# A. Supplementary materials

## A.1. Feature values

The ten colors used in our experiments are: gray, red, blue, green, brown, purple, magenta, yellow, orange, pink. The ten shapes are: cube, sphere, cylinder, pyramid, cone, torus, rectangular box, ellipsoid, octahedron, dodecahedron. And the ten textures are: metal, rubber, chainmail, marble, maze, metal weave, polka dots, rug, bathroom tiles, wooden planks. See additional example images below:

**Figure 1:** Additional example training images

## A.2. Curriculum ablations

To explore the sensitivity of our results to our formulation of the coreset as a curriculum, we devised two alternative approaches, beyond the baseline approach reported of allocating half of the examples in each epoch to the newest task, and splitting the remaining examples evenly between previously learned tasks. With all curricula explored below, we make an effort to attain the same balance both within each epoch as a whole, and within each batch in the epoch. We also maintained the same evaluation protocol as with the baseline curriculum to make sure the results are comparable.

### A.2.1 Ratio curriculum

In this curriculum, we again allocate half of the examples in each epoch to the newest task. We split the remaining half such that the ratio between examples allocated to task $k$ and task $k+1$ is $1:\kappa$, where $\kappa \geq 1$ is a free parameter. This allows newer tasks, which the model learned fewer times, to receive proportionally more instruction than previous tasks. Note that for the first two episodes, this behaves identically to the baseline curriculum we investigated. We explored two settings of $\kappa$: the first, $\kappa = 1.24$, we approximated from the data collected under our baseline curriculum, and a second, $\kappa = 1.5$, we chose to explore the sensitivity to this parameter.

For example, with $\kappa = 1.25$, the third episode, the first task receives 10000 examples, the second 12500, and the third, and newest task, receives its 22500 examples in each epoch. For another example, in the sixth episode, the tasks are allocated the following numbers of examples per epoch: 2741, 3427, 4284, 5355, 6693, and 22500. With $\kappa = 1.5$, the third episode allocations are 9000, 13500, and 22500, and the sixth episode training example allocations are 1706, 2559, 3839, 5758, 8638, and 22500.

### A.2.2 Power curriculum

In this curriculum, unlike the baseline and the ratio curriculum we *omit* the allocation of half of the examples to the newest task. In episode $k$, for task $1, 2, \ldots t, \ldots, k$, we first compute unnormalized proportions using a power function: task 1 receives $\rho_1 = k^{-\alpha}$, task 2 receives $\rho_2 = k - 1^{-\alpha}$, and so on, such that each task $t$ receives a proportion of $\rho_t = (k - t + 1)^{-\alpha}$ of the examples. We then normalize the proportions, such that each task receives $p_t = \frac{\rho_t}{\sum_{i=1}^{k} \rho_i}$ of the 45000 total training examples per epoch. As with the ratio curriculum, we first estimated $\alpha$ from the baseline curriculum data, arriving at $\alpha = 1.14$, and then explored an alternative setting of $\alpha = 2$. For example, with $\alpha = 1.14$, in the third episode, task 1 receives 7394 training examples, task 2 receives 11738 examples, and task 3 (the newest task) receives 25868 examples. In the sixth episode, tasks 1-6 are allocated the following numbers of examples: 2611, 3215, 4146, 5755, 9137, and 20136. With $\alpha = 2$, in the third episode we allocate 3674, 8265, and 33061 training examples to tasks 1-3 respectively, and in the six episode, we allocate tasks 1-6 838, 1207, 1886, 3353, 7543, and 30173 training examples respectively.

### A.2.3 Results

We reproduced Figure 3 with the data from running two Latin square replications in each dimension (resulting in sixty total runs) utilizing each curriculum. The results in panels (a) and (b) are noisier, especially for above 256k examples, as most runs finish sooner, and thus there is less data in in that part of the plot. Qualitative, we find panels (c)-(f) quite similar to the baseline curriculum, especially for the two data-optimized versions of the curriculum (Figure 2 and Figure 4), and slightly less so for the misspecified versions (Figure 3 and Figure 5).

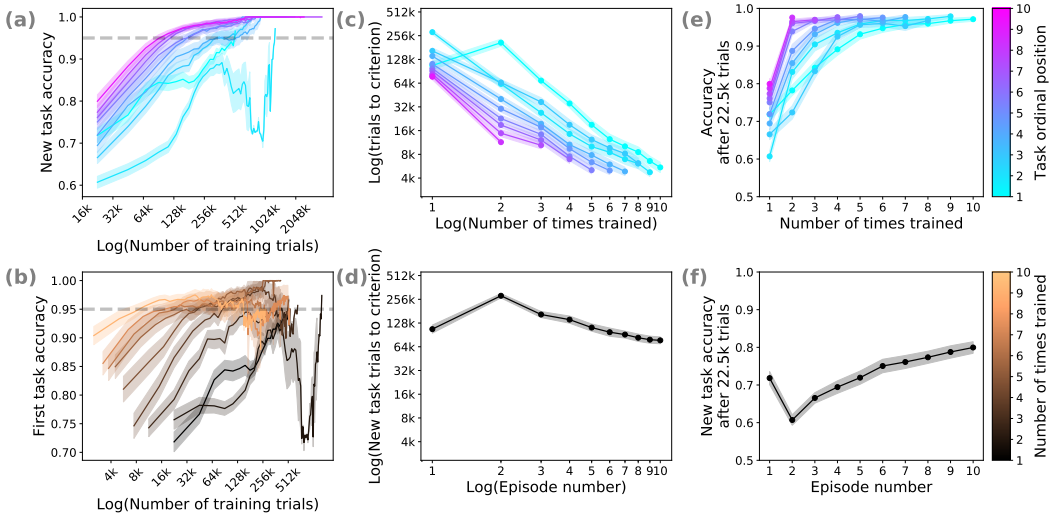


**Figure 2:** Reproduction of Figure 3 with the data from the ratio curriculum using a ratio of $\kappa = 1.25$ approximated from the baseline curriculum data.

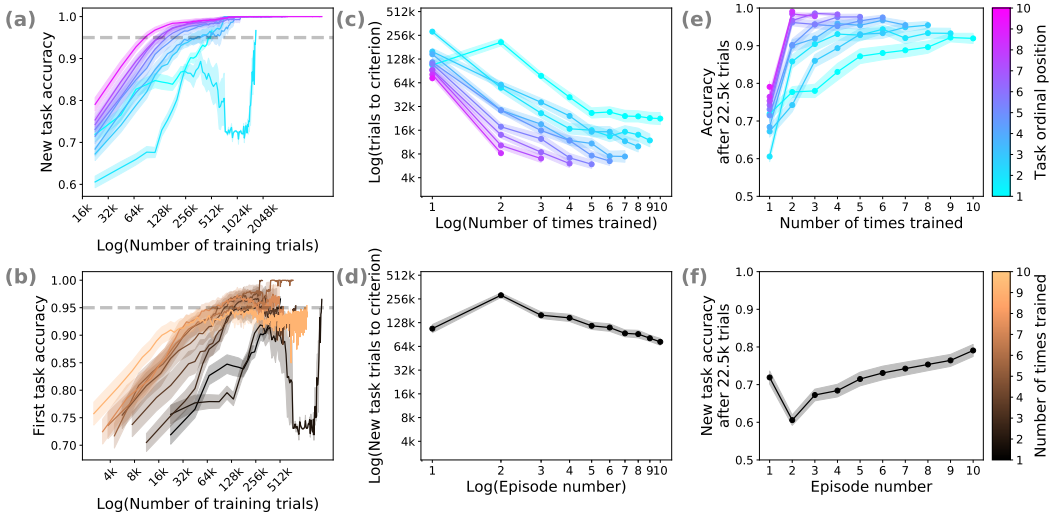


**Figure 3:** Reproduction of Figure 3 with the data from the ratio curriculum using a ratio of $\kappa = 1.5$ designed to explore the sensitivity to the choice of parameter.

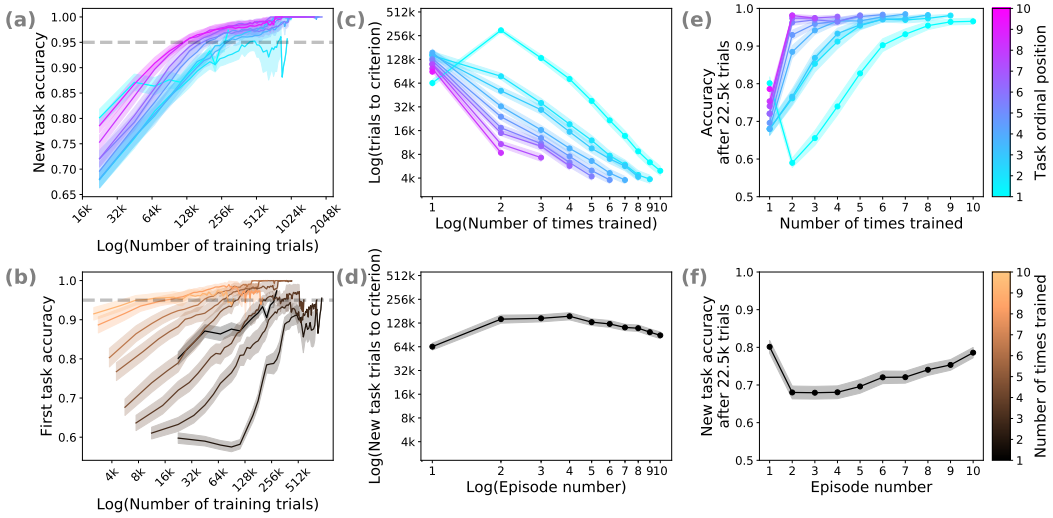


**Figure 4:** Reproduction of Figure 3 with the data from the power curriculum using an exponent of $\alpha = 1.14$ approximated from the baseline curriculum data.

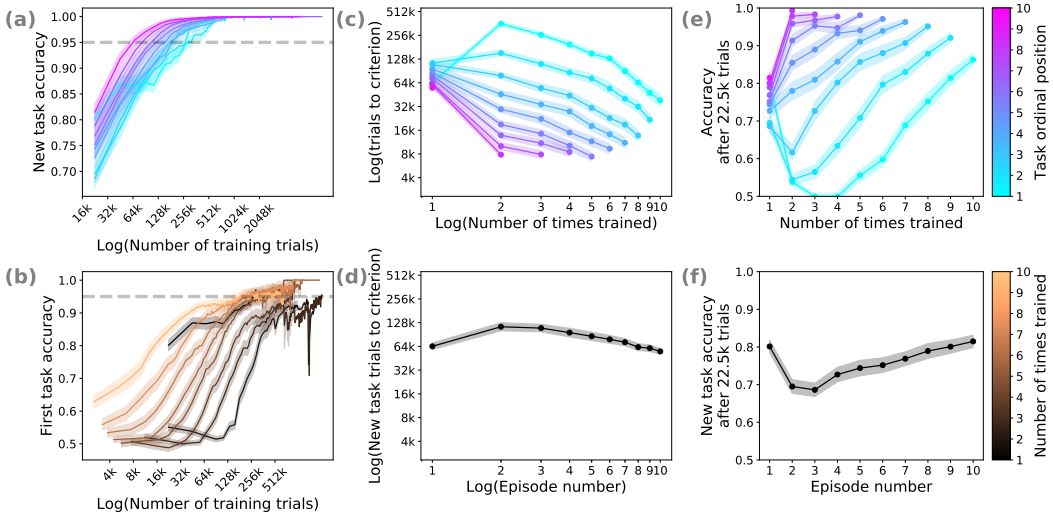


**Figure 5:** Reproduction of Figure 3 with the data from the power curriculum using an exponent of $\alpha = 2$ designed to explore the sensitivity to the choice of parameter.

### A.3. Simultaneous training architecture capacity

We compared a few variations on our model architecture to validate it has the capacity to learn these tasks when trained simultaneously. The loss and AUC curves plotted below provide results from a baseline model, a model with dropout, and the model reported in the paper, which utilizes weight decay but not dropout.

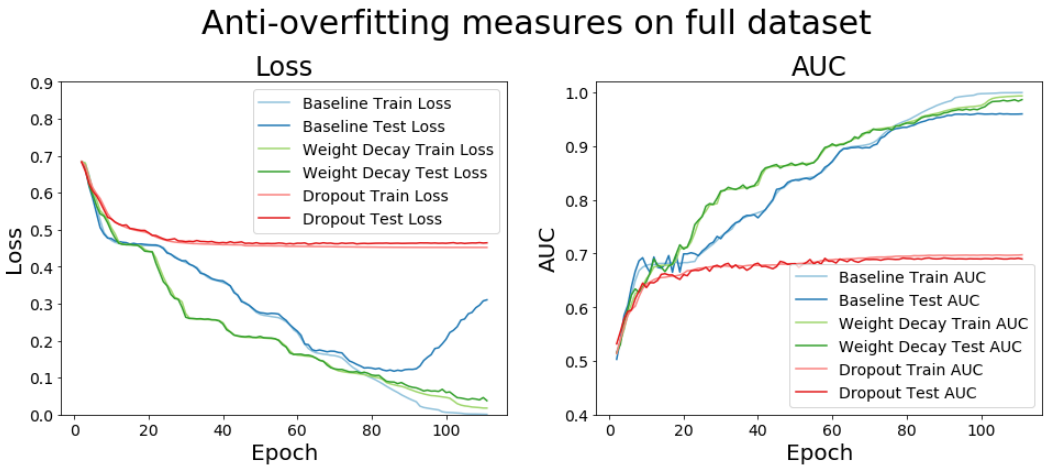


**Figure 6:** Loss and AUC curves for architecture variants when training on all thirty tasks simultaneously.

## A.4. Results by dimension

To justify our collapsing of the results across dimensions, we provide the results broken down for each individual dimension below. Figure 7 depicts the trials required to reach the accuracy criterion, Figure 7g,h reproducing Figure 3c,d, and the rest of the subfigures offering the results for replications within each dimension. While colors are easier to learn than shapes or textures, simulations in all three dimensions show the same qualitative features. Similarly, Figure 8 depicts the accuracy after a fixed small amount of training, with Figure 8g,h reproducing Figure 3e,f. These results provide further evidence for the ease of learning color compared to the other two dimensions, but the qualitative similarity remains striking.

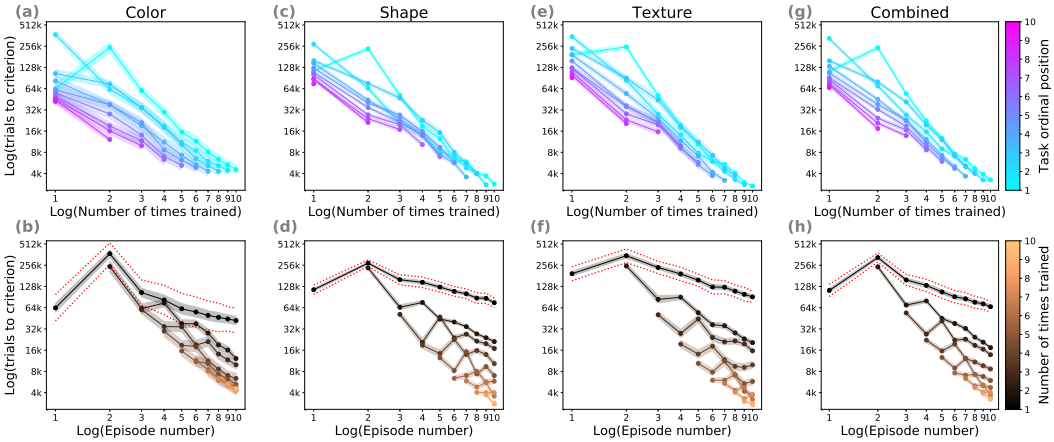


**Figure 7: (a, c, e, g)**: Number of trials required to reach the accuracy criterion (log scale) as a function of the number of times a given task is trained (also log scale). The colored lines indicate task ordinal position (cyan = introduced in episode 1; magenta = introduced in episode 10). **(b, d, f, h)**: Number of trials required to reach the accuracy criterion (log scale) as a function of the episode number. The colored lines indicate the number of times a task was retrained on (black = 1 time, copper = 10 times). In all panels, the shaded region represents ±1 standard error of the mean.

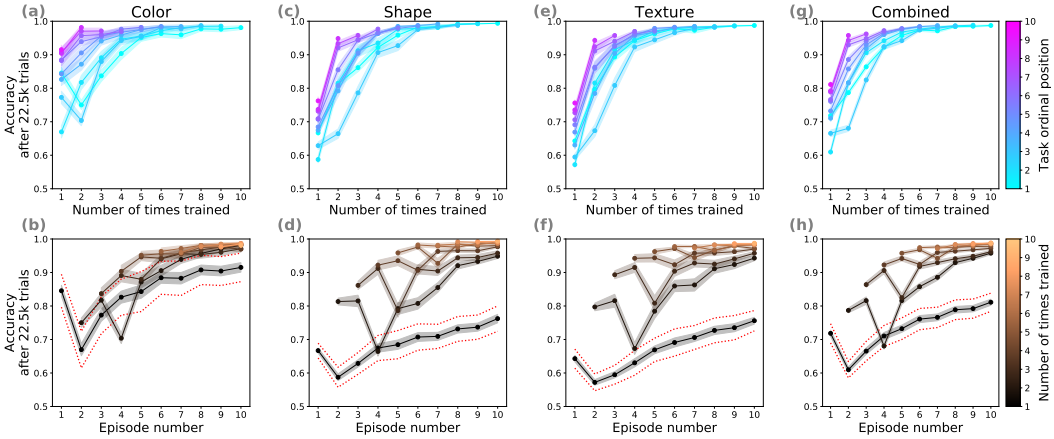


**Figure 8: (a, c, e, g)**: Accuracy after a fixed amount of training (22,500 trials) as a function of the number of times a given task is trained (log scale). The colored lines indicate task ordinal position (cyan = introduced in episode 1; magenta = introduced in episode 10). **(b, d, f, h)**: Accuracy after the same fixed amount of training as a function of the episode number. The colored lines indicate the number of times a task was retrained on (black = 1 time, copper = 10 times). In all panels, the shaded region represents ±1 standard error of the mean.

## A.5. Task-modulated processing at different levels

All figures reported below are combined over replications in all three dimensions, where for each modulation level we performed thirty simulations in each dimension, yielding ninety simulations in total for each modulation level. In Figure 9, we provide the results plotted in Figure 5a-b for task-modulation at each convolutional layer (separately). In Figure 10, we provide equivalent plots to Figure 2e-f for the task-modulated models. In Figure 11, we provide equivalent plots to Figure 5c-d for the task-modulated models. The only anomaly we observe is in Figure 11 for task-modulation at the second convolutional layer, where the eight and ninth tasks appear easier to learn for the first time without task-modulation. Save for this anomaly, we observed remarkably consistent results between the different modulation levels, and hence we reported a single one, rather than expanding about all four.

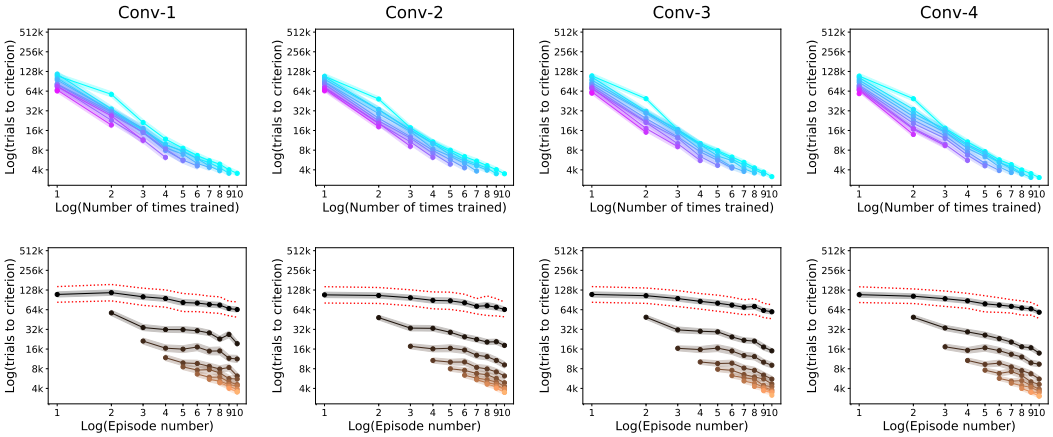


**Figure 9: Top panels:** Number of trials required to reach the accuracy criterion (log scale) as a function of the number of times a given task is trained (also log scale). The colors indicate task ordinal position (the episode in which a task is introduced; cyan = introduced in episode 1; magenta = introduced in episode 10). **Bottom panels:** Similar to the top panels, but graphed as a function of episode number with the line colors indicating the number of times a task is retrained (black = 1 time, copper = 10 times).

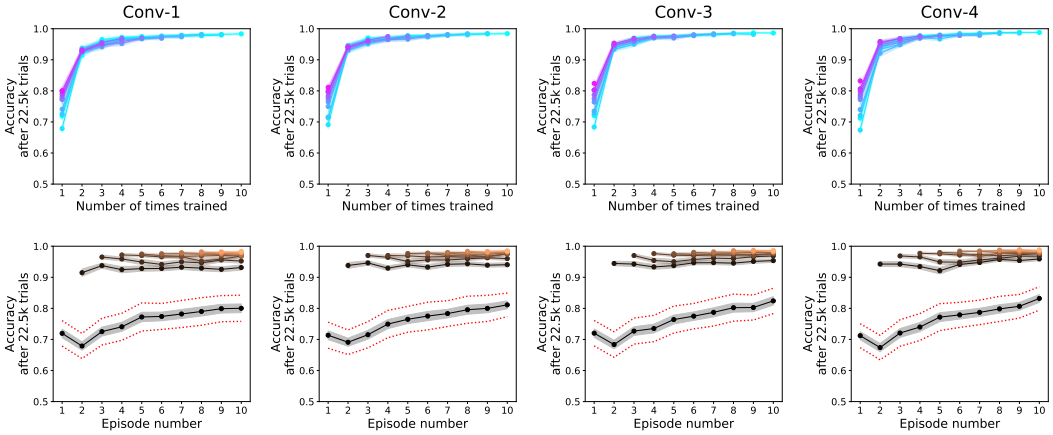


**Figure 10: Top panels:** Hold-out accuracy attained after a fixed amount of training (22.5k trials) of a given task, graphed as a function of number of times a given task is trained. As in Figure 9, the colors indicate task ordinal position (the episode in which a task is introduced; cyan = introduced in episode 1; magenta = introduced in episode 10). **Bottom panels:** Similar to the top panels, but graphed as a function of episode number with the line colors indicating–as in Figure 9–the number of times a task is retrained (black = 1 time, copper = 10 times).

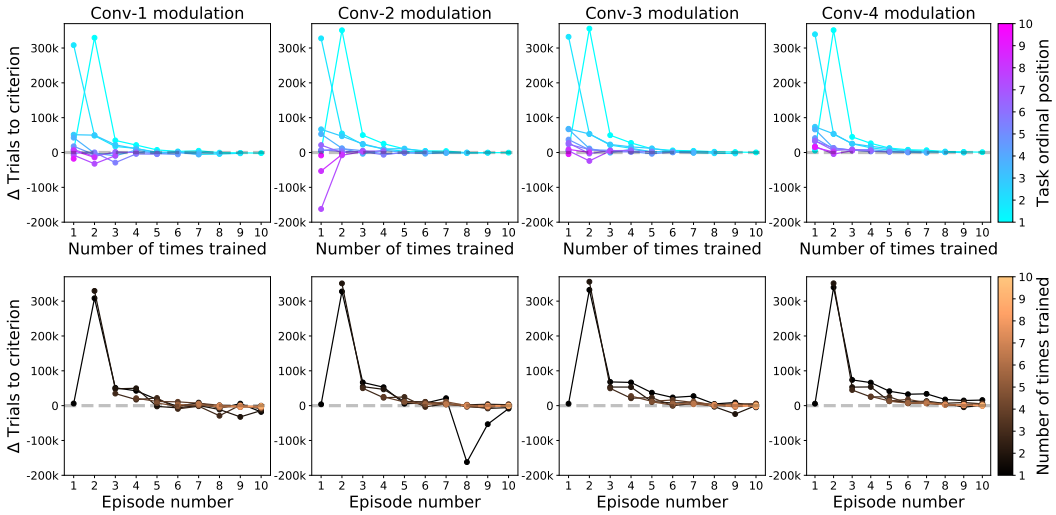


**Figure 11: Top panels:** Increase in number of trials required to reach accuracy criterion for non-task-modulated versus task modulated architectures as a function of the number of times a given task is trained (also log scale). The colors indicate task ordinal position (the episode in which a task is introduced; cyan = introduced in episode 1; magenta = introduced in episode 10). **Bottom panels:** Similar to the top panels, but graphed as a function of episode number with the line colors indicating the number of times a task is retrained (black = 1 time, copper = 10 times).

## A.6. MAML comparison supplement

We compared our baseline model to two versions of MAML. Both utilized the training procedure we describe under the 'Comparison to MAML' section. The first, reported in the middle column below, only utilized this procedure in training, and was tested without the meta-testing step. In other words, this model was tested exactly as our baseline model was tested, to see if MAML manages to learn representations that allow it to answer questions on unseen images without further adaptation. The second version, which we ended up reporting, also utilizes the micro-episode procedure at test time, making train and test identical. The results below demonstrate similar qualitative behavior between our baseline and both versions. However, as the second version, using the meta-testing procedure, fares better, we opt to report it in the submission.

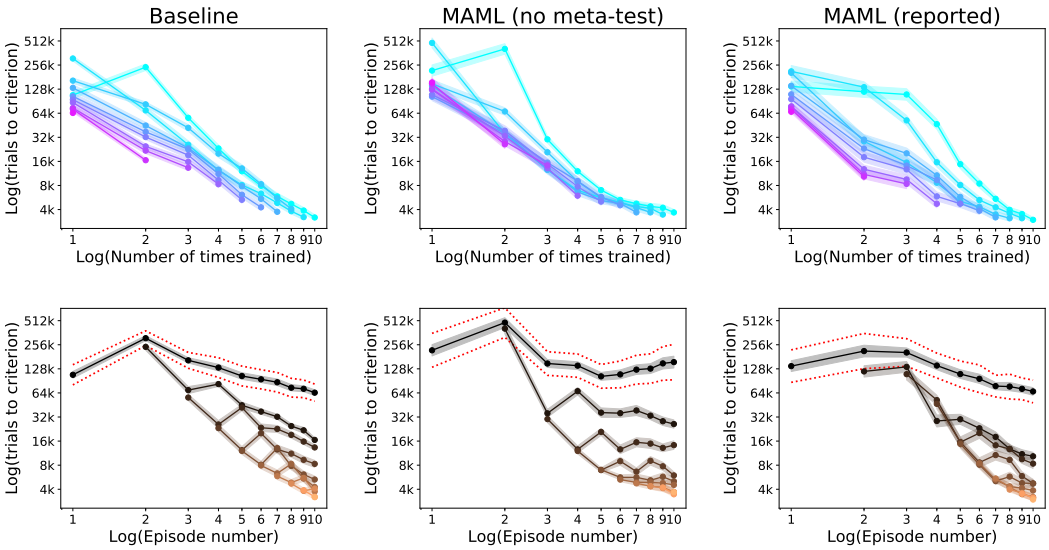


**Figure 12:** MAML comparison. The left column plots result from our baseline condition. The middle column offers results from a version of MAML which did not follow the micro-episode procedure at test that, that is, did not meta-test. The right column, corresponding to the model reported in the submission, follows the micro-episode procedure we describe at both train and test.

## A.7. Comparison to simultaneous learning

We performed a systematic comparison between our sequential method of training and the standard supervised learning approach of training on all tasks simultaneously. We know that sequential training is beneficial to humans–every course covers one topic at a time, rather than throwing the entire textbook and mixing all topics from day one. There is also ample evidence for the value of curricular approaches in machine learning, going as far back as [**?** ]. However, curricula in machine learning usually attempt to scaffold tasks from smaller to larger, or easier to harder, following some difficulty gradient. Our results in Figure 13 suggest, surprisingly, that randomly chosen sequential curriculum (that is, random task introduction orderings) can significantly speed up learning in some cases. We find, interestingly, that this effect varies by dimension. While in the shape condition the simultaneous learning is competitive with sequential training, we find that in both texture and color sequential training proceeds much faster. In those cases, the number of training trials required to learn each task when trained sequentially (the cyan-to-magenta curves) is far less than the number of trials required to learn each task when trained simultaneously (the red curve). That is, task $n+1$ is learned far faster following tasks 1–$n$ than simultaneously with tasks 1–10. The long plateau in the color and texture cases appears to suggest some form of initial representation learning which is made more efficient by learning sequentially, rather than simultaneously.

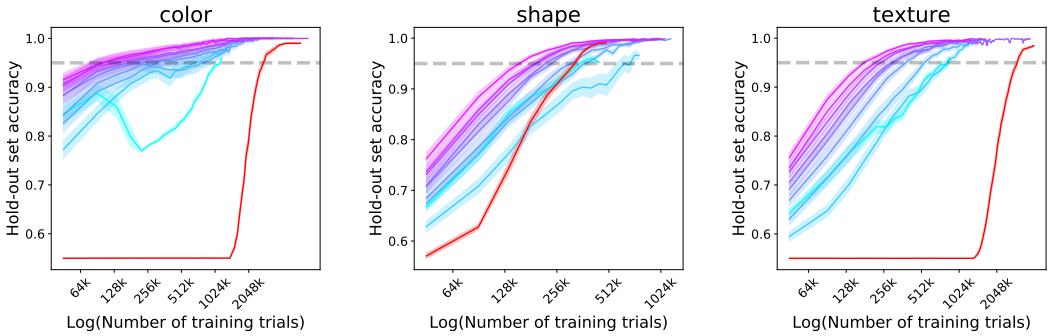


**Figure 13:** Simultaneous vs. sequential training. The cyan (first) to magenta (last) colored lines plot the accuracy after some number of training trials for each task the model learned. The average accuracy over all ten tasks, when learned simultaneously, is plotted in red. To make the comparisons valid, the simultaneous training is in the number of training trials *for each task*, rather than combined for all tasks.