Journal of Artificial Intelligence Research 64 (2019) 529-562 Submitted 01/17; published 03/19

AI Generality and Spearman's Law of Diminishing Returns

José Hernández-Orallo JORALLO@DSIC.UPV.ES
DSIC, Universitat Politècnica de València
Cami de Vera s/n, 46022 Valencia, Spain

Abstract

Many areas of AI today use benchmarks and competitions with larger and wider sets of tasks. This tries to deter AI systems (and research effort) from specialising to a single task, and encourage them to be prepared to solve previously unseen tasks. It is unclear, however, whether the methods with best performance are actually those that are most general and, in perspective, whether the trend moves towards more general AI systems. This question has a striking similarity with the analysis of the so-called positive manifold and general factors in the area of human intelligence. In this paper, we first show how the existence of a manifold (positive average pairwise task correlation) can also be analysed in AI, and how this relates to the notion of agent generality, from the individual and the populational points of view. From the populational perspective, we analyse the following question: is this manifold correlation higher for the most or for the least able group of agents? We contrast this analysis with one of the most controversial issues in human intelligence research, the so-called Spearman's Law of Diminishing Returns (SLODR), which basically states that the relevance of a general factor diminishes for most able *human* groups. We perform two empirical studies on these issues in AI. We analyse the results of the 2015 general video game AI (GVGAI) competition, with games as tasks and "controllers" as agents, and the results of a synthetic setting, with modified elementary cellular automata (ECA) rules as tasks and simple interactive programs as agents. In both cases, we see that SLODR does not appear. The data, and the use of just two scenarios, does not clearly support the reverse either, a Universal Law of Augmenting Returns (ULOAR), but calls for more experiments on this question.

1. Introduction

A good number of AI competitions and benchmarks are encouraging the development of more general-purpose AI systems (Hernández-Orallo, 2016). Examples of this trend are the general game playing AAAI Competition (Genesereth, Love, & Pell, 2005; Genesereth & Thielscher, 2014), the reinforcement learning competition (Whiteson, Tanner, & White, 2010; Dimitrakakis, Li, & Tziortziotis, 2014) (including, e.g., the 'polyathlon', with several domains), the genetic programming benchmarks (McDermott et al., 2012; White et al., 2013), the general video game competition (Schaul, 2014; Perez et al., 2015), and the arcade learning environment (ALE) (Bellemare, Naddaf, Veness, & Bowling, 2013), a collection of Atari 26000 video games. For instance, "the ALE has incentivized the AI community to build more generally competent agents" (Machado et al., 2017) and, as a result, techniques such as deep reinforcement learning are recently achieving a remarkable overall performance (Mnih et al., 2015). Although the systems have to relearn from scratch when the game changes slightly, they are still based on a general-purpose technique (deep reinforcement learning), which can be evaluated for a range of tasks.

Following the idea that a larger and wider variety of tasks prevents AI techniques from specialising to a single task, some new platforms, such as OpenAI's Gym, Microsoft's Malmo, OpenAI's Universe or DeepMind's Lab (Castelvecchi, 2016), allow AI researchers to construct benchmarks with hundreds of different tasks. However, it is still possible that some agents outperform the state-of-the-art techniques on average by specialising on a subset (a bucket) of the tasks, without really featuring any improvement on the rest. This would represent an overall performance gain as the aggregation of a series of particular techniques (or tricks), only useful for specialised subgroups of tasks.

This discussion is especially controversial when one wants to consider *all possible tasks*. On one hand, one can consider every possible problem's output as equally likely. This is technically known as "block uniformity" (Igel & Toussaint, 2005), with the uniform distribution being a special case. Under this assumption, we have the conditions for the so-called no-free-lunch theorems (Wolpert & Macready, 1995; Wolpert, 2012), leading to the conclusion that, on average, no method can be better than any other. According to this, a general-purpose system and, indeed, the very concept of 'general intelligence' would be impossible (Edmonds, 2009). On the other hand, one can consider problems as programs. In this case, a uniform distribution is not possible, but any universal distribution can be assumed instead, which leads to the theory of universal prediction using algorithmic probability, developed by Solomonoff in the 1960s (Solomonoff, 1960, 1964). This has influenced several approaches based on algorithmic information theory about how tasks can be generated —and weighted— in definitions of intelligence (Dowe & Hajek, 1997; Hernández-Orallo, 2000; Legg & Hutter, 2007; Hernández-Orallo & Dowe, 2010; Dowe, Hernández-Orallo, & Das, 2011; Insa-Cabrera, Dowe, & Hernández-Orallo, 2011). Under this setting, theoretically general agents can be defined (Hutter, 2005) —only if weakly optimal or suboptimal in general (Orseau, 2013; Leike & Hutter, 2015)— with some approximations working in practice (Veness, Ng, Hutter, & Silver, 2011). In this context, one particular way of reconciling different views of generality is known as *policy-general intelligence*, which is based on the assumption of a uniform distribution over *solutions* for each task difficulty, and aggregating for a range of difficulties (Hernández-Orallo, 2017). Under this setting, the idea of general intelligence makes sense theoretically: some agents can be better than others *in general*.

This debate replicates the controversy in psychometrics between the IQ scores (results of the IQ tests, which depend on the task distribution used in a test) and the g scores (a magnitude derived from the estimated value of a latent factor, the g factor, which is more independent of the particular task distribution used in a test). The g factor derives from the so-called positive manifold, one of the most replicated experimental findings in the analysis of human intelligence. The positive manifold indicates that given any test composed of a set of (abstract) cognitive tasks we will find a high correlation in the results produced by a human population. In other words, those who perform well on some tasks will usually perform well on any other. This supports the idea of general intelligence.

For artificial intelligence, this suggests the following question: if we aim at building more general AI systems, and evaluate them on a really wide range of tasks, will it be the case that those that are better for some tasks will also be better for other tasks? Is that a necessary or a contingent *property*? Or can we have more and more powerful systems without an increase in their generality?

In order to analyse these questions, we need to clarify what generality is and how it can be analysed. We can define a general agent as one that gets *similar success (either usually good or usually bad) for a wide range of tasks*. Under this view, given a set of tasks, perfect performance on all of them would imply generality. But this would also imply that systems performing consistently poorly would be general. In other words, the two extreme cases, the agent that is perfect for all tasks and the agent that is calamitous for all tasks, would be both completely general, but very different in terms of ability. The interesting cases arise for those agents that are neither perfect nor calamitous. In particular, we are interested in agents that are not necessarily perfect for any task, but reasonably good for all. This, in the context of intelligence, is sometimes referred to as being "second-best" at everything, in contrast to the so-called "idiots savants", which are the best at a very specific range of things.

The notion of generality can be analysed *individually* for a particular system, or in the context of a population or group of systems. The individual approach can be easily translated into a view of the variation of results for a single agent, measured by the variance of results or any other measure of dispersion. However, as we will discuss, this is problematic, as it depends on the magnitude and distribution of results. In some special, but common, cases, like when responses are binary, the variance would be a function of the mean, and the concepts of ability and generality would be completely intertwined.

The issue of generality can also be analysed for a *population* of agents. A populational analysis can give a different insight compared to an individual generality measure. Still, it might be the case that, again, depending on the magnitude of the scores, less generality would be found in the intermediate agents. This populational approach is common in psychology, especially in psychometrics, where the positive manifold originated.

One question we want to study is whether we find a positive manifold in populations of artificial agents in two situations: populations of AI systems originating from AI competitions (so being built and selected by humans) and populations of synthetic agents created automatically for a range of abstract tasks (without human intervention). A second, related, question we would like to study is whether the positive manifold may only start to be observed for artificial agents when we have a population of minimally intelligent agents. A new hypothesis would be that, as long as AI progresses towards more generally intelligent agents, this positive manifold would start to appear and then become stronger. By studying some existing competitions and synthetic scenarios, we can only partially analyse whether this is the case or not, by analysing if the positive manifold is stronger for more able agents.

Surprisingly, regarding the second question in the realm of human intelligence, Spearman found exactly the reverse observation. By taking subpopulations of more able humans, the positive manifold was weaker, something that was later known as Spearman's Law of Diminishing Returns (SLODR). In other words, it looked as if humans could achieve more overall performance by the integration of advanced specific skills rather than the improvement of general skills. As this hypothesis has been strongly disputed and goes against the rationale discussed above (generality growing for high ability), it would then be extremely illuminating for AI —but also for human intelligence research— to see whether SLODR holds for AI systems or not.

In this paper, we study two different settings to experimentally analyse these questions for artificial agents. First, we take the results of the 2015 general video game AI (GVGAI) competition, one of the AI competition that most strongly encourages generality in AI. For this setting, video games are the tasks whereas the "controllers" (AI algorithms) will compose the agent population, so that mean task correlation will be analysed globally and by agent ability bins. Second, we adapt a class of elementary cellular automata (ECA) to create worlds (the ECA rules) with agents interacting inside these worlds (Hernández-Orallo, 2015a). Using these simple worlds and an elementary agent language, we can analyse *all tasks* and all possible policies (solutions) up to a certain size (determining the difficulty of the policy), so really having a diversity of solutions in order to analyse whether some degree of positive manifold appears. More interestingly, we can easily analyse different subsets according to their average performance on all policies (or slices of appropriate difficulty) and study whether the SLODR holds or not.

The rest of the paper is organised as follows. Section 2 introduces some formal definitions about the indicators that will be used throughout the paper, reviews the notions of positive manifold, explains the g factor, and discusses Spearman's Law of Diminishing Returns (SLODR) in the context of the related literature supporting or rejecting the law. Section 3 considers the 2015 general video game AI (GVGAI) competition and studies the existence of a positive manifold and whether this generality becomes stronger or not for the most able subgroups, in agreement (or not) with SLODR. Section 4 introduces the environments and agents used in a second, synthetic setting. We also explore whether generality (the positive manifold) takes place and whether SLODR holds or not. Finally, section 5 closes the paper with a discussion of the results, its applicability, new questions and future work.

2. The Positive Manifold and Spearman's Law of Diminishing Returns

Let us introduce some notation first. Consider the evaluation of a set of M agents on a set of N tasks, with results or responses $R_{i,j}$, for each agent π_i and task μ_j, as represented in Table 1. From this matrix, several statistics derive easily. Looking at the rows, for each agent we have its response mean $\bar{R}_i = \text{Mean}_j[R_{i,j}]$, also referred to as *agent average performance*, and its response variance $\sigma_i^2 = \text{Var}_j[R_{i,j}]$, also referred to as *agent variance.* These two indicators (agent performance and variance) can be understood as a proxy for ability and generality respectively (the higher the variance the lower the generality). However, both are dependent on the magnitude of results. For instance, if tasks have a different result magnitude, the mean $\bar{R}_i$ will average non-commensurate measurements. One option is to scale the columns, such that their scores follow a standard distribution.

Things are still especially complicated for the agent variance, if we want to consider it as a measure of generality. For instance, given two agents a and b scoring $3, 3, 2, 3$ and $6, 6, 4, 6$ respectively, we see that b has a higher variance, just because the magnitude of the responses is higher, although we do not see more variability than for the first agent (actually, the scores for agent b are just the double of those for agent a). The variance can be normalised through the *coefficient of variation* $\frac{\sigma_i}{\bar{R}_i}$. However, this coefficient is also problematic. First, if the values are not in a ratio scale, and there are negative scores, the measure still depends on the scale (and normalising per rows does not make sense). Second, the kinds of values of the response are critical. For instance, if responses are binary, we have a Bernoulli distribution, and the variance is $\bar{R}_i(1-\bar{R}_i)$. In this case, both the variance and the coefficient of variation would be determined by the mean, so we would have that performance and generality could

Table 1: Usual configuration of the response matrix $R_{i,j}$ with columns representing the tasks (also referred to as tests, rules or games) and the rows representing the agents (also referred to as individuals or controllers). Correlations will always be between tasks (columnwise).

	μ_1	$\cdots$	μ_N
π_1	$R_{1,1}$	$\cdots$	$R_{1,N}$
$\vdots$	$\vdots$	$\ddots$	$\vdots$
π_M	$R_{M,1}$	$\cdots$	$R_{M,N}$

not be disentangled. When responses are non-binary (either originally or on expectation), the connection can also be tight, depending on the distribution for real-valued responses and the number of possible values for ordinal responses. We will explore this graphically in the following sections. Finally, an indicator that has an important effect on the populational analysis is the variance of all agent performances, i.e., $Var_i[\bar{R}_i]$. Actually, this is at the core of whether there are actual differences in the population.

If we now look at the columns, we can calculate the linear (Pearson) correlation between tasks, denoted by $\rho_{a,b}$ (for tasks μ_a and μ_b). A high pairwise correlation between two tasks is understood as having many agents scoring similarly for both tasks; basically both tasks measure the same thing for that agent population. From here, we can derive an $N \times N$ correlation matrix $\rho_{a,b}$. The average correlation is simply defined as follows:

$$\bar{\rho} = \sum_{a>b} \frac{2\rho_{a,b}}{N(N-1)}$$

All these indicators were already examined more than a century ago by Charles Spearman. He became one of the pioneers of a numerical analysis of human intelligence, by compiling the results of several tests on human populations. He found one important phenomenon; when he analysed a set of different tests taken by the same population, he found a positive average correlation in their results ($\bar{\rho} \gg 0$). In other words, the individuals that obtained good results for some tests usually obtained good results for the rest. This correlation was stronger the more culture-fair and abstract the tests were. This phenomenon was known as the 'positive manifold' (Spearman, 1904, 1927), meaning that most of the tests measure the same thing for the agents in the population. It is important to clarify that this phenomenon is not a property of the tests (tasks) alone nor a property of the population (agents) alone. A correlation is clearly an effect that takes place for two tests for a set of subjects, but the average correlation is calculated from the correlation matrix, thereby involving both the tests and the population. Nevertheless, the positive manifold appeared again and again for different human populations and different sets of tests, provided they were not too linked to particular cultural or educational backgrounds (e.g., a chess-playing test or a Korean vocabulary test). Spearman tried to understand the findings through the invention of a rudimentary factor analysis. He identified a dominant *latent factor* that explained much of the subjects' variance, and called it the g factor. Since then, this factor has been one of the most relevant (and replicated) findings in psychometrics (Jensen, 1998; Sternberg, 2000) and has been found to predict many facets of life, from academic performance to (lack of) religiosity in humans. The dominance of g and its explanatory character for the positive manifold led to the association of g with general intelligence, a latent factor that was said to pervade all other factors and facets of intelligence. Of course, this interpretation has been challenged many times, even if g appears again and again.

More controversial than the interpretation of the g factor is another finding that Spearman discovered. He calculated the strength of g on subpopulations of different abilities. In particular, in one of the analysis, he separated the results of several tests on a human population into two groups, group A with normal abilities and B with low abilities. After the split, he analysed the correlation matrices separately. The result was that the mean correlation for group A was 0.47 but the mean correlation for group B was 0.78. Note that this does not mean that group A had worse results (in fact, it was precisely the group with highest average results), but rather that the *proportion of the variance* explained by g for the low-ability group was much higher than for the normal-ability group. This result was striking, especially if g is understood as general intelligence. It looked as if the more intelligent a population is, the less important g would be, in relative terms, to explain its variability. This observation turned to be known as Spearman's Law of Diminishing Returns (SLODR). The finding was replicated many times since then with different experimental settings (Detterman & Daniel, 1989; Deary et al., 1996; Tucker-Drob, 2009).

Spearman looked for an explanation and found it in the *law of diminishing returns* in economics. Many processes that are affected by many factors do not grow continuously as the result of the increase of one factor, so the influence of a single, albeit dominant, factor can become less relevant at a given point, being saturated. Spearman expressed it in this way: "the more 'energy' a person has available already, the less advantage accrues to his ability from further increments of it" (Spearman, 1927, p. 219).

But this simile was not an explanation. Spearman postulated the "ability level differentiation", which considered that challenging items (those that can only be solved by the most able individuals) require the combination of many skills, and the prevalence of g would be smaller. Basically, for the easy items, the general intelligence or some general resources would be the only available skills for low-ability subpopulations. Detterman and Daniel (1989) argued similarly that if "central processes are deficient, they limit the efficiency of all other processes in the system. So all processes in subjects with deficits tend to operate at the same uniform level. However, subjects without deficits show much more variability across processes because they do not have deficits in important central processes". Other explanations were introduced, such as that the "genetic contribution is higher at low-ability levels" (Deary et al., 1996).

Not only have the above explanations been put into question but the experimental evidence itself has been contested. One common counter-explanation of the phenomenon argues that it is not that g is less important for able subjects, but that they find many of the problems in the tests less challenging than the normal population so they are not forced to use general intelligence. They can solve the problems without (deep) thinking, i.e., more mechanically. In other words, the use of the same tests for both groups would be creating the effect. In fact, Fogarty and Stankov (1995) performed an experiment where the more able group had to solve problems of higher difficulty whereas the less able group had to solve problems of lower difficulty. Under these conditions not only did SLODR vanish but even the more able group showed higher correlations. This seems to agree with the idea that general intelligence is used when the individual finds a problem challenging. It is important, however, to check that the difficult problems are created without the use of spurious complications, in order to prevent that more difficult items are more specialised than the simple items. For instance, in number series problems, one can create a complex series by using the Fibonacci series. This, however, will just assess whether the subject has some particular mathematical knowledge, not really expecting that the subject is going to discover the Fibonacci series from scratch by combining the basic arithmetic operators. This was already warned by Jensen, pointing out "that it is the highly g-loaded tests that differ the least in their loadings across different levels of ability, whereas the less g-loaded tests differ the most" (Jensen, 2003). Usually, problems featuring abstract thinking (inductive inference, analogies, etc.) are those with higher g loadings.

One of the most relevant criticisms (or explanations), which will reappear later on in this paper, had a more statistical character. Jensen (1998, p. 587) argued that the subgroups with higher abilities had lower variance than the subgroups with lower abilities. This may be caused by the way the tests are designed to cover a wide range of subjects or the way the two groups are split, but the different variances were generally the case. As a consequence, the relative relevance of g would be lower for more able groups as there is less variance to explain.

All of the above suggests that there are several methodological problems about the analysis of SLODR in human intelligence. This starts with putting into question all the results for which both groups do not have the same variance and also those that include spurious problems or sample the populations in ways to get the same variance by introducing some other confounding factors. In the end, Murray et al. (2013) argue that SLODR could just be "a statistical artifact".

In what follows we take a different perspective of the debate by using non-human subjects, namely AI systems. This can help us to rule out some of the confounding factors by focusing on a popular AI competition first and a controlled experiment second, where we can play with the population of agents and the choice of tasks more freely. Nevertheless, our interest is to analyse whether SLODR happens or not for artificial agents, and see whether the results can tell us something about the construction and evaluation of general-purpose AI agents.

3. Analysing Generality in a General Video Game AI Competition

In the introduction we discussed that many AI evaluation platforms and competitions are now comprising a diversity of tasks, and many new platforms are including hundreds, when not thousands, of different tasks (Castelvecchi, 2016). If success is measured by aggregating or averaging over the range of tasks, an AI algorithm may get a decent result overall by excelling at some subsets (or pockets) of the tasks whereas another can be similarly successful overall by being just relatively good at a larger subset. In other words, generality is of course linked to overall success on a range of tasks but also on whether this success is well distributed.

3.1 General Video Game AI

The name of some competitions seem to explicitly encourage more general AI systems, such as the general game playing (GGP) AAAI Competition (Genesereth et al., 2005; Genesereth & Thielscher, 2014), or the more recent general video game AI (GVGAI) competition (Schaul, 2014; Perez et al., 2015). But are the sets of systems that take part in these competitions really general? And are the best agents more general? In order to analyse these questions, we are going to focus on the latter competition. There are several reasons for this choice. The GVGAI competitions consist of a collection of games over a video game description language. Games in this language can be created at will (even automatically) to cover any kind of task. Also, the GVGAI competition does not give the agents any previous information —before the game starts— about the task. This is different from GGP, which provides the rules of the game in a logical language (Schiffel & Thielscher, 2014). Consequently, in GVGAI, the agents can improve from interaction, a setting that has now become common in many AI evaluation platforms (Castelvecchi, 2016), using a very general reinforcement learning setting. In this regard, GVGAI is closer to the arcade learning environment (Bellemare et al., 2013) mentioned in the introduction, which only gives access to regular screenshots and rewards. The GVGAI setting does not use screenshots as interface but an abstract description of the state, so complex perception is not needed. Also, the setting provides a look-ahead access to the game, so that it can explore different sequences of actions with a perfect forward model of the game —albeit black-box and stochastic. As a result, planning approaches might be more successful than pure RL solutions. More precisely, the video game AI competition provides each algorithm with "a structured representation of the game state as well as a simulator that allows the use of tree search algorithms to play the games" (Bontrager, Khalifa, Mendes, & Togelius, 2016). Because of this access to the simulator, some of the best algorithms so far are based on Monte Carlo Tree Search methods and variants. Figure 1 shows some screenshots of four games created for the platform.

GVGAI is an interesting setting also because there is an ongoing discussion about the need for generality to get good results in the competition. Nielsen et al. have found that "performance is non-transitive", meaning that "different algorithms perform best on different games" of the GVGAI competition (Nielsen, Barros, Togelius, & Nelson, 2015; Bontrager et al., 2016). This conclusion emerges from the observation of results, seeing that no algorithm dominates all the others. However, this is not drawn for looking into whether the correlations between games are positive, even slightly. Indeed, it is important to highlight that the purpose of raising this 'non-transitivity' is not motivated by the interest in finding generality. On the contrary, the motivation of the analysis is to "predict the performance of an algorithm on a task" (Bontrager et al., 2016) so that metalearning through hyper-heuristics and algorithm portfolios is effective (Mendes, Nealen, & Togelius, 2016).

Another important characteristic of the GVGAI competition is that the games are new for the competition, i.e., they are unseen for the participating AI algorithms, also to prevent agents from specialisation to previous games. This is in contrast to other benchmarks and many other competitions, where the teams can tune their techniques and parameters to particular games.

Figure 1: Several games created for the general video game AI (GVGAI) competition. [Images courtesy of Julian Togelius.]

Table 2: The 23 controllers used in the GVGAI experiments.

AIJim	MH2015	Rooot	TeamTopbug
sampleGA	simulatedAnnealing	adrienctx	jaydee
NovTea	SJA862	sampleMCTS	depthFirstSearch
hillClimber	MnMCTS	mrtndwrd	roskvist
SJA86	breadthFirstSearch	evolutionStrategies	sampleRandom
sampleonesteplookahead	aStar	iterativeDeepening	

We will work with a dataset composed of 49 games and the 23 controllers (agents) that were submitted to the 2015 GVGAI competition, as described by Bontrager et al. (2016). Each game has 5 levels, and each level was attempted 5 times. This makes a total of $23 \times 49 \times 5 \times 5 = 28175$ trials. Although there are several metrics of performance, for each trial the data includes a "win/loss" attribute (1:win, 0:loss), which describes whether the agent was successful ("beat" the game). Tables 2 and 3 show the controllers and games respectively.

As there are five repetitions for each game (49) and level (5) (between which the agents and games are reinitialised), we average the win/loss values to get $49 \times 5 = 245$ tasks. The different levels for task may be considered clusters, although some of the modifications to get different levels may force different approaches and lead to different required behaviours and responses. Since some tasks have the same (0) result for all agents (which would be problematic for the calculation of correlations) we add a very small random value to all results. Also, we scale the results for each game to have standardised values with zero mean

Table 3: The 49 games used in the GVGAI experiments.

aliens	bait	boloadventures	boulderchase	boulderdash
brainman	butterflies	camelRace	catapults	chase
chipschallenge	crossfire	defem	digdug	eggomania
escape	factorymanager	firecaster	firestorms	frogs
iceandfire	infection	jaws	labyrinth	lemmings
missilecommand	modality	overload	pacman	painter
plants	plaqueattack	portals	racebet2	realportals
realsokoban	roguelike	seaquest	sokoban	solarfox
superman	surround	survivezombies	tercio	thecitadel
waitforbreakfast	whackamole	zelda	zenpuzzle	

and unit standard deviation. As we will work with Pearson (linear) correlations, this scaling does not affect the correlations, but allows a more commensurate aggregation to determine the abilities of each agent and the comparison with the overall variance as well.

Figure 2 (left) gives a first impression of the results. We see that responses show quite an irregular distribution. A first view at how agents behave can be seen on the plot on the right. It shows the average responses per agent ($\bar{R}_i$, what we denote by "Agent performance" on the x-axis) against the variance of responses per agent (σ_i^2, denoted by "Agent variance"). What we see is that the most able agents seem to have higher variance. But does this mean that they are less general? Were not we expecting low variance for the worst agents, scoring poorly on all tasks, and also for the best agents, scoring well on all tasks? We see this phenomenon on the left, but not on the right. Nevertheless, it is important to look at both the left and right figures to understand what this really means. If we have two agents with roughly the same performance (e.g., 2 and 7, on the right figure), we can compare their generality by looking at their variances. However, it is misleading to compare the variances of two agents with very different performance, because *the magnitude of the responses will have a very important effect, especially for the agents with high performance.* This has much to do with the asymmetric distribution of responses on the left. These are some of the reasons that motivate us (and Spearman) to look at the correlations instead.

3.2 Analysis of Subpopulations Binned by Abilities

Using these 245 tasks as columns and the 23 controllers as rows, we calculated the 245 × 245 correlation matrix $\rho_{i,j}$. Removing the reflexive and symmetric correlations, we have $(\frac{245\times244}{2} = 29890)$ correlations, from which 17464 are positive. The distribution of these correlations can be seen in Figure 3 (left). The average correlation of these 29840 values is 0.106. This is just slightly positive, which means that a hypothetical general factor would have a limited effect on the population of techniques. Basically, we see the "non-transitivity" effect, i.e., a high degree of specialisation in these algorithms. Figure 3 (right) shows the agent performance ($\bar{R}_i$), its average score for all tasks, as a proxy for ability. Using this ability, we now sort the agents and split the agent population according to it. Instead of simply separating the agents in two groups (most able and least able), we get a

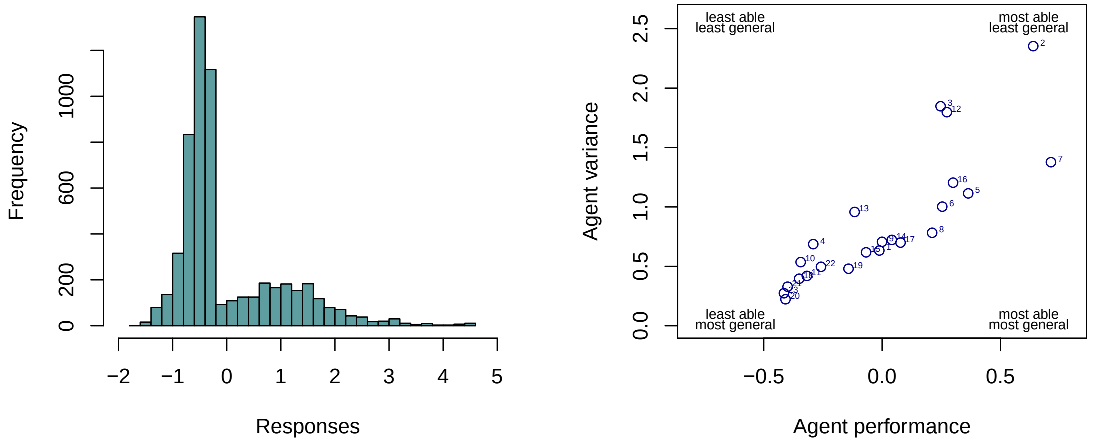

Figure 2: GVGAI results for 23 agents (controllers) and 245 tasks (49 games with 5 levels each). Left: histogram of the 23 × 245 = 5635 responses. Right: agent performance versus agent response variance, showing the regions of low/high ability and low/high generality. We see that agents 2 and 7 have similar performance, but agent 7 has smaller variance and seems therefore more general.

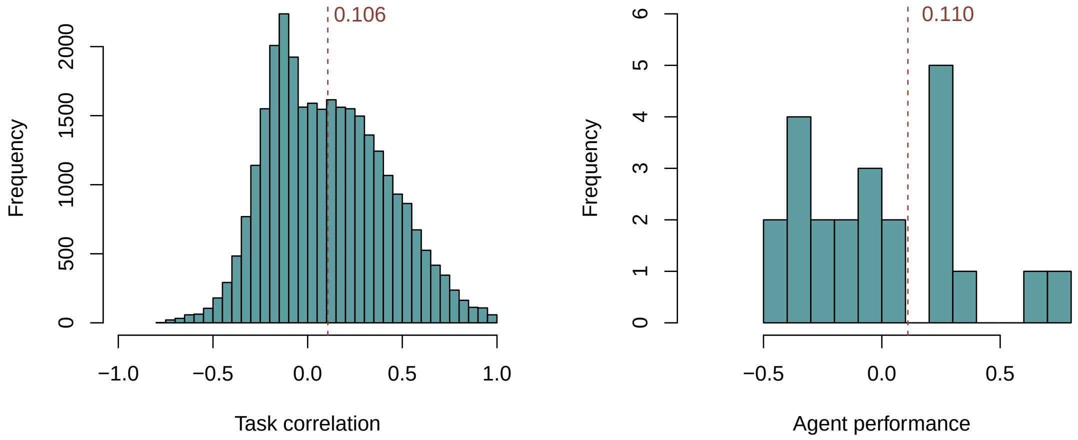

Figure 3: GVGAI results for 23 agents (controllers) and 245 tasks (49 games with 5 levels each). Left: histogram of the 29890 correlations. Right: histogram of agent performances.

more detailed analysis if we analyse different quantiles (from worst to best). This is shown in Figure 4 (left).

The top black cross uses all tasks (245) and all agents (23) together, with 0.106 correlation (as mentioned above). The second decomposition (orange triangles) shows 4 bins,

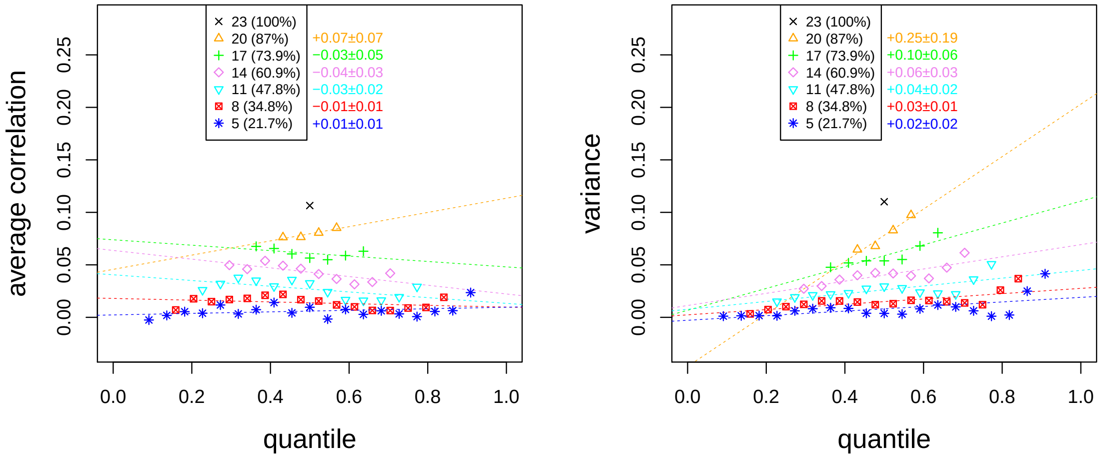

Figure 4: GVGAI results for 23 agents (controllers) and 245 tasks (49 games with 5 levels each). Note that the x-axis goes from lower to higher abilities. Linear regressions in six different colours are shown for the six series having more than one bin. The slopes of these regression models and the 95% confidence intervals for the slopes are also shown in six different colours at the top of the plots. Left: average correlation per quantiles using several bin sizes where agents are sorted by overall performance. Right: the variance of agent performance of the bins.

using agents 1..20, 2..21, 3..22 and 4..23 respectively, sorted by increasing ability (most able on the right). The plot shows that as the bins become smaller, the average correlation also becomes smaller. This is natural as a result of having smaller samples. If we look at all the decompositions, from triangles to asterisks, none of them is clearly decreasing or increasing, according to the slopes and the 95% confidence intervals of the linear fits. This means that the correlation is usually the same for all bins. According to this, we do not observe the diminishing returns here (the decompositions should be decreasing as the x-axis goes from lower abilities to higher abilities). In any case, recalling Jensen's criticism in Section 2 that the variance could explain part of it, Figure 4 (right) shows the variance of agent performance for each bin. We see that it explains most of what happens on the left plot. Actually, the overall variance is 0.11 (note that the magnitudes are commensurate as we normalised the scores per column).

3.3 Analysis of Subpopulations Binned by Abilities and Difficulties

We can further investigate the behaviour of these correlations by ability bins if we consider that we might get higher diversity of abilities (performance variance) for groups of tasks with a difficulty that is more centred around the agent ability, ensuring more diversity than a collection of very easy tasks or a collection of very hard tasks. In other words, it seems reasonable to expect that many of the most able agents will succeed for the easy tasks and many of the least able agents will fail for the difficult tasks. Interestingly, the data from the GVGAI competition incorporates the "level" of each task. These levels "differ from each other through variations on the locations of sprites, resources available and variations on non-player character (NPC) behavior" (Bontrager et al., 2016). Using this level as difficulty, Figure 5 (left) shows the same analysis of the correlation by different bin decompositions.

As mentioned above, the top black cross uses all tasks (245) and all agents (23) together, again with 0.106 correlation, and the second decomposition (orange triangles) again shows 4 bins, using agents 1..20, 2..21, 3..22 and 4..23, sorted by increasing ability (most able on the right). But now, for each of the bins, we also slice the games (tasks) according to their difficulty. For instance, for the leftmost bin of the orange triangles, the least able agents 1..20, we calculate the correlation with only the least difficult tasks 1..213.

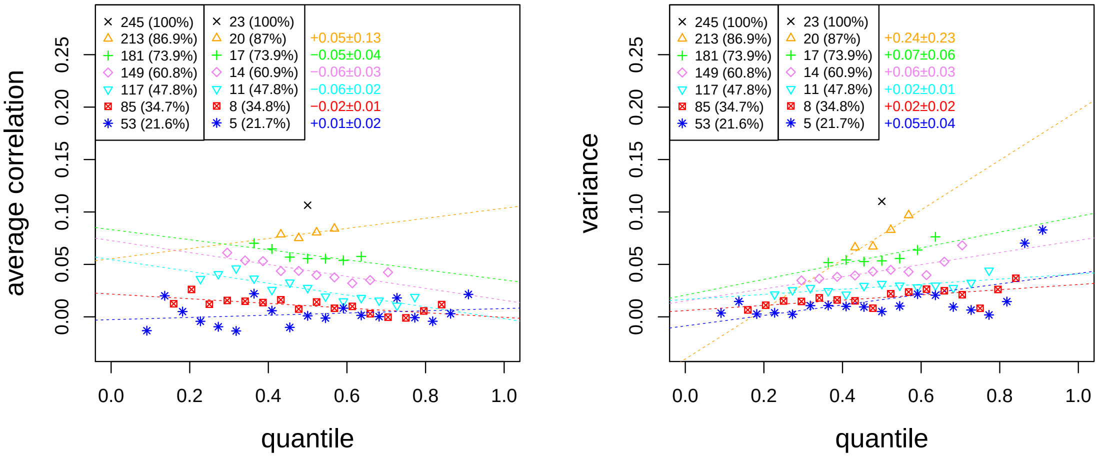

Figure 5: Left: Average correlations for 23 agents (controllers) and 245 tasks (49 games with 5 levels each). Note that the x-axis goes from lower to higher abilities. Linear regressions in six different colours are shown for the six series having more than one bin. The slopes of these regression models and the 95% confidence interval for the slopes are also shown in six different colours at the top of the plots. Left: average correlation per quantiles using several bin sizes, where results are sorted by agent performance, and each bin is only evaluated with the tasks of that quantile of difficulty ("level"). Right: Variance of agent performances.

Basically, we see the same picture as Figure 4 (left), just a little bit more bumpy (because the bins have fewer instances and are hence more subject to random effects). With this correction, we do not see that the most able agents have a higher correlation, so there are no diminishing returns but neither do we find any augmenting returns here (the subpopulation of more able agents is not more general either). The slopes and 95% confidence intervals of the linear fits are now even less conclusive. Again, Figure 5 (right) shows the variances of agent performance for each bin, which do not change much with respect to Figure 4 (right) either.

Before making the interpretation that using difficulty in the bins changes (almost) nothing, we have to really see whether the "level" is actually a good metric of difficulty. Unfortunately, if we calculate the correlation between the level value (0..4) and average task performance (before normalisation) we get a surprising -0.03 correlation. Basically, the level is not a good metric of difficulty and the slicing by difficulty has no significant effect, exactly what we see on Figure 5 with respect to Figure 4.

Consequently, to complete the analysis, we are going to derive an empirical metric of difficulty and repeat the plot. The empirical difficulty of each of the 245 tasks is derived as 1 minus the average performance of all agents for that task (before normalisation). This gives a value between 0 (easiest) and 1 (most difficult) for each task (note that each pair of game and level is still considered a task since "level" is actually a variation of the game). Using this new empirical metric of difficulty, we get the decompositions of Figure 6 (left). Now we see a much more bumpy picture, especially for small bin decompositions (red squares and blue stars), but the shapes look slightly increasing now. The slopes and 95% confidence intervals of the linear fits are only significant for the 4-bin decomposition (yellow triangles) and the 12-bin decomposition (blue triangles). Were this trend confirmed, this would be the reverse to Spearman's Law of Diminishing Returns and would indicate that the bins with more able agents would have higher correlation. In fact, the bin with the best 5 agents applied to the most difficult 53 tasks has a correlation of about 0.13 (shown as a blue star on the right of Figure 6, left), one of the highest found for any bin so far.

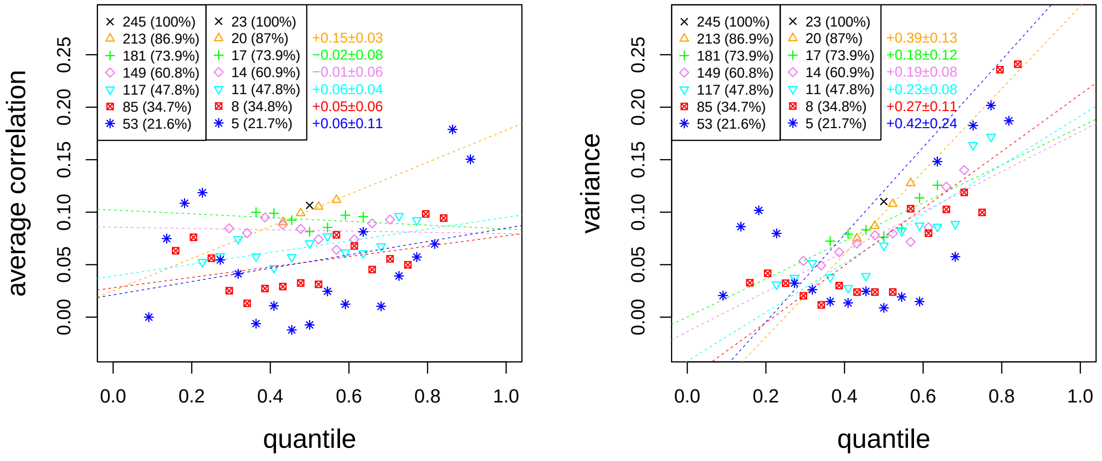

Figure 6: Same as Figure 5, but difficulty is now derived empirically from performance instead of using the "level".

Finally, once again, Figure 6 (right) shows the variances of agent performance for each bin. In this case, the association between left and right plots is even more complicated, but the increasing trend for the variance is significant for all decompositions.

The previous analysis is performed with a small population of tasks (245) and a very small population of agents (23). Also, we do not have any control at all about the population of agents and its distributional properties, as they just come from the participants of one competition. The original "levels" did not represent an appropriate difficulty function, so we had to infer an empirical difficulty metric from the data, which can include other confounding factors. Accordingly, we cannot extrapolate for other situations, but nevertheless the analysis is useful to see whether there is a general factor emerging and whether it is stronger or not for most or least able agents. In this case, we do not see a significant level of generality in the results, so the best agents are not really general-purpose game players. Also, we do not see the appearance of Spearman's Law of Diminishing Returns for these artificial agents. Overall, despite the small sample, this correlation analysis is still insighful to understand whether a competition that aims at "general" game playing is actually succeeding in this direction and whether the best agents are both good and general.

4. Analysing Generality in a Controlled Scenario of Tasks and Agents

In order to have a more controlled analysis with fewer extraneous factors, we are now going to adapt the simple setting introduced by Hernández-Orallo (2015a) to have a population of agents and tasks. The most important difference from the scenario in the previous section is that we are going to exhaust all tasks, and sample the space of agents in a very even way, without human intervention (the agents are not coded by humans as in the previous section). This is an appropriate scenario for practical reasons. First, the use of minimalistic environments where the number of observations and actions is extremely reduced avoids many confounding factors, while still having some relatively rich phenomena. Second, we are interested in policy languages where agents can be generated automatically, but still be meaningful. Third, we want something simple that allows us to evaluate a large amount of tasks and agents quickly.

4.1 Agent-populated Elementary Cellular Automata: Definition and Examples

The environments we will work with are composed of an elementary cellular automaton (ECA) (Wolfram, 2002) with the addition of an agent that will be able to see and modify part of the usual behaviour of the automaton. The following definition specifies the structure of this kind of environment:

Definition 1 Framework: *We define a single-agent elementary cellular automaton (SAECA) as a tuple $\langle \mathcal{S}, \boldsymbol{\sigma}^0, p^0, \nu, \pi \rangle$. The state space $\mathcal{S}$ is represented by a one-dimensional array of bits or cells $\boldsymbol{\sigma} = \langle \sigma_1, \sigma_2, \ldots, \sigma_k \rangle$, also known as* configuration. *We consider the array to be finite ($|\boldsymbol{\sigma}| = k$) but circular in terms of neighbourhood ($\sigma_0 = \sigma_k$ and $\sigma_{k+1} = \sigma_1$). There is an initial array $\boldsymbol{\sigma}^0$, also known as* seed, *and an initial position of the agent, p^0. The behaviour of the environment transitions ν and agent π will be explained in subsequent definitions, but both interact over the configuration, which changes step by step. The order of events for each step in the system is: observations are given to the agent, agent actions are performed, the environment transition function is applied and finally, rewards to the agent are produced.*

Definition 2 Transition function: *The environment transition function is determined by a number ν, as any of the $2^{2^3} = 256$ rules that can be defined looking at each cell and its two neighbours. Specifically, a triplet transition is denoted by $XYZ \rightarrow B$. As there are 8 possible triplets to be defined, there are 2^8 possibilities. The numbering scheme convention introduced by Wolfram (2002) sorts the possibilities in the following order:*

$$\begin{array}{cccccccc} 111 & 110 & 101 & 100 & 011 & 010 & 001 & 000 \\ B_1 & B_2 & B_3 & B_4 & B_5 & B_6 & B_7 & B_8 \end{array}$$

so that $B_1B_2B_3B_4B_5B_6B_7B_8$ is mapped to an 8-bit number ν, the ECA rule. For every (circular) substring XYZ in $\boldsymbol{\sigma}$, we apply the matching rule $XYZ \rightarrow B$ and the bit Y is replaced by B.

For instance, the following transitions for each triplet define an ECA rule:

111	110	101	100	011	010	001	000
0	1	1	0	1	1	0	1

The digits of the second row represent the new state for the middle cell after each transition, depending on the triplet. In the above case, 01101101, in binary, corresponds to decimal number 109, the ECA rule number with Wolfram's convention. Given this rule, the configuration array 01100 would evolve in the following way, looping at the end:

$$\begin{array}{ll} 01100 & \\ 01101 & \\ 01111 & \\ 11001 & \\ 01001 & \Lsh \\ 11001 & \end{array}$$

Let us see a few examples of how these environments work. Figure 7 shows the evolution of several environments, with several values of ν, all with seed "010101010101010101010". We do not include any agent in the environments for the trials in this first figure. As a result, the space-time diagrams after 200 iterations are the same as a classical elementary cellular automaton with each number ν (Wolfram, 2002).

Let us now explore what happens when we include agents in these environments.

Definition 3 Agent interface: *Given the behaviour of the space, we consider just one agent π. The agent is located at one cell (its position p) with $1 \leqslant p \leqslant k$. Its initial position is denoted by p^0. The set of observations $\mathcal{O}$ is given by two bits $\langle \sigma_{p-1}, \sigma_{p+1} \rangle$ representing the contents of the left and right neighbouring cells respectively, i.e., σ_{p-1} and σ_{p+1}. The actions $\mathcal{A}$ are given by a 'move' and an 'upshot', denoted by the pair $\langle V, U \rangle$. For each timestep, the move is performed before the upshot. The ordered set of moves is given by {`left`=0, `stay`=1, `right`=2}, and the ordered set of upshots is {`keep`=0, `swap`=1, `set0`=2, `set1`=3}, which respectively mean that the content of the cell where the agent is does not change, the content of the cell is complemented ($0 \rightarrow 1$, $1 \rightarrow 0$), the content is set to 0 and the content is set to 1 respectively. The rewards are calculated according to the number of 1s that are in the neighbourhood of the agent, weighted by their proximity. More precisely, if the agent is at position p at time t, then the reward is given by:*

$$r^t \leftarrow \sum_{j=1..\lfloor k/2 \rfloor} \frac{\sigma^t_{p+j} + \sigma^t_{p-j}}{2^{j+1}}$$

It is easy to see that $0 \leqslant r^t \leqslant 1$.

Basically, the goal of the agent is to be surrounded by the highest number of 1s possible, by creating them or by exploiting the changes performed by the ECA rule. Of course, some ECA rules can undo the efforts of the agent to turn cells into 1s and others can be more

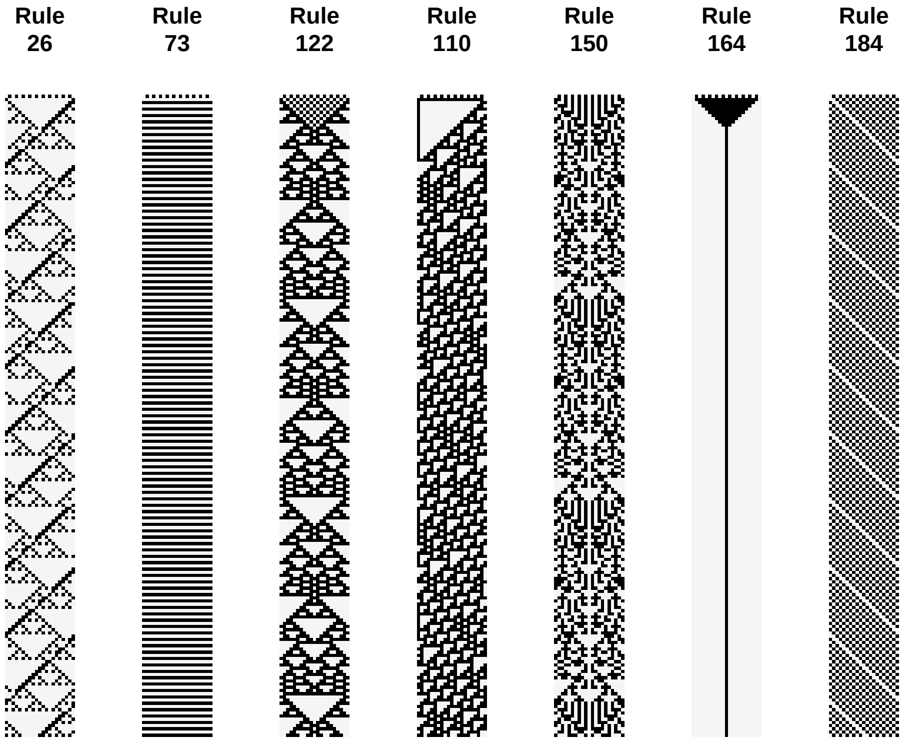

Figure 7: Space-time diagram (evolution for $t = 200$ steps) of several elementary cellular automata without agent. The initial array (seed) is always 010101010101010101010, whose length is 21 bits.

benevolent for the agent. This includes the extreme cases, rules 0 and rule 255, which set everything to 0s and 1s respectively.

Now we define the syntax and semantics of the agent. There are a few agent languages in the literature (Boutilier, Reiter, Soutchanski, & Thrun, 2000; Leonetti & Iocchi, 2010; Andre & Russell, 2002), but they are too oriented towards the architecture, are too focused on Markov Decision Processes or are not sufficiently minimalistic for bounding their size and having some interesting programs. Instead, we present a very minimalist language, also taking into account the minimalist environment.

Definition 4 APL **syntax**: *The agent policy language* APL *is given by an ordered set of instructions* $\mathcal{I} = \{$ `back`$=0$, `fwd`$=1$, `Vaddm`$=2$, `Vadd1`$=3$, `Uaddm`$=4$, `Uadd1`$=5$ $\}$. *The numbers on the right will be used as shorthand for the instruction. A program or policy* π *is a finite sequence of instructions* $\iota_1, \iota_2, ...$ *in* $\mathcal{I}$.

For instance, the string 20242335 represents a program in APL.

Definition 5 APL **instruction semantics:** *Instructions operate on a memory (or history) binary array* $\mathbf{m}$ *(not circular), and two accumulators* V *and* U*. The execution of instruction* ι *on the memory* $\mathbf{m}$ *with memory pointer at position* b *and accumulators* V *and* U *is given by:*

```
ExecInst($\iota$, $\mathbf{m}$, $b$, $V$, $U$)
  case $\iota$:
    back  : $b \leftarrow \max(b-1, 1)$
    fwd   : $b \leftarrow \min(b+1, |\mathbf{m}|)$
    Vaddm : $V \leftarrow (V + m_b) \, mod \, 3$
    Vadd1 : $V \leftarrow (V + 1) \, mod \, 3$
    Uaddm : $U \leftarrow (U + m_b) \, mod \, 4$
    Uadd1 : $U \leftarrow (U + 1) \, mod \, 4$
  end case
```

The execution takes the instruction ι and $\mathbf{m}$ as inputs, while b, U and V are both inputs and outputs.

Definition 6 APL **program semantics:** *Given a program π and a memory $\mathbf{m}$, the action that the agent performs is given by the result of the accumulators at the end of a process, defined as follows:*

```
Read the observation $\langle \sigma_{p-1}, \sigma_{p+1} \rangle$ and append these two elements at the end of the
   history array $\mathbf{m}$.
Place the memory pointer $b$ at the end of $\mathbf{m}$ (i.e., $b = |\mathbf{m}|$).
$V \leftarrow$ stay
$U \leftarrow$ keep
forall $\iota \in \pi$
  ExecInst($\iota$, $\mathbf{m}$, $b$, $V$, $U$)
endfor
return $\langle V, U \rangle$
```

The memory array $\mathbf{m}$ is empty when a trial is started, but its content is preserved between steps.

Let us see an example. Consider the policy 20242335. If the agent is located at the fifth position of the configuration 000101111 and has a current history $\mathbf{m} = 111010$ then the observations 1 and 1 will be appended to $\mathbf{m}$, leading to $\mathbf{m} = 11101011$. We start the process with $b = 8$, $V = 0 =$ `stay` and $U = 0 =$ `keep`, and we have the following execution:

1. $\iota_1 = 2 =$ `Vaddm`, $V \leftarrow (V + \mathbf{m}_8) \bmod 3 = 1 =$ `stay`.
2. $\iota_2 = 0 =$ `back`, $b \leftarrow max(8-1, 1) = 7$.
3. $\iota_3 = 2 =$ `Vaddm`, $V \leftarrow (V + \mathbf{m}_7) \bmod 3 = 2 =$ `right`.
4. $\iota_4 = 4 =$ `Uaddm`, $U \leftarrow (U + \mathbf{m}_7) \bmod 4 = 1 =$ `swap`.
5. $\iota_5 = 2 =$ `Vaddm`, $V \leftarrow (V + \mathbf{m}_7) \bmod 3 = 0 =$ `left`.
6. $\iota_6 = 3 =$ `Vadd1`, $V \leftarrow (V + 1) \bmod 3 = 1 =$ `stay`.
7. $\iota_7 = 3 =$ `Vadd1`, $V \leftarrow (V + 1) \bmod 3 = 2 =$ `right`.
8. $\iota_8 = 5 =$ `Uadd1`, $U \leftarrow (U + 1) \bmod 4 = 2 =$ `set0`.

After this program, which is run internally, we obtain the action that the agent will perform on the environment, which is given by $\langle V, U \rangle = \langle 2, 2 \rangle = \langle \texttt{right}, \texttt{set0} \rangle$. This means that the agent will move right and set the content of the cell to 0.

While the class of policies generated by this language is infinite, the language is still not universal, and all (finite) programs end. The goal of this language is to be able to express some simple policies that may be useful in the environment. The policies that can be expressed in this language can represent reactive behaviours: moving in different directions and/or changing cell contents, according to the perceived adjacent cells in the current or previous steps (because of its perception memory). The policies do not have working memory (although the environment can be considered as an external memory) and do not include loops or recursion. As a result, the language just does simple processing of the inputs. Longer programs usually allow for the combination of more cells in the agent history.

Figure 8 shows how the environment with elementary cellular automaton number 110 varies for several agent policies. The resulting space-time diagram patterns are different. Similar things (where differences are more visible with respect to the corresponding diagram in Figure 7) happen with rule number 164 (Figure 9).

For this scenario, the result $R_{\pi,\mu}$ is defined as the (expected) response of agent π in task μ, which is calculated as an average of the rewards r^t for the 200 steps t. For instance, in Figure 9 policy 23555 for rule 164 seems to have higher R than policy 24 for the same rule.

4.2 Analysis of Subpopulations Binned by Abilities

Given the setting seen above, we are now going to analyse whether some agents display some generality and whether this increases or decreases for more able subpopulations. Using the agent policy language APL we generated 400 agents with their instructions chosen uniformly from the instruction set and a program length also uniformly distributed between 1 and 20. We evaluated each agent with all the 256 possible ECA rules, with 21 cells, random initialisation (seed), using 100 of iterations per trial. We scaled the 256 × 400 results, task per task, so that for each task (ECA rule) we had mean 0 and standard deviation 1. Similarly to the GVGAI competition case, this does not affect the correlation.

As we did for GVGAI, Figure 10 (left) gives a first impression of the results. We see that responses show a very regular (Gaussian) distribution. The plot on the right shows the average responses per agent ($\bar{R}_i$) against the variance of responses per agent (σ_i^2). We do not see a clear pattern, but the agent variance is almost always above 0.5, especially for the most able and least able agents. It seems there are less generality everywhere than for the previous scenario, but, again, this variance depends on the magnitude of results. It is better to look at the correlations.

In order to this analysis, we calculated the 256 × 256 correlation matrix for the 256 rules. From all the correlations ($\frac{256\times255}{2} = 32640$), 29612 were positive. The average correlation is 0.125. The distribution of these correlations can be seen in Figure 11 (left). It is interesting to see that we get a positive, albeit small, average correlation, even if we are using randomly-generated agents for all possible tasks (all ECA rules). This is given by the reward mechanism, which is the same for all tasks (having 1s in the surrounding cells) and there are some agents that go well for this reward criterion disregarding the task.

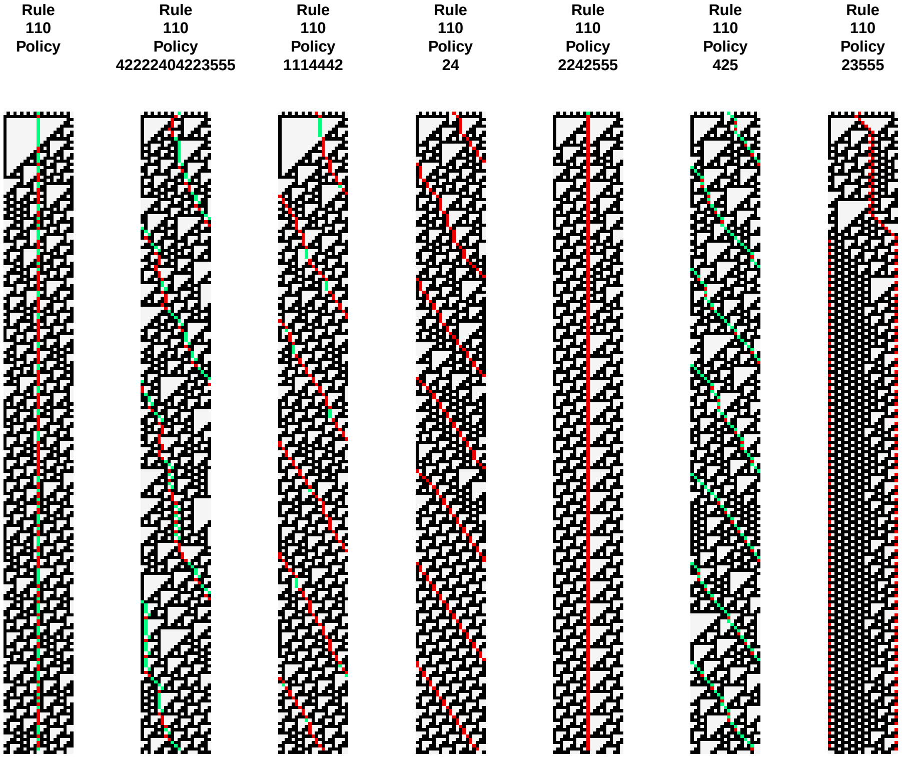

Figure 8: Space-time diagram (evolution for $t = 200$ steps) with different agent policies for the elementary cellular automaton with rule 110. The initial array (seed) is always 010101010101010101010, whose length is 21 bits. The agent is represented by a red dot when the cell has a 0 (like the white ones) and by a green dot when the cell has a 1 (like the black ones). The leftmost diagram is the empty policy ($\langle \texttt{stay}, \texttt{keep} \rangle$).

Then we averaged the results for each agent to get their average performance (ability). Figure 11 (right) shows this agent performance ($\bar{R}_i$). It seems there is a peak around 0, which, for this scenario, can be full of agents that do not manage to have any effect on the environment and hence score 0 on average. Using these agent performances as ability, we sorted the agents and split the agent population according to different quantiles, from worst to best[1]. Different sizes of the bins (subpopulations) were used for the quantiles. The bin decompositions are shown in Figure 12 (left).

The black cross on the top represents one bin with the whole population (400 agents), with an average correlation of 0.125, as said above. The second shape (using orange triangles) is formed by the 51 possible bins using agents 1..350, 2..351, . . . , 51..400. For smaller bins underneath we see that the average correlation decreases (in other colours). If we look at the concave shapes we clearly see that the average correlation is not the same for the

[1] In some previous work (Martínez-Plumed, Prudêncio, Martínez-Usó, & Hernández-Orallo, 2016; Hernández-Orallo, 2017), they were ordered from best to worst.

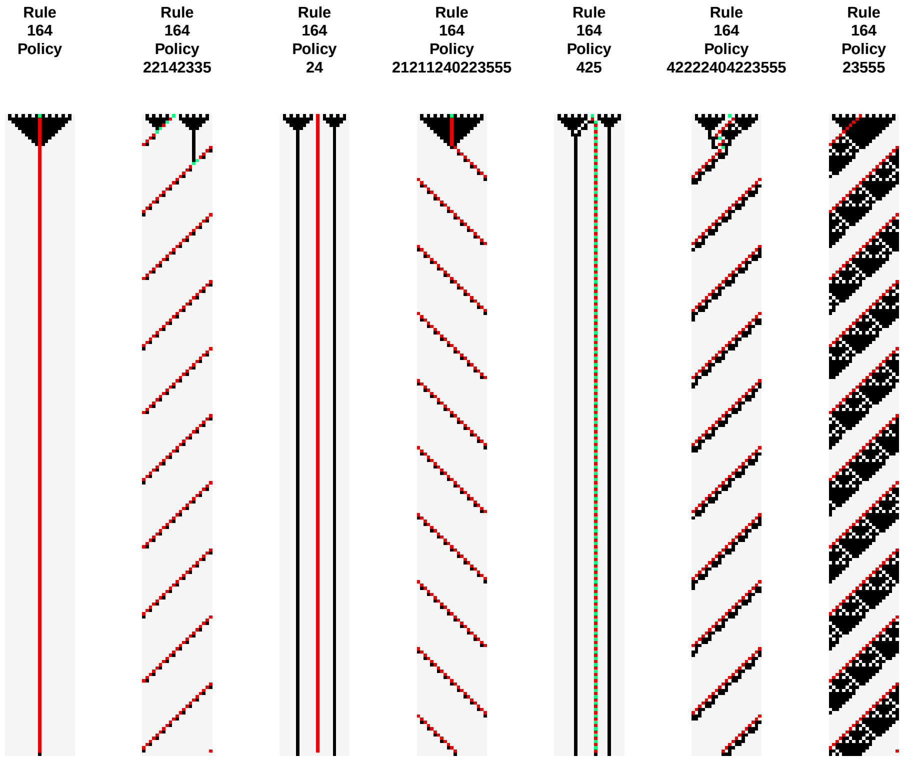

Figure 9: Same as Figure 8, with rule 164 and other policies.

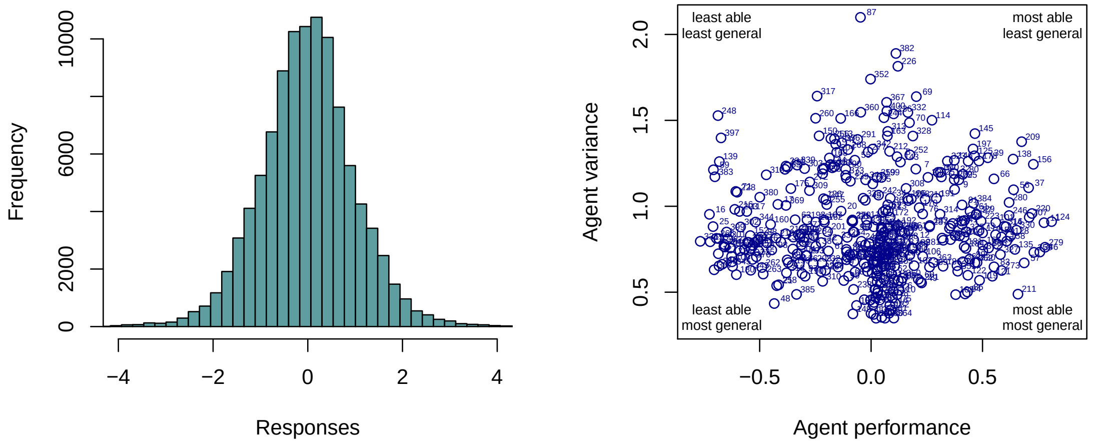

Figure 10: Results for 400 agents (randomly-generated programs) and 256 tasks (all the ECA rules). Left: histogram of the 400 × 256 = 102400 responses. Right: agent performance versus agent response variance, showing the regions of low/high ability and low/high generality.

whole range, with smaller values for middle quantiles (around 0.5 on the x-axis). In fact,

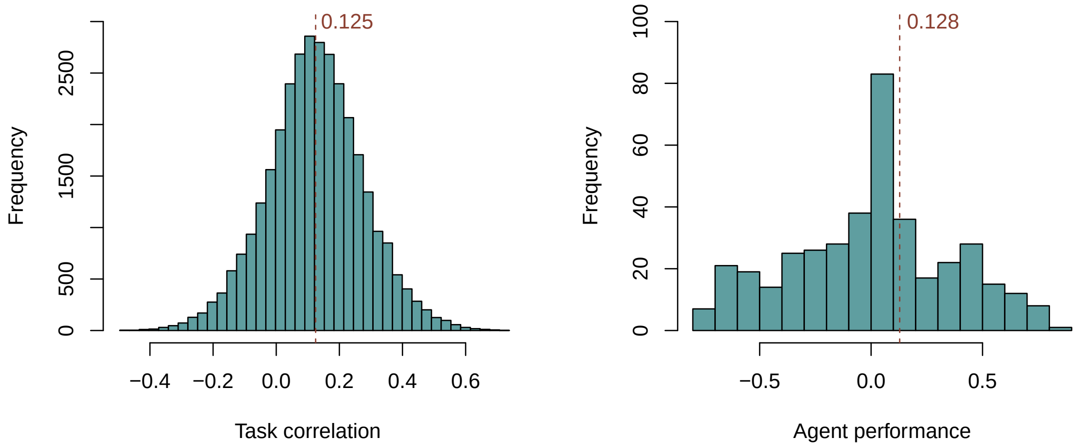

Figure 11: Results for 400 agents (randomly-generated programs) and 256 tasks (all the ECA rules). Left: histogram of the 32640 correlations. Right: histogram of agent performances.

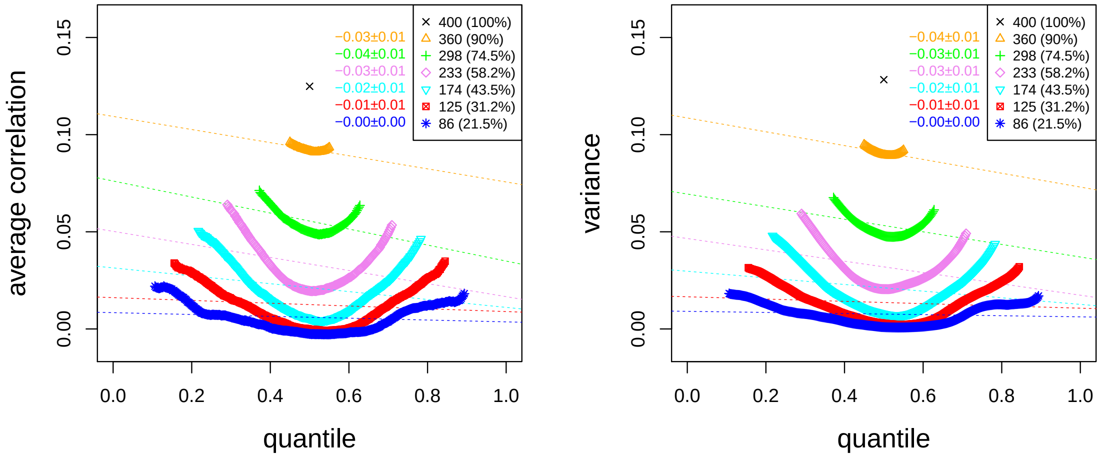

Figure 12: Results for 400 agents (randomly-generated programs) and 256 tasks (all the ECA rules). Note that the x-axis goes from lower to higher abilities. Linear regressions in six different colours are shown for the six series having more than one bin. The slopes of these regression models and the 95% confidence interval for the slopes are also shown in six different colours at the top of the plots. Left: average correlation per quantiles using several bin sizes where agents are sorted by overall performance. Right: the variance of the agent performance per bin.

we see correlations are higher for the more able group (high performance) and the less able group (low performance).

Trying to interpret these first results, we first see that the trends are very timid. Looking at the regression lines, we see some decrease, which is significant for some of the decompositions. Before seeing this as support for SLDR, we can look at the variances in Figure 12 (right), which are practically equal to the average correlations, as pointed by Jensen. This might be related to the way the environments are defined, and suggest that we should explore some variations.

In particular, the reward mechanism makes that those agents that commonly generate 1s will be usually good for many tasks, and those agents that commonly generate 0s will be usually bad for many tasks. This thing alone could be behind the 0.125 task correlation. In order to analyse the relevance of the reward criterion on the results, we perform a second experiment where the reward mechanism is being mirrored for half of the tasks (so agents cannot specialise to it). By mirroring we mean that the goal is now to be surrounded by as many 0s as possible. This is introduced in order to have those policies that just create 1s to be compensated by other dual environments, for which this strategy would not work. This creates a balance and symmetry in the way programs deal with cells. However, with the original agent policy language, this would also lead to every possible agent scoring equally on average (a kind of no-free lunch result). In order to prevent this, we have to include a second modification to the agent policy language: agents can now see the rewards. This is achieved by modifying definition 6, in such a way that when the observation is read $\langle \sigma_{p-1}, \sigma_{p+1} \rangle$ and these two elements are appended at the end of the history array $\mathbf{m}$, an extra bit is appended as follows: if the previous reward is greater than 0.25 then a 1 is appended. Otherwise, a 0 is appended. Note that by making agents observe the rewards we allow them to react to changing reward settings, but it is the mirroring mechanism that benefits the agents that do so. These small modifications lead to very important changes in the results, as we can see in Figures 13 and 14. First we see that the distribution of responses is more multimodal now, and agent performance is much smaller now, as are the correlations.

We can now look into the bins in Figure 15, where we now use 256 agents instead of 400. The top black cross uses all tasks (256) and all agents (256) together. The second shape (orange triangles) shows 17 bins, using agents 1..240, 2..241, ..., 17..256, and so on for the other shapes, according to the sizes shown in the legend.

The average correlation almost disappears. It is now just 0.004, which is more consistent with a population of agents that are generated randomly. This also matches the variances on the right.

4.3 Analysis of Subpopulations Binned by Abilities and Difficulties

We now do an extra change in our analysis, as we did for the GVGAI competition data. Figure 16 also slices the tasks by *difficulty*. We evaluate the more able agents with more difficult tasks. In order to do this, we calculate difficulty of a task as the complexity of the simplest policy that is successful for the task (Hernández-Orallo, 2015b). Complexity/simplicity is measured as a combination of the size of the policy and its execution time. We simplify the estimation of difficulty by only considering the length of the policies (and not the execution time). This simplification is not very important for this agent language, since all policies have a finite execution time that directly depends on the number of in-

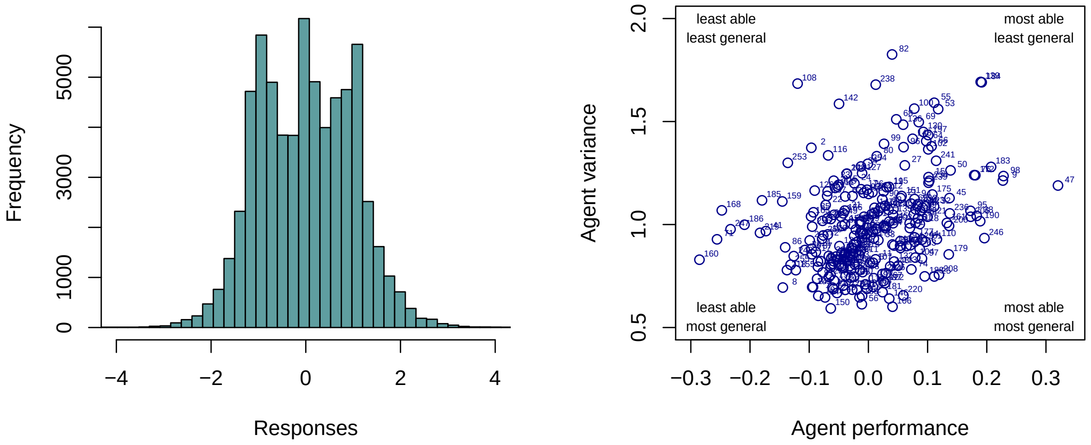

Figure 13: Same as Figure 10 but now the average correlations are for 256 agents (randomly-generated programs) and 256 tasks (all the ECA rules), using mirrored rewards for half of the trials.

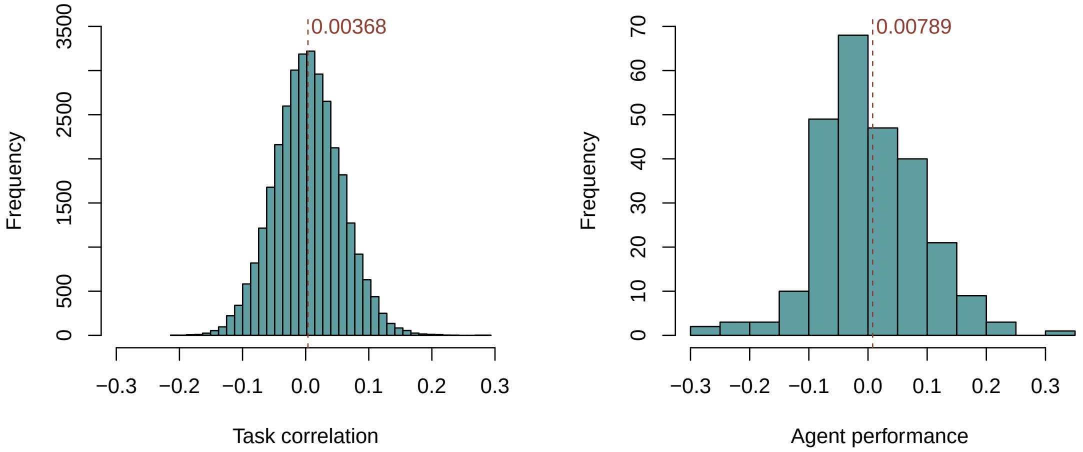

Figure 14: Same as Figure 11 but now the average correlations are for 256 agents (randomly-generated programs) and 256 tasks (all the ECA rules), using mirrored rewards for half of the trials.

structions (there are no loops or jump instructions). Consequently, given a tolerance ϵ, the difficulty $\hbar^{[\epsilon]}(\mu)$ of a task μ is the length of shortest policy π that is acceptable for the task:

$$\hbar^{[\epsilon]}(\mu) \triangleq \min_{\pi \in \mathcal{A}^{[\epsilon]}(\mu)} L(\pi) \tag{1}$$

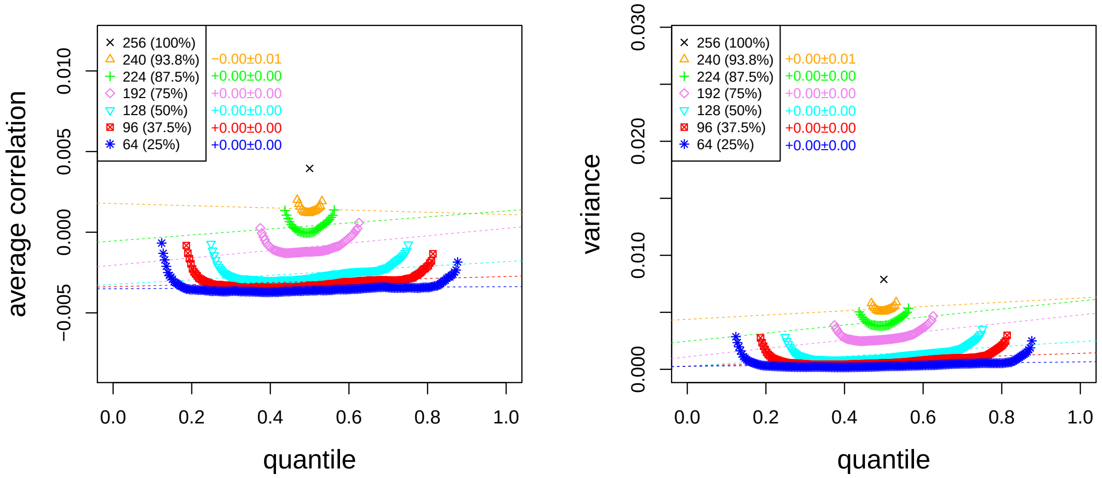

Figure 15: Same as Figure 12 but now the average correlations are for 256 agents (randomly-generated programs) and 256 tasks (all the ECA rules), using mirrored rewards for half of the trials. Again, the x-axis goes from lower to higher abilities. Left: average correlation per quantiles using several bin sizes, where results are sorted by agent performance. Right: the variance of the agent performance of the bins.

Note that this is not the Kolmogorov complexity of the task (i.e., the shortest description for the task) but rather the shortest description of any (acceptable) *solution* for the task. Acceptability is defined using a tolerance ϵ:

$$\mathcal{A}^{[\epsilon]}(\mu) \triangleq \{\pi \ : \ R_{\pi,\mu} \geqslant 1 - \epsilon\} \tag{2}$$

i.e., the set of all acceptable policies for a task μ is given by those policies whose expected response is above a threshold, given by the tolerance ϵ. Recall that we defined expected response R as the average reward result of agent π in task μ. Instead of exploring all possible policies, we implemented this approximation to difficulty exploring 400 random policies for each task and finding the simplest ones in this set. This represents an upper bound to difficulty. We chose tolerance to be the response that separates the 10% best agents for each task.

Using this metric of difficulty, we sliced the set of tasks into bins (ties were broken randomly). As we only use 256 agents in this experiment, the sizes of the bins were also the same. As a result of all the changes, Figure 16 shows different shapes. As mentioned above, the top black cross uses all tasks (256) and all agents (256) together, with 0.004 correlation, and the second shape (orange triangles) shows 17 bins, using agents 1..240, 2..241, ..., 17..256, and so on. But now, for each of the bins, we also slice the problems (tasks) according to their difficulty. For instance, for the first bin of the orange triangles (the least able agents 1..204), we calculate the correlation with only the least difficult tasks 1..204.

The slicing by ability and corresponding difficulty for each group now shows a very different picture. Figure 16 shows some very slight slopes in the descompositions, where we find higher correlations for higher abilities (the regressions have a very small, but significant slope). In this case, the variance on the right does not explain the plot on the left, and variance for small bins increase, somewhat surprisingly. Overall, this plot gives no evidence in favour of Spearman's Law of Diminishing Returns.

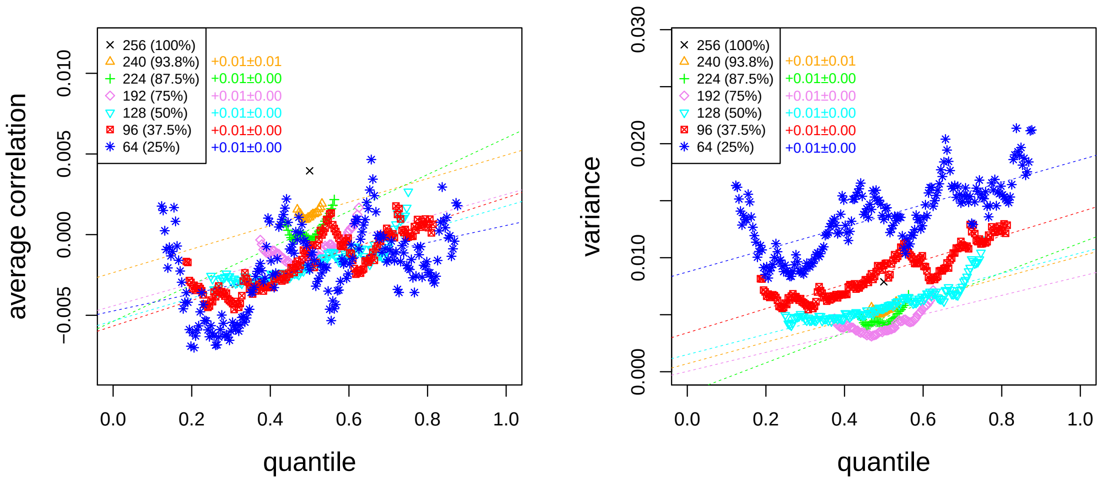

Figure 16: Same as Figure 15, but each bin is only evaluated with the tasks of that quantile of difficulty.

We have to keep in mind that we are generating agents with a very simple policy language. Still, the agents can now access the rewards and compute actions with them so that meaningful policies are generated. For instance, the policy that repeats the same action if the reward is good and does another action otherwise can be coded with a relatively short program in this language. Nevertheless, despite the flexibility of the language, we cannot expect any especially good agent since programs are very small. But still, as they are randomly generated, most agents are completely clueless in the environments. However, after the GVGAI competition data, this was precisely the basic scenario we wanted to explore, resembling some kind of minimal artificial life situation where we can consider all agents up to a certain complexity and see if any correlation appears, however small it is.

5. Discussion

Looking at the scenarios of the two previous sections together (the GVGAI competition and the synthetic setting) we find a consistent picture, and we can answer the questions that were formulated in the introduction. The positive manifold is very small for the GVGAI case and almost non-existent for the cellular automata scenario. In order to confirm this, we also applied factor analysis to the data (using the R package "psych"), to both the GVGP case and the cellular automata (non-symmetric rewards case and symmetric rewards case). A one-factor model covered a proportion of the variance of 0.22, 0.16 and 0.01, respectively, which is also consistent with the absence of a clear general factor in any of the three cases.

Since the ability levels of the agents are small, as well as their generality (interpreted as their manifold), the most interesting thing to look at is their trends, after binning by both ability and difficulty. In these situations, the SLODR did not appear, although one can argue that at this level of ability there is nothing yet to saturate, one of the rationales behind the SLODR. For instance, if we focus on the most able agents in Figures 6 and 16, we see that an incipient positive manifold may start appearing. The trend is extremely small, and the slope barely significant in a few cases according to the confidence intervals.

If more experiments with more able agents brought positive slopes, we should consider the possibility of postulating the reverse hypothesis, formulated as follows: given a population of agents, the most able they are, the stronger the positive manifold will be, provided they are evaluated at a difficulty level that suits and discriminates their ability range. We can call this hypothesis the *universal law of augmenting returns* (ULOAR), opposing SLODR. In fact, at a conceptual level, the ULOAR makes more sense for AI; there is no reason to think that for artificial agents we may find some kind of saturation, once the tendency is initially found at very low degrees of general ability. Indeed, the postulation of this hypothesis comes more from the very notion of generality, especially if we assume no limits on the difficulty scale. In this case the scores can range freely as well without a possible saturation (in human intelligence, the limits to formulating extremely difficult questions come from the fact that humans themselves formulate the questions).

Of course, solely on our experimental data, we cannot extrapolate from just two cases that simply show a slightly higher (yet very small) correlation for the more able groups. Still, this contributes to the intuition that, for artificial agents, SLODR may not hold in general. Also, we cannot extrapolate this for human populations. Indeed it is still unclear whether SLODR holds for humans or, more precisely, whether it holds for some human populations with some distributions of tests.

The small statistical significance of the results is not the only limitation of this work. There are also some issues about the way tasks are taken by each agent. We consider a situation where agents are reinitialised after each task, and so the order of tasks is irrelevant. In human evaluation, this is clearly not realistic, but also possibly not desirable, as we can use adaptive testing to get a good evaluation by using tasks that are closer to the ability of the individual that is being tested. Especially for machine learning in areas such as gradual and incremental learning (Madden & Howley, 2004; Carroll & Seppi, 2005), we can have *transfer* and *graduation* effects, in which a particular order of tasks (from easy to more difficult, or from one domain to others) might be more beneficial for some agents than others, or make some agents more or less general. For instance, humans are said to be able to cope with any kind of task, but this could only be true when one considers the necessary or convenient acquisition of some other skills first. Even seemingly unrelated tasks can have effects on each other with an important effect on the development of some skills. Correlations might appear when presented in different orders, such as chess for mathematics or reading skills in humans (Sala & Gobet, 2016).

In a broader context, this paper has explored the notion of general intelligence as performance for a range of tasks, following the recent theoretical and experimental analysis of the problem of AI evaluation (Legg & Hutter, 2007; Hernández-Orallo & Dowe, 2010; Mnih et al., 2015; Hernández-Orallo, 2016). Despite the indications, the ultimate goal has not been to make any definitive claim about AI and generality, but to propose a novel methodology (brought from a different area) of how generality can be analysed in AI. Also, this can have implications in human and animal intelligence research, and areas such as artificial life and multi-agent systems. However, some areas in AI can actually benefit from some of the issues raised in this paper more immediately. The dilemma between SLODR, ULOAR or neither of them can suggest new ways of empirically analysing AI systems, of devising new benchmarks and competitions and, most especially, of analysing the experimental results. Let us give some insight of how this could be done:

- A first important issue for the analysis of real competitions and benchmarks would be the estimation of task difficulty, which is necessary to make the analysis properly, as we have seen here with the bins per difficulty. We advocate for principled approaches to difficulty, based on the policy descriptions (as done for the synthetic setting) instead of deriving it from average performance (as done for the GVGAI competition). This is closely related to the possibility of deriving tasks and their score magnitudes from first principles, such that ratio scales are obtained, so that we are sure how all contribute to the overall ability in a meaningful way. A particularly consistent combination is the use of a tolerance of difficulty, which makes tasks binary, so converting the aggregation of results of the same difficulty truly commensurate. Its variance would be equal to that of a Bernoulli distribution, which is a function of the mean. In this case, an increase in ability beyond random guess would necessarily entail an increase in generality. Still, some other populational approaches such as Item Response Theory (IRT) could be very appropriate (Embretson & Reise, 2000; De Ayala, 2009; Martínez-Plumed et al., 2016; Martínez-Plumed & Hernández-Orallo, 2017). For instance, in IRT models of ability, a better person-fit can represent that the agent is more general.

- A second, more controversial, issue is the possibility that generality can also be measured with fewer, yet more abstract, tasks, without the need of an empirical analysis. This is perhaps possible, but it requires a strong theoretical understanding of what a task really measures. It is important to highlight that this is one of the traditional issues in behavioural evaluation (Hernández-Orallo, 2017). For instance, IQ tests can be a proxy for general intelligence in humans, but it has been shown that, if applied to AI systems, their scores become meaningless (Dowe & Hernández-Orallo, 2012). Actually, AI systems can get good scores by *specialising* for them (Hernández-Orallo, Martínez-Plumed, Schmid, Siebers, & Dowe, 2016). This not only discourages the use of human intelligence tests as a sufficient condition for generality (or intelligence), but it also applies to the use (or overuse) of *challenging* tasks in AI. It is quite common that when AI researchers attempt a previously unattainable problem (e.g., self-driving cars or solving Go), they end up with very specialised systems (albeit the ideas can be re-applied to other problems with human intervention). This suggests that fully autonomous general-purpose AI is perhaps better evaluated and encouraged against a wide range of simple tasks (instead of a few complex tasks), and that only when the systems in these competitions and benchmarks start to show some *general* competence, the difficulty of the tasks can be increased. This contrasts with a tradition in AI of seeking human or superhuman level for a particular (perhaps important) task, while being calamitous for other tasks.

- A third issue is that, in the context of a diversity of tasks, it is always important to analyse the effect of agents based on a "big switch", which, depending on some detected features of the problem, can delegate the task to more specialised subprocedures. This is the approach taken by techniques such as meta-learning and algorithm portfolios (Gerevini, Saetti, & Vallati, 2014), at least if features are detected and not given, as done by Bontrager et al. (2016). The resulting systems are not specific any more. For instance, algorithm selection "improves the performance and versatility of those systems across a broad range of application domains" (Lindauer, Hoos, Hutter, & Schaub, 2015). An important question is whether the results in future competitions or benchmarks that use these "hyper-agents" (Mendes et al., 2016) will have a higher general factor if these systems are included. This question is not straightforward since there is no strict line between modularity and generality in the way these systems work. Being good at a range of tasks can entail some previous exploration to decide which specific policy to use for the task. Also, the big switch can just work at a higher level, with some subgroups of problems. Nevertheless, it is clear that an exhaustive big switch approach with particular solutions for every single specific task would be impractical for a really wide range of tasks (except for memory-oriented or retrieval tasks).

- A fourth issue takes place when we analyse a population of agents that compete against each other. This is a common situation in multiplayer games (such as most table games), multi-agent systems and artificial life environments. For all these cases, the individual analysis of generality cannot be done without the populational perspective. A general system here would be one that consistently scores against all the other systems. This seems easier for the best and worst systems (they can win or lose against all, respectively) than any middle-rank system. Should it get the same expected results independently of the ability of the opponent or should win/lose against bad/good opponents, respectively? We see the connection of generality and the ability of performing well against a diversity of other agents, usually associated with the term "social intelligence", as relevant here. An interesting phenomenon to study in a competitive scenario is whether correctly ranking the agents in terms of ability is easier the more general the agents are. In competitions where there is more regularity, and good agents usually beat bad agents, we can properly talk about the term "transitivity" (which we mentioned in the introduction), and generality is expected to be high. On the contrary, if agents are less reliable and do not win or lose consistently with agents of lower or higher rank respectively, the overall generality is expected to be lower. Some of these issues are sometimes implicit in methods that derive rankings from competitions, such as the Elo rating (1978) or more sophisticated schemes (Aziz et al., 2015).

All these issues set some challenges, but also opportunities, that may require some extra guidelines (Togelius, 2016) about how to organise and analyse the results of a competition or a benchmark, especially if the aim is to encourage the construction of general-purpose AI systems. It is also important to analyse how competitions evolve year after year by comparing the indicators of generality. In this case, the competitions must preserve a subset (sufficiently large but still small compared to the number of tasks in the competitions), kept for all the editions, such that results can be compared on this same set of data. Of course, some areas in AI would be better suited than others to the concept of generality. For instance, we can understand the notion of generality for a planner (Long & Fox, 2003) (i.e., a general planner would be the one that is good for a wide range of planning problems). Nevertheless, it is still for learning systems where the notion of a general factor is more intuitive and closer to the original notions in human intelligence. There are already several 'experiment databases' (Vanschoren, Blockeel, Pfahringer, & Holmes, 2012; Vanschoren, van Rijn, Bischl, & Torgo, 2014) whose results can be used to analyse correlations, positive manifolds and whether SLODR (or ULOAR) is taking place there. Also, apart from the general video game competition studied here, new platforms for AI evaluation, where hundreds of tasks are integrated (Castelvecchi, 2016), can provide AI research with huge amounts of experimental data.

The need for more abstract and powerful techniques to analyse the behaviour of AI systems against a range of tasks, and the relation of these tasks with each other can be done theoretically or empirically (Hernández-Orallo, Dowe, & Hernández-Lloreda, 2014; Hernández-Orallo, 2017), but it is in increasing demand. For instance, the organisers and community around the GVGAI competition seek to "categorize the games based on properties in the games and how various algorithms play them" (Bontrager et al., 2016). The use of correlation matrices for AI competitions and benchmark results, and the use of bins in terms of ability and difficulty is just a tip of an iceberg of possibilities.

Acknowledgments

I thank the anonymous reviewers of ECAI'2016 for their comments on an early version of the experiments shown in Section 4. I'm really grateful to Philip J. Bontrager, Ahmed Khalifa, Diego Pérez-Liébana and Julian Togelius for providing me with the GVGAI competition data that made Section 3 possible. David Stillwell and Aiden Loe suggested the use of person-fit as a measure of generality. The JAIR reviewers have provided very insightful and constructive comments, which have greatly helped to improve the final version of this paper. This work has been partially supported by the EU (FEDER) and Spanish MINECO grant TIN2015-69175-C4-1-R, and by Generalitat Valenciana PROMETEOII/2015/013 and PROMETEO/2019/098. I also thank the support from the Future of Life Institute through FLI grant RFP2-152. Part of this work has been done while visiting the Leverhulme Centre for the Future of Intelligence, generously funded by the Leverhulme Trust. I also thank the UPV for granting me a sabbatical leave and the funding from the Spanish MECD programme "Salvador de Madariaga" (PRX17/00467) and a BEST grant (BEST/2017/045) from the Generalitat Valenciana for another research stay also at the CFI.

References

Andre, D., & Russell, S. (2002). State abstraction for programmable reinforcement learning agents. In *Proceedings of the National Conference on Artificial Intelligence*, pp. 119–125.

Aziz, H., Brill, M., Fischer, F., Harrenstein, P., Lang, J., & Seedig, H. G. (2015). Possible and necessary winners of partial tournaments. *Journal of Artificial Intelligence Research*, *54*, 493–534.

Bellemare, M. G., Naddaf, Y., Veness, J., & Bowling, M. (2013). The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, *47*, 253–279.

Bontrager, P., Khalifa, A., Mendes, A., & Togelius, J. (2016). Matching games and algorithms for general video game playing. In *Twelfth Artificial Intelligence and Interactive Digital Entertainment Conference*, pp. 122–128.

Boutilier, C., Reiter, R., Soutchanski, M., & Thrun, S. (2000). Decision-theoretic, high-level agent programming in the situation calculus. In *Proceedings of the National Conference on Artificial Intelligence*, pp. 355–362.

Carroll, J. L., & Seppi, K. (2005). Task similarity measures for transfer in reinforcement learning task libraries. In *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, Vol. 2, pp. 803–808. IEEE.

Castelvecchi, D. (2016). Tech giants open virtual worlds to bevy of AI programs. *Nature*, *540*, 323–324.

De Ayala, R. J. (2009). *Theory and practice of item response theory*. Guilford Publications.

Deary, I. J., Egan, V., Gibson, G. J., Austin, E. J., Brand, C. R., & Kellaghan, T. (1996). Intelligence and the differentiation hypothesis. *Intelligence*, *23*(2), 105–132.

Detterman, D. K., & Daniel, M. H. (1989). Correlations of mental tests with each other and with cognitive variables are highest for low IQ groups. *Intelligence*, *13*(4), 349–359.

Dimitrakakis, C., Li, G., & Tziortziotis, N. (2014). The reinforcement learning competition 2014. *AI Magazine*, *35*(3), 61–65.

Dowe, D. L., & Hajek, A. R. (1997). A computational extension to the Turing Test. In *Proceedings of the 4th Conference of the Australasian Cognitive Science Society, University of Newcastle, NSW, Australia, also as Technical Report #97/322, Dept Computer Science, Monash University, Australia.*

Dowe, D. L., & Hernández-Orallo, J. (2012). IQ tests are not for machines, yet. *Intelligence*, *40*(2), 77–81.

Dowe, D. L., Hernández-Orallo, J., & Das, P. K. (2011). Compression and intelligence: social environments and communication. In Schmidhuber, J., Thórisson, K., & Looks, M. (Eds.), *Artificial General Intelligence*, Vol. 6830, pp. 204–211. LNAI series, Springer.

Edmonds, B. (2009). The social embedding of intelligence. In Epstein, R., Roberts, G., & Beber, G. (Eds.), *Parsing the Turing Test*, pp. 211–235. Springer.

Elo, A. E. (1978). *The rating of chessplayers, past and present*, Vol. 3. Batsford London.

Embretson, S. E., & Reise, S. P. (2000). *Item response theory for psychologists*. L. Erlbaum.

Fogarty, G. J., & Stankov, L. (1995). Challenging the "law of diminishing returns". *Intelligence*, *21*(2), 157–174.

Genesereth, M., Love, N., & Pell, B. (2005). General game playing: Overview of the AAAI competition. *AI Magazine*, *26*(2), 62.

Genesereth, M., & Thielscher, M. (2014). General game playing. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, *8*(2), 1–229.

Gerevini, A., Saetti, A., & Vallati, M. (2014). Planning through automatic portfolio configuration: The pbp approach. *Journal of Artificial Intelligence Research*, *50*, 639–696.

Hernández-Orallo, J. (2000). Beyond the Turing Test. *J. Logic, Language & Information*, *9*(4), 447–466.

Hernández-Orallo, J. (2015a). On environment difficulty and discriminating power. *Autonomous Agents and Multi-Agent Systems*, *29*, 402–454.

Hernández-Orallo, J. (2015b). Stochastic tasks: Difficulty and Levin search. In Bieger, J., Goertzel, B., & Potapov, A. (Eds.), *Artificial General Intelligence - 8th International Conference, AGI 2015, Berlin, Germany, July 22-25, 2015, Proceedings*, pp. 90–100. Springer.

Hernández-Orallo, J. (2016). Evaluation in artificial intelligence: From task-oriented to ability-oriented measurement. *Artificial Intelligence Reviews*, *online*, 1–51.

Hernández-Orallo, J. (2017). *The Measure of All Minds: Evaluating Natural and Artificial Intelligence.* Cambridge University Press.

Hernández-Orallo, J., & Dowe, D. L. (2010). Measuring universal intelligence: Towards an anytime intelligence test. *Artificial Intelligence*, *174*(18), 1508 – 1539.

Hernández-Orallo, J., Dowe, D. L., & Hernández-Lloreda, M. V. (2014). Universal psychometrics: Measuring cognitive abilities in the machine kingdom. *Cognitive Systems Research*, *27*, 5074.

Hernández-Orallo, J., Martínez-Plumed, F., Schmid, U., Siebers, M., & Dowe, D. L. (2016). Computer models solving human intelligence test problems: progress and implications. *Artificial Intelligence*, *230*, 74–107.

Hutter, M. (2005). *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability.* Springer.

Igel, C., & Toussaint, M. (2005). A no-free-lunch theorem for non-uniform distributions of target functions. *Journal of Mathematical Modelling and Algorithms*, *3*(4), 313–322.

Insa-Cabrera, J., Dowe, D. L., & Hernández-Orallo, J. (2011). Evaluating a reinforcement learning algorithm with a general intelligence test. In Lozano, J., Gamez, J., & Moreno, J. (Eds.), *Current Topics in Artificial Intelligence. CAEPIA 2011.* LNAI Series 7023, Springer.

Jensen, A. R. (1998). *The g factor: The science of mental ability.* Westport, Praeger.

Jensen, A. R. (2003). Regularities in Spearman's law of diminishing returns. *Intelligence*, *31*(2), 95–105.

Legg, S., & Hutter, M. (2007). Universal intelligence: A definition of machine intelligence. *Minds and Machines*, *17*(4), 391–444.

Leike, J., & Hutter, M. (2015). Bad universal priors and notions of optimality. In *Conference on Learning Theory, CoLT*, pp. 1244–1259.

Leonetti, M., & Iocchi, L. (2010). Improving the performance of complex agent plans through reinforcement learning. In *Proceedings of the 2010 International Conference on Autonomous Agents and Multiagent Systems: volume 1*, pp. 723–730.

Lindauer, M., Hoos, H. H., Hutter, F., & Schaub, T. (2015). Autofolio: An automatically configured algorithm selector. *Journal of Artificial Intelligence Research*, *53*, 745–778.

Long, D., & Fox, M. (2003). The 3rd international planning competition: Results and analysis. *J. Artif. Intell. Res. (JAIR)*, *20*, 1–59.

Machado, M. C., Bellemare, M. G., Talvitie, E., Veness, J., Hausknecht, M., & Bowling, M. (2017). Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. arXiv preprint arXiv:1709.06009.

Madden, M. G., & Howley, T. (2004). Transfer of experience between reinforcement learning environments with progressive difficulty. *Artificial Intelligence Review*, *21*(3-4), 375–398.

Martínez-Plumed, F., & Hernández-Orallo, J. (2017). AI results for the Atari 2600 games: difficulty and discrimination using IRT. Evaluating General-Purpose Artificial Intelligence, August 20, 2017, 2nd Intl. Workshop held in conjunction with IJCAI, Melbourne.

Martínez-Plumed, F., Prudêncio, R. B. C., Martínez-Usó, A., & Hernández-Orallo, J. (2016). Making sense of item response theory in machine learning. In *ECAI 2016 - 22nd European Conference on Artificial Intelligence, 29 August-2 September 2016, The Hague, The Netherlands - Including Prestigious Applications of Artificial Intelligence (PAIS 2016)*, pp. 1140–1148.

McDermott, J., White, D. R., Luke, S., Manzoni, L., Castelli, M., Vanneschi, L., Jaśkowski, W., Krawiec, K., Harper, R., Jong, K. D., & O'Reilly, U.-M. (2012). Genetic programming needs better benchmarks. In *Proceedings of the 14th international conference on genetic and evolutionary computation conference*, pp. 791–798. ACM.

Mendes, A., Nealen, A., & Togelius, J. (2016). Hyperheuristic general video game playing. In *Proceedings of Computational Intelligence and Games (CIG). IEEE.*

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(26 February), 529–533.

Murray, A. L., Dixon, H., & Johnson, W. (2013). Spearman's law of diminishing returns: A statistical artifact?. *Intelligence*, *41*(5), 439–451.

Nielsen, T. S., Barros, G. A., Togelius, J., & Nelson, M. J. (2015). Towards generating arcade game rules with VGDL. In *Computational Intelligence and Games (CIG), 2015 IEEE Conference on*, pp. 185–192. IEEE.

Orseau, L. (2013). Asymptotic non-learnability of universal agents with computable horizon functions. *Theoretical Computer Science*, *473*, 149–156.

Perez, D., Samothrakis, S., Togelius, J., Schaul, T., Lucas, S., Couëtoux, A., Lee, J., Lim, C.-U., & Thompson, T. (2015). The 2014 general video game playing competition. *IEEE Transactions on Computational Intelligence and AI in Games*, *8*, 229–243.

Sala, G., & Gobet, F. (2016). Do the benefits of chess instruction transfer to academic and cognitive skills? a meta-analysis. *Educational Research Review*, *18*, 46–57.

Schaul, T. (2014). An extensible description language for video games. *Computational Intelligence and AI in Games, IEEE Transactions on*, *6*(4), 325–331.

Schiffel, S., & Thielscher, M. (2014). Representing and reasoning about the rules of general games with imperfect information.. *J. Artif. Intell. Res.(JAIR)*, *49*, 171–206.

Solomonoff, R. J. (1960). A preliminary report on a general theory of inductive inference.. Report V-131, Zator Co., Cambridge, Ma. Feb 4, revision, Nov.

Solomonoff, R. J. (1964). A formal theory of inductive inference. Part I. *Information and control*, *7*(1), 1–22.

Spearman, C. (1904). General Intelligence, Objectively Determined and Measured. *The American Journal of Psychology*, *15*(2), 201–92.

Spearman, C. (1927). *The abilities of man: Their nature and measurement.* Macmillan, New York.

Sternberg, R. J. (2000). *Handbook of intelligence.* Cambridge University Press.

Togelius, J. (2016). How to run a successful game-based ai competition. *IEEE Transactions on Computational Intelligence and AI in Games*, *8*(1), 95–100.

Tucker-Drob, E. M. (2009). Differentiation of cognitive abilities across the life span.. *Developmental psychology*, *45*(4), 1097.

Vanschoren, J., Blockeel, H., Pfahringer, B., & Holmes, G. (2012). Experiment databases. *Machine Learning*, *87*(2), 127–158.

Vanschoren, J., van Rijn, J. N., Bischl, B., & Torgo, L. (2014). OpenML: networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, *15*(2), 49–60.

Veness, J., Ng, K., Hutter, M., & Silver, D. (2011). A Monte Carlo AIXI Approximation. *Journal of Artificial Intelligence Research, JAIR*, *40*, 95–142.

White, D. R., McDermott, J., Castelli, M., Manzoni, L., Goldman, B. W., Kronberger, G., Jaśkowski, W., O'Reilly, U.-M., & Luke, S. (2013). Better GP benchmarks: Community survey results and proposals. *Genetic Programming and Evolvable Machines*, *14*, 3–29.

Whiteson, S., Tanner, B., & White, A. (2010). The Reinforcement Learning Competitions. *The AI magazine*, *31*(2), 81–94.

Wolfram, S. (2002). *A new kind of science.* Wolfram media, Champaign, IL.

Wolpert, D. H. (2012). What the no free lunch theorems really mean; how to improve search algorithms. Tech. rep., Santa fe Institute Working Paper.

Wolpert, D. H., & Macready, W. G. (1995). No free lunch theorems for search. Tech. rep., SFI-TR-95-02-010 (Santa Fe Institute).