Measuring Abstract Reasoning in Neural Networks

David Barrett*, Felix Hill*, Adam Santoro*, Ari Morcos, Timothy Lillicrap

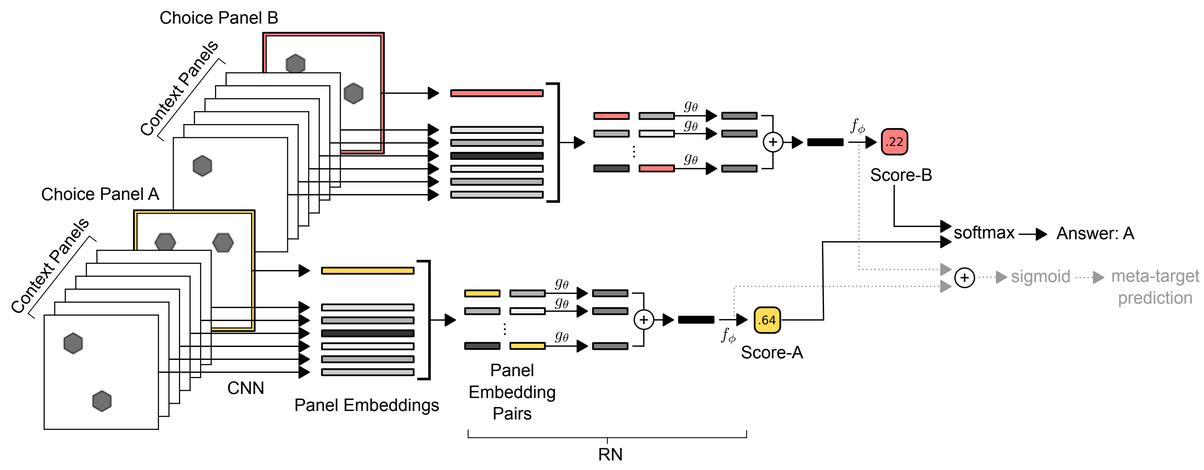
* equal contribution, ordered alphabetically by surname

barrettdavid@, felixhill@, adamsantoro@, arimorcos@, countzero@ | DeepMind, London, United Kingdom

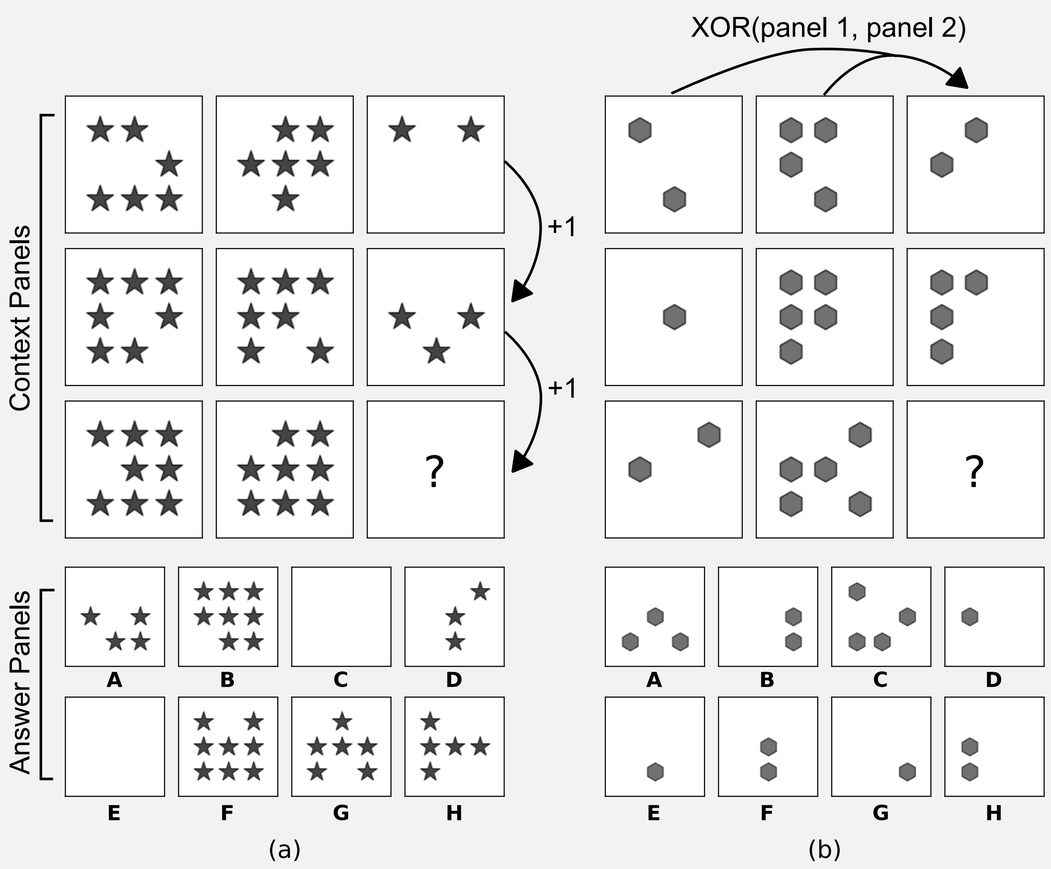
ABSTRACT

Whether neural networks can learn abstract reasoning or whether they merely rely on superficial statistics is a topic of recent debate. Here, we propose a dataset and challenge designed to probe abstract reasoning, inspired by a well-known human IQ test. To succeed at this challenge, models must cope with various generalisation "regimes" in which the training and test data differ in clearly-defined ways. We show that popular models such as ResNets perform poorly, even when the training and test sets differ only minimally, and we present a novel architecture, with a structure designed to encourage reasoning, that does significantly better. When we vary the way in which the test questions and training data differ, we find that our model is notably proficient at certain forms of generalisation, but notably weak at others. We further show that the model's ability to generalise improves markedly if it is trained to predict symbolic explanations for its answers. Altogether, we introduce and explore ways to both measure and induce stronger abstract reasoning in neural networks. Our freely-available dataset should motivate further progress in this direction.

MODEL

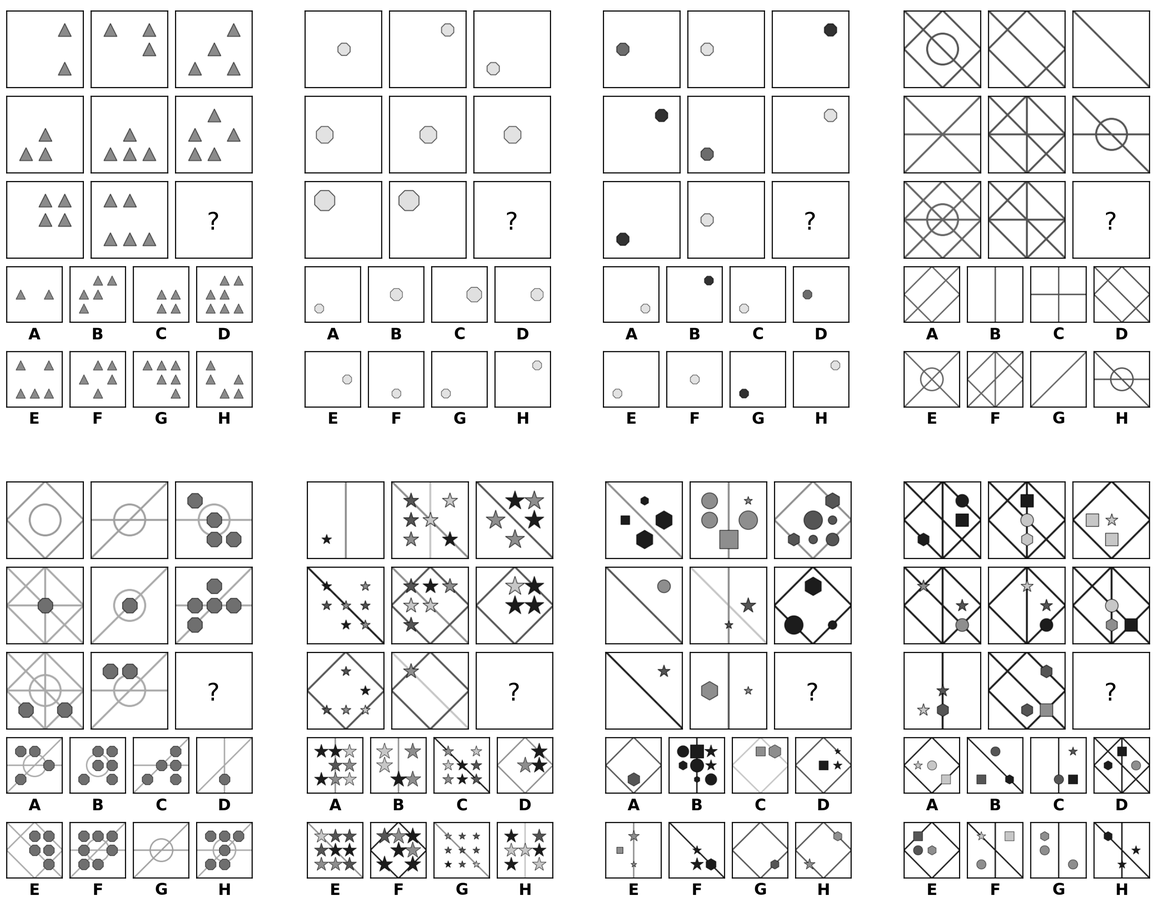


RAVEN'S PROGRESSIVE MATRICES



Raven-style Progressive Matrices. In (a) the underlying abstract rule is an arithmetic progression on the number of shapes along the columns. In (b) there is an XOR relation on the shape positions along the rows (panel 3 = XOR(panel 1, panel 2)). Other features, such as shape type or color, do not factor in. **A** is the correct choice for both. See the appendix for more examples, including some that are quite challenging for humans.

PROCEDURALLY GENERATING MATRICES



GENERALISATION REGIMES

(1) Neutral In both training and test sets, the structures S can contain any triples $[r, o, a]$ for $r \in \mathcal{R}$, $o \in \mathcal{O}$ and $a \in \mathcal{A}$. The training and test sets are disjoint, but this separation was at the level of the input variables (i.e., the pixel manifestations of the matrices).

(2) Interpolation; (3) Extrapolation As in the neutral split, S consisted of any triples $[r, o, a]$. For interpolation, in the training set, when $a =$ `colour` or $a =$ `size` (the ordered attributes), the values of a were restricted to even-indexed members of the discrete set V_a, whereas in the test set only odd-indexed values were permitted. For extrapolation, the values of a were restricted to the lower half of their discrete set of values V_a during training, whereas in the test set they took values in the upper half. Note that all S contained some triple $[r, o, a]$ with $a =$ `colour` or $a =$ `size`. Thus, generalisation is required for every question in the test set.

(4) Held-out Attribute `shape-colour` or (5) `line-type` $\mathcal{S}$ in the training set contained no triples with $o =$ `shape` and $a =$ `colour`. All structures governing puzzles in the test set contained at least one triple with $o =$ `shape` and $a =$ `colour`. For comparison, we included a similar split in which triples were held-out if $o =$ `line` and $a =$ `type`.

6: Held-out Triples In our dataset, there are 29 possible unique triples $[r, o, a]$. We allocated seven of these for the test set, at random, but such that each of the $a \in \mathcal{A}$ was represented exactly once in this set. These held-out triples never occurred in questions in the training set, and every $\mathcal{S}$ in the test set contained at least one of them.

RESULTS

Model	Test (%)
WReN	**62.6**
Wild-ResNet	48.0
ResNet-50	42.0
LSTM	35.8
CNN + MLP	33.0
Blind ResNet	22.4

		$\beta = 0$			$\beta = 10$		
Regime		**Val. (%)**	**Test (%)**	**Diff.**	**Val. (%)**	**Test (%)**	**Diff.**
Neutral		63.0	62.6	-0.6	77.2	76.9	-0.3
Interpolation		79.0	64.4	-14.6	92.3	67.4	-24.9
H.O. Attribute Pairs		46.7	27.2	-19.5	73.4	51.7	-21.7
H.O. Triple Pairs		63.9	41.9	-22.0	74.5	56.3	-18.2
H.O. Triples		63.4	19.0	-44.4	80.0	20.1	-59.9
H.O. `line-type`		59.5	14.4	-45.1	78.1	16.4	-61.7
H.O. `shape-colour`		59.1	12.5	-46.6	85.2	13.0	-72.2
Extrapolation		69.3	17.2	-52.1	93.6	15.5	-78.1

CONCLUSIONS

Neural networks can indeed learn to infer and apply abstract reasoning principles. Our best performing model learned to solve complex visual reasoning questions, and to do so, it needed to induce and detect from raw pixel input the presence of abstract notions such as logical operations and arithmetic progressions, and apply these principles to never-before observed stimuli.

An important contribution of this work is the introduction of the PGM dataset, as a tool for studying both abstract reasoning and generalisation in models. Generalisation is a multi-faceted phenomenon; there is no single, objective way in which models can or should generalise beyond their experience. The PGM dataset provides a means to measure the generalization ability of models in different ways, each of which may be more or less interesting to researchers depending on their intended training setup and applications.