itself on a direction the task already uses. A quiet direction orthogonal to the spike may still exist, but it would not be an axis, and this scout does not look for it. Under Adam a quiet axis is already there; whether the anchor term can clear a direction as cheaply is a question for anchored runs, and none ran here.

Glossary

Privileged basis

Directions that training treats differently from the rest. Adam divides each coordinate of its update by a running scale of its own, so rotating the coordinates changes the update, and the axes become special directions.

Isotropic

Isotropy

The same in every direction. Here, a stream whose energy is spread evenly, with no direction holding much more than another.

Principal directions

Principal direction

Where the energy of the states sits: the directions along which the states hold the most energy, found as the eigenvectors of the second-moment matrix of the states (the average outer product of each state with itself, a covariance without the mean taken out).

Kurtosis

How heavy-tailed the values along a direction are. A direction where most states are near zero and a few are large has a high kurtosis; a Gaussian has 3.

Equivariant

Equivariance

Rotating the input rotates the output the same way. Here, rotating the residual coordinates rotates the update along with them, so no basis is special.

---

1. Participation ratio: the number of directions a set of vectors effectively spreads over, from the variance along each principal direction: 1 when all of it is on one, 64 when it is even over all 64. ↩

2. Second moment: the running average of the squared gradient that Adam keeps for each parameter, and divides its step by. ↩