Project 3 Reflection: Occam's Razor - Simple approaches solve 3x3 Raven's Progressive Matrices

Manikandan Ravikiran
mravikiran3@gatech.edu

1 INTRODUCTION, SOLUTION PROPOSAL & ALGORITHM

In this report, an agent to solve 3x3 (Set D & E) Raven's Progressive Matrices (RPM) (Raven, 1962) is presented (Figure 1b) . The developed agent solves **12/12 on Basic & Challenge sets** (Set E& D both) and **7/12 and 8/12 across Test sets** respectively. The report presents details on the incremental design of the solution, error analysis, efficiency, generality, cognitive connection and concludes with some implications on future work. The tutorial for understanding this report is in 5.1 and consolidated set of rules in 5.6.

Knowledge Representation, Agent Reasoning and Design: Inline to Project 1 & 2 [1] we use **using pixel based visual representations** for images (Figure 1a) and develop an production system with series of rules, that can solve one or more RPM's. Every production rule consist of an ***if case*** that helps in problem type identification and ***then case*** that calculates the solutions, for successful if cases. Much of these rules exploit relationships that exists across rows/column/diagonals (Kunda, McGreggor, and Goel, 2009) within the RPM images (Figure 1a) and use **Pixel Counts** & **Root Mean Square (RMS)** metrics for image similarity estimation.

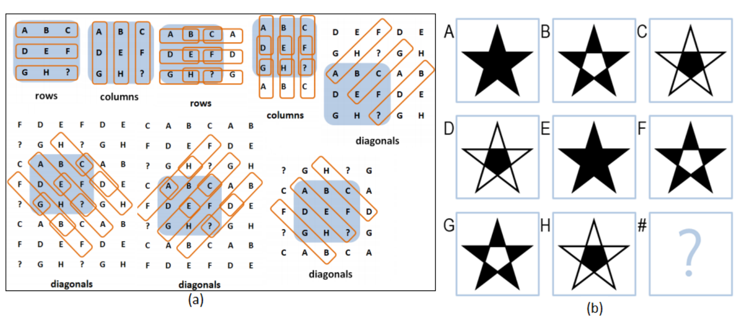


Figure 1—Affine symbolic Reasoning (a) and Sample 3x3 RPM (b).

The final developed production system consists of 33 production rules under 18 broad categories a.k.a ***if else*** cases incrementally developed by testing on auto-

1 Detailed description of algorithm used in project 1 & 2 is in section 5.3 & 5.5

grader, which encompasses simple relationships in RPM's. During processing, each of the rules is executed in sequence and whenever the input RPM violates a given rule, the agent moves onto the next rule otherwise computes the result and outputs the corresponding answer choice. These *18* broad category (see 5.6) of rules are explained across the submissions in section 2.

Performance Evaluation Metrics: Performance of the agent is accessed using accuracy, efficiency & generality metrics. Also, errors are categorized as **Wrong Principle (WP)** and **Incomplete Correlate (IC)** (Kunda et al., 2016) highlighted in blue and orange (Tables 1-9) with in-depth descriptions in sections 5.7 & 5.8.

2 EXPERIMENTAL RUNS AND DISCUSSION

This section presents and analyzes, submissions tested on the auto grader. Each submission's description begins with selecting one or more RPM's, followed by its analysis and solution description development, ending with cognitive connection, errors and improvement proposals. Also, basic problems and challenge problems are represented by **BP** and **CP** respectively.

2.1 Submission-1: Solving BP's (2019-11-07 15:45:39 UTC)

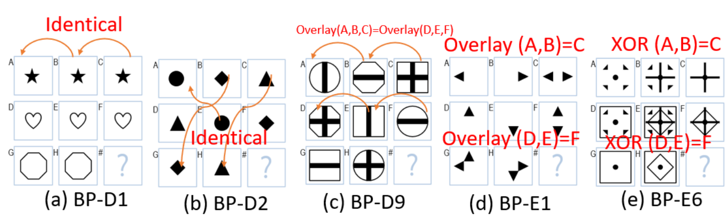


Figure 2—Examples of relationship across BP's in set E/D.

Intuition: To begin with, **manual analysis of basic problems in both sets D & E was carried out**. As seen in Figure 2, these problems satisfy simple relationships such as ***XOR, Overlay, Identity*** etc. Hence, rules shown in Figure 3 was added to the agent, with an option to return answer choice unseen in images (A-F) of the problem. The results so obtained are in Table 1, with total execution time of **27 secs**.

Cognitive Connection: Human solving process and its relationship with the RPM's is known to be incremental with typically human's solving one or more similar problems at a time (Carpenter, Just, and Shell, 1990). This can be seen in the current submission with an incremental solution by analysis of relation-

Rule – D1	If A=B=C and D=E=F then find answer choice S such that G=H=S
Rule – D2	If A=D or A=E or A=F and B=D or B=E or B=F or A=D and C=D or C=F or C=E the find answer choice S find such that S is missing in A to H
Rule – D3	If Overlay(A,B,C)=Overlay(D,E,F) then find S such that Overlay(G,H,S)=Overlay(A,B,C)
Logical Rules-Set E	If %ofblackpixels(**LOGICALOPERATOR**(A,B,C)> Threshold, then find answer choice S such that **LOGICALOPERATOR**(G,H,S)> Threshold, where **LOGICALOPERATOR={add,subtrack,and,xor}**

Figure 3—Rules developed for submission-1.

ships such as XOR, overlay, identity, etc. Further adding logical operations into RPM solving process causes agent behavior and reasoning strategy to be logical and in natural sync with human thinking (Axten, 1973). However the agent doesn't have any inductive reasoning for the solving process and unlike humans the agent doesn't have any meta-cognition built in to gauge the certainty of solutions, thereby resulting in the return of wrong answers rather than skipping problems. These observations are similar to Project 1 and 2 (See 5.3 & 5.5).

Table 1—Results from submission-1. ✓, ✗ & ✈ indicates correct, incorrect and skipped answers. WP & IC are highlighted.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	Basic	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✗	10
	Challenge	✗	✓	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	2
Set E	Basic	✓	✓	✓	✗	✓	✓	✓	✓	✗	✓	✓	✗	9
	Challenge	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	2

	Accuracy
Test	8
Ravens	6
Test	7
Ravens	4

Error Analysis: The agent produces a score of 8/12 & 7/12 on BP's and high error across the CP's. Analysis of errors reveals that for **BP-D8, BP-D12, CP-D10** for some forms of relationships between number black and white pixels. Further **CP-D3** follows relationship of identity diagonally. Further, the agent only solved 2 CP's in both sets, which adheres to existing rules, suggesting better rules and analysis are needed to handle CP's.

Improvement Proposal: The limitations of the agent could be improved considerably by adding new rules to tackle erroneous CP's and adapting the above analysis information.

2.2 Submission-2: Solving BP-D8, BP-D12, CP-D10 (2019-11-16 10:09:19 UTC)

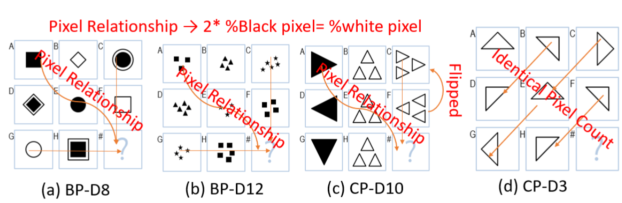


Figure 4—Relationships in BP-D8, BP-D12, CP-D10 & CP-D3.

Improvement Intuition and Rules: Based on error analysis from submission-1, we can see that BP-D8, BP-D12, CP-D10 adhere to relationship of $2 \times \%blackpixels = \%whitepixels$ *across CF, GH and AE. (Figure 4)* These information's were merged to form rules as shown in Figure 5. Problems still return unseen answer choice when none of the rules fail to come through. So the agent has rules from submission 1-2 combined as part of solving process to obtain results as shown in Table 2 with total execution time of **27.11 secs**. No improvements on test set was seen from previous submission.

Table 2—Results from submission-2. ✓, ✗ & ✈ indicates correct, incorrect and skipped answers. WP & IC are highlighted.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	**Basic**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✗	✓	✗	✗	✗	✗	✗	✓	✗	✓	✗	✗	3
Set E	**Basic**	✓	✓	✓	✗	✓	✓	✓	✓	✗	✓	✓	✗	9
	Challenge	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	2

	Accuracy
Test	8
Ravens	6
Test	7
Ravens	4

Cognitive Connection: After introducing the newer rules, the agent still mimics the way a human would think about the solution of overlaying up to some extent. ***Some extent***, because while identity along diagonals follow human thinking, the agent lacks features of analysis of images (induction) required for inferring pixel relationships. Further the agent's behavior can be considered to be similar to learning by adapting cases, where previously for BP-D2 is established similarity between different images, but adapted similarity computation for diagonal elements.

Rule - D4	If B=D and C=G and H=F, then find answer S such that A=E and E=S.
Rule - D5	Find answer S such that 2* L(X) = K(Y) where X = L(C)-L(S)+L(G)-L(H)+L(A)-L(E) and Y = K(C)-K(S)+K(G)-K(H)+K(A)-K(E), where L = %blackpixel and K= %white pixel
Rule - D6	If Overlay(A,B,C)=Overlay(D,E,F) then find S such that Overlay(G,H,S)=Overlay(A,B,C)

Figure 5—Rules developed in submission-2 & 3.

Error Analysis and Improvement Proposal: Basic set D was solved successfully so was problem CP-D10. In addition it can also be seen that CP-D3 (See 2.3) shows identity property along diagonals where *D=B, C=G and F=H* in terms of number of black pixels. Additionally, *CP-D6 & BP-E4* (See 2.4) shows row sum property where sum of elements across the rows are similar. Modifications to accommodate previous observations should improve the results across CP set. Each of these improvements are addressed in upcoming submissions.

2.3 Submission-3: Solving CP-D3 (2019-11-16 16:56:55 UTC)

Improvement Intuition, Rules & Performance: Analysis of submission-2 shows CP-D3 follows identity property across diagonals (see Figure 4). Moreover this similarity could be established through counting pixels without any transformations. Hence a rule was introduced as shown in Figure 5. Also for this submission, the code was optimized by adding avoid multiple local variable calculation and unnecessary image transformations. This resulted in performance as shown in Table 3. More specifically, adding this new rule improved performance as seen by comparing results from previous section. Further no changes in test set results and the execution time was reduced by 2 to **25secs**.

Table 3—Results obtained from submission-3.✓, ✗ & ✈ indicates correct, incorrect and skipped answers. WP & IC are highlighted.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✗	✓	✓	✗	✗	✗	✗	✓	✗	✓	✗	✗	4
Set E	Basic	✓	✓	✓	✗	✓	✓	✓	✓	✗	✓	✓	✗	9
	Challenge	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	2

	Accuracy
Test	8
Ravens	6
Test	7
Ravens	4

Cognitive Connection, Error Analysis & Improvement Proposal: No changes in cognitive connection since previous submission and no error analysis was done in this submission. Instead, an investigation was done to see, number of problems that are solved only by the added rules. Results revealed that the rule could be adapted similar problems like CP-D4, CP-E05, CP-E09 etc. As such adapting these rules together should improve performance. We will use this observation in section 2.5.

2.4 Submission-4 : Solving CP-D6, BP-E4 (2019-11-17 04:43:01 UTC)

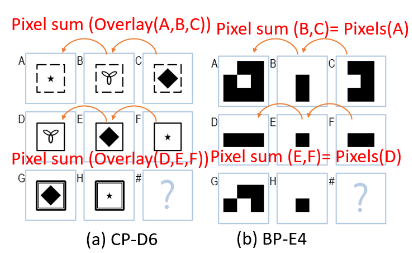


Figure 6—Relationships in CP-D6, BP-E4.

Improvement Intuition: Based on analysis from submission-2, we can see that CP-D6 & BP-E4 shows row sum property where sum of elements across the rows are similar. More specifically from Figure 6 we can see that for CP-D6 overlying the elements row wise results in similar objects across the rows i.e ***Pixelsum(Overlay(A,B,C)≈Pixelsum(Overlay(D,E,F)***. Further for BP-E4 we can see that *Figure A* = *Figure B*+*Figure C* i.e. ***PixelSum(B,C)=PixelCount(A)***. Both of the problem use common functions of counting pixels and overlay. These analysis was introduced as rule, shown in Figure 7

Performance: Introducing the rules from Figure 7, doesn't have any side effects and instead improves results across the said two problems. The agent obtained performance 12 and 10 on basic sets and 5 and 2 on challenge sets. No improvements on the test set were seen from the previous submission. The total execution time for the agent was **24 secs**.

Rule - D7	If (Count(A,B,C)-Count(D,E,F))-(Count(C)-Count(E)) < Threshold, then find S such that SUM(G,H,S) = SUM(A,B,C) where Count calculates total number of black pixels
Rule - E6	If Count(B)+Count(C) - Count(A) < Threshold and Count(E)+Count(F)=Count(D), then find answer choice S Count(S)+Count(H)=Count(G),

Figure 7—Rules developed in submission 4.

Cognitive Connection: The agent solves CP-D6 and BP-E4, without side effects. Further, the agents' behavior could be regarded as common sense reasoning, where the agent on seeing that image A is a combination of image B and image C, assumes that solution to follow similar mechanism. From the human cognition point of view, the behavior is similar, however, unlike human the agent doesn't have multiple view hypothesis testing on a single problem. For example for BP-E4, currently, the agent uses the count of black pixels across rows to find a solution. However, as a human one could think that the proposed problem could be solved by taking column relationships in a similar manner, which would help in generalization.

Table 4—Results obtained from submission-4.✓, ✗ & ✈ indicates correct, incorrect and skipped answers. WP & IC are highlighted.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	**Basic**	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✗	✓	✓	✗	✗	✓	✗	✓	✗	✓	✗	✗	5
Set E	**Basic**	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✗	10
	Challenge	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	2

	Accuracy
Test	8
Ravens	6
Test	7
Ravens	4

Analysis & Proposed Improvement: While cognitive connection points on column-based analysis are true, an error analysis was also carried out. Firstly we could see that problem CP-E6, CP-E10 exhibit row, and column-based relationships, however, these were unsolved by the current set of rules. **Overlay of images and**

row sum properties (previously BP-D9, D2) are satisfied by CP-D9 but problems were unsolved. **Approximate similarity property** (previously like CP-D6) can be seen in BP-E9, but the result was erroneous. Further CP-E08, like BP-D12 (see submission 1) exhibits relationships between a few problems that don't follow strict affine symbolic reasoning. Adapting these findings in existing rules should improve performance. We will use this in section 2.6.

2.5 Submission-5 : Solving CP-D4,D11,E5,E9 & BP-E12 (2019-11-17 13:13:25 UTC)

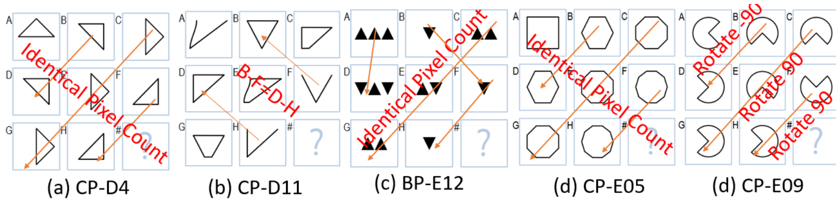


Figure 8—Relationship in CP-D4,D11,E5,E9 & BP-E12.

Improvement Intuition: Based on analysis from submission-3, we can see that CP-D4, CP-D11, BP-E12 and CP-E5, E9 follows relationships inline to submission-3. More specifically, as we can see from Figure 8. CP-D4, CP-E5 shows a diagonal identity relationship. CP-E9 shows diagonal identity combined with rotation. Similarly, CP-D11 shows relationship B-F=D-H in terms of visual analysis which in turn could be expressed through dark pixel count. Finally, BP-E12 shows an identity relationship between the problems that doesn't obey any conventions from (Kunda, McGreggor, and Goel, 2013). These analyses were introduced as rule, shown in Figure 9. Also, code was modified to remove unnecessary stubs.

Rule - D8	IF B-F = D-H, then find answer S that has maximum similarity such that 2*Count(H)=G and 2* Count(F)=C
Rule - D9	If B=D and C=G and H=F, then find answer S such that A=E and E=S. This is similar to submission 3 except we use different threshold to calculate similarity b/w images
Rule - E7	If Rotate90(D)=B and Rotate90(G)=C and Rotate90(H)=F, then find S such that Rotate-90=S
Rule - E8	If Count(A)=Count(D) and Count(B)=Count(F) and Count(F)=Count(H) and Count(G)=Count(C) then find S such that Count(HH)-Count(S) is minimum.

Figure 9—Rules developed in submission 4.

Performance: Introducing the rules from Figure 9, doesn't have any side effects and instead improves results across the said problems. More specifically, CP-E's got improved by 5 points so are BP's and CP-D's. The agent obtained performance 7 each on challenge sets. No improvements on the test set were seen. The total execution time for the agent was 22 secs, lower than all previous submissions.

Table 5—Results obtained from submission-5.✓, ✗ & ✈ indicates correct, incorrect and skipped answers. WP & IC are highlighted.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✗	✓	✓	✓	✗	✓	✗	✓	✗	✓	✓	✗	7
Set E	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	11
	Challenge	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✓	7

	Accuracy
Test	8
Ravens	6
Test	7
Ravens	4

Cognitive Connection & Error Analysis: No changes in cognitive connection since previous submission and no error analysis was done in this submission. Instead, solution to CP's D-9,D6,E-8 & BP's D-9,D-2,D-12 was developed and submission-6 was executed.

2.6 Submission-6 : Solving CP's D9, BP-E9 and CP-E6 (2019-11-18 10:15:30 UTC)

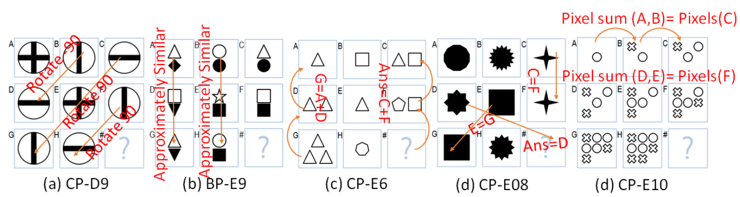


Figure 10—Relationships in CP's D9, BP-E9 and CP-E6.

Improvement Intuition: Based on analysis from submission-4, we can see that CP-D9 executes diagonal identity and rotation relationship. Meanwhile, we can see column-wise relationships in BP-E9, CP-E6. And all of the relationships are addition or overlay relationships. Finally, we have CP-E08 that have problems that are similar but don't confine to any of the rows/columns/diagonals. These observations see in Figure 10 are exploited to create rules as shown in Figure 11 below.

Rule - D10	If Rotate90(D)=B and Rotate90(G)=C and Rotate90(H)=F, then find S such that Count(A,B,C)=Count(G,H,S)
Rule - D11	If A~G and B~H then find S such that S~C
Rule - E7	If Count(A)+Count(D)=G then find such that Count(C)+Count(F)=Count(S)
Rule - E8	If Count(A)+Count(B)=Count(C) and Count(D)+Count(E)= Count(F) then find S such that Count(G)+ Count(H) = Count(S) and Count(S)-Count(E)is minimal
Rule - E9	If E=G and C=F then find S such that Count(S)-Count(D) is minimal

Figure 11—Rules developed in submission 6.

Performance: Introducing the rules from Figure 11, doesn't have any side effects and instead improves results across the said problems. More specifically CP-E's got improved by 2 points each across both the sets. Both BP's were completely solved. The agent obtained performance is 9/12 each on challenge sets. No improvements on the test set were visible. The total execution time for the agent was **24 secs**, due to the addition of these new rules.

Table 6—Results obtained from submission-6.✓, ✗ & ✈ indicates correct, incorrect and skipped answers. WP & IC are highlighted.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✗	✓	✓	✓	✗	✓	✗	✓	✓	✓	✓	✓	9
Set E	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✗	✓	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓	9

	Accuracy
Test	8
Ravens	6
Test	7
Ravens	4

Cognitive Connection: At this point, the agent is a complete production system for solving basic problems confirming to affine symbolic and similarity-based reasoning. Moreover, the agent learned concepts incrementally by generalizing for newer problems and adapting existing cases to newer problems as visible in rules across Figures 3-11. Much of these behavior draws parallel references from human cognition and behavior.

Error Analysis & Proposed Improvements: No error analysis was done & instead we proceed with submission-7.

2.7 Submission-7 : Solving CP-D1,5,7 & CP-E8,10 (2019-11-19 00:12:21 UTC)

Improvement Intuition and rules: Based on analysis from submission-4, the analysis was carried out. First, it can be seen that CP-E10, E08 exhibit row-wise relationships and identity relationships between problem images respectively. CP D-1 shows relationships between E, D and B images where *E=D and E=B*. Meanwhile *Overlaying(D,H)*$\approx$*D* & *overlay(B,F)*$\approx$*B* in CP-D7. These relationships were introduced as rules to solve the problems.

Table 7—Results obtained from submission-7.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
Set E	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12

	Accuracy
Test	6
Ravens	6
Test	7
Ravens	4

Performance: The agent solves 12 problems on all basic and challenge sets. Further, one of the test set problems was skipped reducing test performance. The final results are as shown in Table 7. **Total execution time is 24 secs**.

Cognitive Connection & Error Analysis: The agent is a complete system that uses abductive reasoning, where it starts with an observation or set of observations then seeks to find the simplest and most likely explanation for the observations in line with (Kunda et al., 2016). Further, since much of the BP's and CP's are solved no error analysis was done, instead of optimizing code and improving generalization was explored.

2.8 Submission-8 : Optimizing Efficiency (2019-11-21 12:48:25 UTC)

Improvement Intuition & Performance: By the end of submission-7, we can see that the net execution time for the complete the B-E sets are 24 secs. In that sense, considering the efficiency of the agent, for this submission code optimization was carried out. More specifically, for this submission, we removed multilevel function calls and instead replaced them with single global computations. Further, the number of variables for storing a variety of computations was increased. All the computations within the rules were modified to behave a single computation for each problem. **This resulted in the efficiency improvement of 3 secs resulting in the run time of 21 secs for B-E sets respectively.** No changes in accuracy was observed.

Cognitive Connection, Error Analysis & Improvement: No changes in cognitive connection since previous submission and no error analysis was done in this submission. Instead, an investigation was done to see if generalization could be improved. The analysis revealed that the drop in performance of test set D since submission 6, might be because of rule specialization.

2.9 Submission-9 : Improving Generalization (2019-11-22 09:05:04 UTC)

Improvement Intuition & Performance: In submission-6 we saw that increasing performance of challenge sets caused a drop in results of Test set D from 8 to 6, moreover there was no improvement across test sets since submission-2, to alleviate this and improve performance, here we focus of combining and pruning of developed rules. More specifically, for problems CP D2 & D3, CP E10 & E2 in submission we merged their similar rules. The rest of these rules were retained to obtain performance similar to that of the previous one. **In a sense, merging of similar rules has no impact. The total execution time was similar to the previous submission.**

Cognitive Connection, Error Analysis & Improvement: The system at this point follows version space learning, where based on the examples the model is specialized and generalized by pruning/merging the rules. However, merging is not implicit. Alternatively, we observed that ordering some of the rules has an impact on the results especially with a plurality of rules satisfying many problems.

2.10 Submission-10 : Improving Generalization (2019-11-24 10:49:05 UTC)

Improvement Intuition & Performance: For the final submission we modify the agent in two parts, firstly we resort to the rules where specific and hard thresholded rules are given higher preference. Further, each of these rules was further specialized to a fixed threshold that is satisfiable for the given problem. Further, this was done only for the rules of set D. Set E was unmodified, all the code optimizations were retained. The final results so obtained are as shown in Table 8. The reorganization and specialization rules improved the performance back to 7/12 for Test set D.

Table 8—Results obtained from submission-10.

		1	2	3	4	5	6	7	8	9	10	11	12	Accuracy
Set D	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
Set E	Basic	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
	Challenge	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12

	Accuracy
Test	7
Ravens	6
Test	7
Ravens	4

Cognitive Connection, Error Analysis & Improvement: The agent at this point uses conflict resolution strategy, where it uses the order in which production rules were written no assigned weights or priorities to production rules, rather sort the conflict set accordingly. While conflict resolution strategy is implemented, the method is indeed crucial to the efficiency and correctness of the production system, which is visible in the agent's latest performance. Further unlike some systems the agent simply doesn't fire all matching productions.

Table 9—Coverage of Rules for Basic Problems in Set-D.

		Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8	Rule 9	Rule 10	Rule 11	Rule 12	Rule 13	Rule 14	Rule 15	Rule 16	Rule 17	Rule 18
Basic	D-01	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✓	✗
	D-02	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
	D-03	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
	D-04	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓
	D-05	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓
	D-06	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
	D-07	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
	D-08	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-09	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-10	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓
	D-11	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
	D-12	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

3 CONCLUSION

Efficiency and Generality: Section 2.1-2.10 incrementally presents various rules designed, modified & adapted to improve results across all the sets. The developed agent achieved 12/12 on Basic, Challenge sets with at least 7/12 on Test sets. However, Raven's set performance requires improvement.

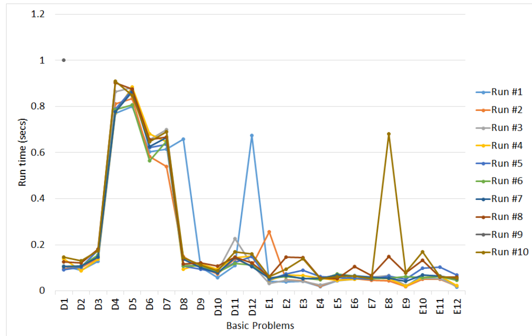


Figure 12—Efficiency of the agent on Basic Problems.

The final agent used a total of 18 rules set D and 15 rules for set E, under **18** broad categories across sections 2.1-2.10 that are specific to a group of problems as the agent was able to address 12 BP's and CP's, with a fair amount of success on Test and Raven's set. Table 9 shows the generalization of each rule of set D, across the basic set problems. Similar coverage details for set E is shown in Table 11. As we can see, incrementally adding new rules, increases the overall coverage of the system, more specifically the ***rule 18*** which corresponds to return unseen answer choices that cover a maximum of 5 BP's, while rest are very specific to each of the problems. Similar behavior was seen even in case of a challenge set (See section 5.7). Also as we can see, increasing problem complexity increases (problems D4-D8 & E7-E12) the agent's time consumption (Figure 12), the final agent consumes 21 secs to solve all the problems.

Human Cognition and AI: Inline with Project's 1 & 2 the production system's design simulates human thinking, with incremental discerning of relationships between the problems and adapting the rules across different problems. Further we can also see identification of mistakes and correcting them through the re-designing of the rules.

Considering execution time, human's initial RPM's in negligible time and **as problems become harder, the time consumption also increases**. Such a behavior can be seen in the agent where it spends lesser time on BP-(D1-D3) and more time on BP-(D4-D8). Further the agent solves the complete sets from B-E in ≈35 secs, which is similar to how fast humans solve RPM's, further the pattern of time consumption is very similar.

Further, we can see the agent uses more logical operations such as ***AND, XOR*** in the rules for reasoning on RPM's, this is similar to human behavior which appears to be logical but doesn't use any logic as part of the reasoning strategy (Carpenter, Just, and Shell, 1990).

Multiple aspects of the agent is similar to human reasoning strategy, firstly **multiple hypothesis analysis** where the agent solves RPM's through combination of multiple strategies like humans, **Common sense reasoning** - where the agents design uses similar rules for problems based on commonsense of similarity in problem types and **case based reasoning** where agent adapts the existing solution for new RPM's incrementally. Further agent exploits **constraints and uses different configurations** of rules to solve similar RPM's.

The **design includes concepts learned during the class**, where the system designed is a **production system** with series of rules with **case-based reasoning, correcting mistakes** where multiple different thresholds are devised using heuristics to adapt for newer problems, **partial order planning** where the rules are sorted to avoid wrong solutions. Also, similar to projects 1& 2 the **errors made by the agent are contradictory to human** testing, where most errors made by humans are repetition type (Kunda et al., 2016), while the **agent makes mostly wrong principle type errors**.

Finally, designing the agent based on visual representation, closely relates the agent to human's (Soulières et al., 2009), especially with **human relying on image relationship to solve the problems**.

Design Rectifications and Improvements: With the availability of unlimited time and resources, the following are the possible changes that can be done, to achieve more accurate results even more efficiently.

- **Threshold:** Inline with project-1 & 2, the problem of selecting a threshold still persists, previously it was selected manually however for this submission the threshold selection was empirical and also the number of thresholds is more. Instead of this, the images could be realigned to get a closed interval of thresholds.
- **Merging Rules:** Coverage analysis reveals that ***Rule 18*** alone solves the majority of BP's and some of the rules could be merged across the sets of D and E. As such for upcoming submissions and future works, merging similar rules with interval thresholds could be explored. Further, this would also improve the generalization of the method across the sets.

4 REFERENCES

[1] Axten, Nick (1973). "Human Problem Solving". In: *Contemporary Sociology* 2.2, pp. 169–170. ISSN: 00943061, 19398638. URL: http://www.jstor.org/stable/2063712.

[2] Carpenter, Patricia, Just, Marcel Adam, and Shell, Peter (1990). "What one intelligence test measures: a theoretical account of the processing in the Raven Progressive Matrices Test." In: *Psychological review* 97 3, pp. 404–31.

[3] Kunda, Maithilee, McGreggor, Keith, and Goel, Ashok K. (2009). "Addressing the Raven's Progressive Matrices Test of "General" Intelligence". In: *AAAI Fall Symposium: Multi-Representational Architectures for Human-Level Intelligence*.

[4] Kunda, Maithilee, McGreggor, Keith, and Goel, Ashok K. (2013). "A computational model for solving problems from the Raven's Progressive Matrices intelligence test using iconic visual representations". In: *Cognitive Systems Research* 22–23, pp. 47–66.

[5] Kunda, Maithilee, Soulières, Isabelle, Rozga, Agata, and Goel, Ashok K. (2016a). "Error patterns on the Raven's Standard Progressive Matrices Test". In: *Intelligence* 59, pp. 181–198. ISSN: 0160-2896. DOI: https://doi.org/10.1016/j.intell.2016.09.004. URL: http://www.sciencedirect.com/science/article/pii/S0160289616300149.

[6] Kunda, Maithilee, Soulières, Isabelle, Rozga, Agata, and Goel, Ashok K. (2016b). "Error patterns on the Raven's Standard Progressive Matrices Test". In: *Intelligence* 59, pp. 181–198. ISSN: 0160-2896. DOI: https://doi.org/10.1016/j.intell.2016.09.004. URL: http://www.sciencedirect.com/science/article/pii/S0160289616300149.

[7] Raven, John C. (1962). "Manual for Raven's progressive matrices and vocabulary scales". In: San Antonio, Texas: Pearson.

[8] Soulières, Isabelle, Dawson, Michelle, Samson, Fabienne, Barbeau, Elise Brochu, Sahyoun, Chérif P., Strangman, Gary, Zeffiro, Thomas A., and Mottron, Laurent (2009). "Enhanced visual processing contributes to matrix reasoning in autism." In: *Human brain mapping* 30 12, pp. 4082–107.

5 APPENDIX

5.1 Tutorial

This section presents a simple tutorial to understand the overall structure of the report and identify important elements for scoring purposes.

1. The entire report is divided in to three parts
   - **Introduction:** This summarizes the idea and approach used in the project and some connection to project 1. This is in Pages 1-2
   - **Submission:** This presents each submissions with details on approach for submission, modifications from previous submissions with results, error analysis and cognitive connection. This is in pages 3-11
   - **Conclusion:** Conclusion is presented in pages 12-13, with analysis of efficiency, coverage and cognitive connection.
2. Each submission section has two figures, one to show the logic behind the production rule and other showing the developed rules itself.
3. Every submission shows either addition of new rules or adapting some existing rule for new problems. This can be seen in very first ***Intuition and Rules*** section. Some cases ***rules*** section follows ***Intuition*** section.
4. Further in some submissions the results include code and efficiency optimization.
5. Results are all in the Tables 1-8.

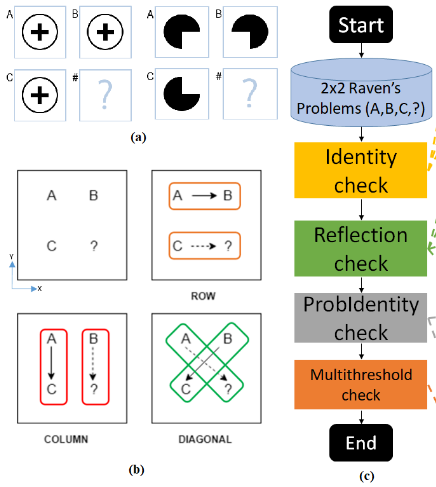
5.2 Representation and Reasoning

Knowledge Representation: Pixel based **visual representations** for images are used (see figure 1a), based on intuitions from (Carpenter, Just, and Shell, 1990) which suggest that pairwise spatial relationships between the problem are exploited during human problem solving process, which is represented as structural correspondence between the problem input images.

Reasoning: For submissions combination of Affine Symbolic reasoning and similarity based reasoning is used, where RPM problem is viewed as a sequence of images, where some transformation T can transform one image into a corresponding adjacent image (Kunda, McGreggor, and Goel, 2009).

5.3 Project 1: Algorithm

Algorithm: The final developed algorithm and process flow for project-1 are as shown in Figure 13 respectively, which consists of nine production rules a.k.a ***if-else*** cases which encompasses three broad categories relationships, namely ***Identity***, ***Reflection***, ***ProbIdentity*** & ***Multithreshold*** respectively. Each of the rules are described in Figure 13 and is developed incrementally by testing on the auto grader. During processing, each of rules are executed according to the numbered sequence **(Rules #)**. Whenever the input RPM violates a given rule, the agent



Rules #	Name	Description
1	Identity (A, B, C)	If A, B and C are identical, find image like C from answer choice
2	Identity (A, B)	If A and B are identical, find image like C from answer choice
3	Identity (A, C)	If A and C are identical, find image like B from answer choice
4	ReflectionY (A, B)	If vertically flipped A and B are identical, find image like vertically flipped C from answer choice
5	ReflectionX (A, C)	If horizontally flipped A and C are identical, find image like horizontally flipped B from answer choice
7	ReflectionY (A, C)	If vertically flipped A and C are identical, find image like vertically flipped B from answer choice
8	ReflectionY (B, C)	If vertically flipped B and C are identical, find image like vertically flipped A from answer choice
6.1	ProbIdentity (C, B)	Find image whose absolute difference of the RMS values for CB and AK pairs is the least, where K is set of answer choices
6.2	ProbIdentity (C, A)	Find image whose absolute difference of the RMS values for CA and BK pairs is the least, where K is set of answer choices
9	Multithreshold (A, B, C)	If Euclidean (B, C) in range (90,160) and RMS (C, A) in range (970, 990) then ProbIdentity (C, A)

(d)

Figure 13—Sample ravens matrices (a), Affine Relationships (b) Process flow (c) and production rules (d).

moves onto the next rule, otherwise computes the result and outputs the corresponding answer choice.

5.4 Image Similarity Metrics

Root Mean Square (RMS): To get a measure of how similar two images are, root-mean-square (RMS) value of the difference between the images are calculated. If the images are exactly identical, this value is zero. The following function uses the difference function, and then calculates the RMS value from the histogram of the resulting image. Given two images X and Y of size $W x H$ the RMS is calculated as

$$RMS = \sqrt{\frac{1}{W x H} \Sigma_{i=1}^{n} \left(X_i - Y_i\right)^2} \tag{1}$$

Euclidean Distance: Given two binarized images X and Y of size $W x H$, Euclidean Distance is calculated as

$$ED = \sqrt{\Sigma_{i=1}^{n} \left(X_i - Y_i\right)^2} \tag{2}$$

If the images are exactly identical, this value is zero, else the similarity is decided using the threshold. In this work, multiple thresholds are selected for each of the rules. Most of them are selected empirically by running experiments on the Basic Set of problems.

5.5 Project 2 - Algorithm

The final production system consisted of 17 rules with varying complexity, encompassing plurality of relationships that identifies and solves the RPM. The consolidate set of production rules are shown in Figure 14.

Rules #	Name	Description
1	Identity	If A=B and B=C then find image same as H from answer choices.
2	DiagonalFlip	If **Flip**(C)=G and **Flip**(F)=H and **Flip**(B)=D then find image like E from the answer choice
3.1	Rotate270	If **Rotate270**(C) = G and **Rotate270** (F) = H and **Rotate270** (B) = D, then find image like E from the answer choice
3.2	LogicalRotate270	If **Rotate270**(C) = G and **Rotate270** (F) = H and **Rotate270** (B) = D and A && B = E && A, then find answer **S** such that E && **S** = H && F
4	RowFlip (A, C)	If **Flip**(A)=C and **Flip**(D)=F, then find image like Flip(G)
5	AdjrowXOR	If **XOR** (A, D) =**XOR** (D, G) and **XOR**(H,E)=**XOR**(B, E) then find answer **S** such **XOR**(C,F)=**XOR**(F,**S**)
6	AdjcolXOR	If **XOR** (A, C) =B and **XOR** (D, F) =E then find answer **S** such **XOR**(G,**S**)=H
7	AdjDiaXOR	If **XOR** (A, E′) =E then find answer **S** such **XOR** (E′, **S**) =E
8	N2NXOR	If **XOR** (B, C) =A and **XOR** (E, F)=D then find image S such that **XOR**(S,H)=G
9	TowardsIdentity	If **ED**(B,A)>**ED**(C,B) and **ED**(D,E)>**ED**(E,F) and **ED**(A,B,C)<=ED(**A**,D,G) and **ED**(D,E,F)>=**ED**(A,B,C) the find answer S such that **ED**(G,H)>=**ED**(H,S)
10	ATowardsIdentity	If **ED**(B-A)<=**ED**(C-B) and **ED**(D-E)>=**ED**(E-F) and **ED**(A,B,C)=ED(**A**,D,G) and **ED**(D,E,F)>**ED**(A,B,C) the find answer **S** such that **ED**(G,H)>=**ED**(H,S)
11	ComplexRule	If **ED**(A,B,C)> **ED**(A,D,G) and **ED**(A,D)<= **ED**(D,G) and **ED**(B,E) <= **ED**(E,H) and **ED**(A,D,G)> **ED**(B,E,H) and **ED**(C,F) > **ED**(B,E) > **ED**(A,B) and **ED**(E,F) > **ED**(E,H) and **ED**(F,H) > **ED**(B,F) and **ED**(B,D)> **ED**(C,E) and **ED**(BC)-ED(GH) <threshold then find choice S such that **ED**(**XOR**(A,E),**XOR**(E,S)) is minimum
12	DiagonalRotate270	If **Rotate270**(B) = D and **Rotate270** (F) = H and **Rotate270** (C) = G, then find answer **S** such that **ED**(**S**, H)=**ED**(F,H)
13	PartsCompare	If **LeftPart**(A)=**RightPart**(C) and **RightPart**(A)=**LeftPart**(C) then find answer S such that **LeftPart**(G)=**RightPart**(S) and **RightPart**(G)=**LeftPart**(S)
14	AComplexRule-1	If **ED**(A,B,C)> **ED**(A,D,G) and **ED**(A,D)<= **ED**(D,G) and **ED**(B,E) <= **ED**(E,H) and **ED**(A,D,G)> **ED**(B,E,H) and **ED**(C,F) > **ED**(B,E) > **ED**(A,D) and **ED**(E,F) > **ED**(E,H) and **ED**(F,H) > **ED**(B,F) and **ED**(B,D)> **ED**(C,E) and **ED**(BC)-ED(GH) <threshold then find choice S such that **ED**(**XOR**(A,E),**XOR**(E,S)) is minimum
15	AComplexRule-2	If **ED**(B,C)< **ED**(A,B) and **ED**(D,E)<= **ED**(E,F) and **ED**(B,E) <= **ED**(E,H) and **ED**(A,D,G)< **ED**(B,E,H) and **ED**(A,B,C)< **ED**(D,E,F) and **ED**(E,F) > **ED**(E,H) and **ED**(G,H) < **ED**(E,F) and **ED**(A,C) > **ED**(A,G) and **ED**(C,F)> **ED**(G,H) and **ED**(A,G) <threshold:
16	AComplexRule-3	If **ED**(A,D)< **ED**(D,G) and **ED**(E,F)> **ED**(E,H) and **ED**(B,E) <= **ED**(E,H) and **ED**(A,D,G)< **ED**(A,B,C) and **ED**(B,E,H)> **ED**(A,D<G) and **ED**(F,H) > **ED**(B,F) and **ED**(B,D) < **ED**(C,E) and **ED**(A,D) > **ED**(D,G) and **ED**(B,E)> **ED**(E,H) and **ED**(C,F) > **ED**(B,E) > **ED**(A,D) and **ED**(BC)-**ED**(GH) <threshold
17	PixelRule	Find answer choice **S** that has maximum similarity such that 2* **NoofBlackPixels**(H)=G and 2* **NoofBlackPixels**(F)=C is satisfied

Figure 14—Production rules developed for project 2.

5.6 Project-3 Rules

The consolidated set of rules use in project 3, is as shown in Figure 15 below.

Rule - D1	If A=B=C and D=E=F then find answer choice S such that G=H=S
Rule - D2	If A=D or A=E or A=F and B=D or B=E or B=F or A=D and C=D or C=F or C=E the find answer choice S find such that S is missing in A to H
Rule - D3	If Overlay(A,B,C)=Overlay(D,E,F) then find S such that Overlay(G,H,S)=Overlay(A,B,C)
Logical Rules-Set E	If %ofblackpixels(**LOGICALOPERATOR**(A,B,C)> Threshold, then find answer choice S such that **LOGICALOPERATOR**(G,H,S)> Threshold, where **LOGICALOPERATOR={add,subtrack,and,xor}**
Rule - D4	If B=D and C=G and H=F, then find answer S such that A=E and E=S.
Rule - D5	Find answer S such that 2* L(X) = K(Y) where X = L(C)-L(S)+L(G)-L(H)+L(A)-L(E) and Y = K(C)-K(S)+K(G)-K(H)+K(A)-K(E), where L = %blackpixel and K= %white pixel
Rule - D6	If Overlay(A,B,C)=Overlay(D,E,F) then find S such that Overlay(G,H,S)=Overlay(A,B,C)
Rule - D7	If (Count(A,B,C)-Count(D,E,F))-(Count(C)-Count(E)) < Threshold, then find S such that SUM(G,H,S) = SUM(A,B,C) where Count calculates total number of black pixels
Rule - E6	If Count(B)+Count(C) - Count(A) < Threshold and Count(E)+Count(F)=Count(D), then find answer choice S Count(S)+Count(H)=Count(G),
Rule - D8	IF B-F = D-H, then find answer S that has maximum similarity such that 2*Count(H)=G and 2* Count(F)=C
Rule - D9	If B=D and C=G and H=F, then find answer S such that A=E and E=S. This is similar to submission 3 except we use different threshold to calculate similarity b/w images
Rule - E7	If Rotate90(D)=B and Rotate90(G)=C and Rotate90(H)=F, then find S such that Rotate-90=S
Rule - E8	If Count(A)=Count(D) and Count(B)=Count(F) and Count(F)=Count(H) and Count(G)=Count(C) then find S such that Count(HH)-Count(S) is minimum.
Rule - D10	If Rotate90(D)=B and Rotate90(G)=C and Rotate90(H)=F, then find S such that Count(A,B,C)=Count(G,H,S)
Rule - D11	If A~G and B~H then find S such that S~C
Rule - E7	If Count(A)+Count(D)=G then find such that Count(C)+Count(F)=Count(S)
Rule - E8	If Count(A)+Count(B)=Count(C) and Count(D)+Count(E)= Count(F) then find S such that Count(G)+ Count(H) = Count(S) and Count(S)-Count(E)is minimal
Rule - E9	If E=G and C=F then find S such that Count(S)-Count(D) is minimal

Figure 15—Production rules developed in this work.

5.7 Error Metrics

The details of each of the error metrics is as explained below.

Accuracy: We use **precision** as the accuracy metric, which computes fraction of problems correctly answered by the agent. The consolidates results across all the runs are as shown in Figure 16 below.

Efficiency: Efficiency is computed as time consumed in sec to solve a given problem i.e time consumed to execute the *Agent()* call for a given problem. Figure 9 for Basic Problems set D, where we could see that efficiency changed with the complexity of the problem. Figures 17 shows details of efficiency over all the sets together. Again the trend is consistent with basic set, where the agent takes more time to solve complex problems and vice versa.

Generality: Generality is evaluated as the fraction of problems, that a given rule can cover in the absence of other rules. The generality over challenge set is as shown in Table 10. The generality of developed production system on set E are shown in Table 11.

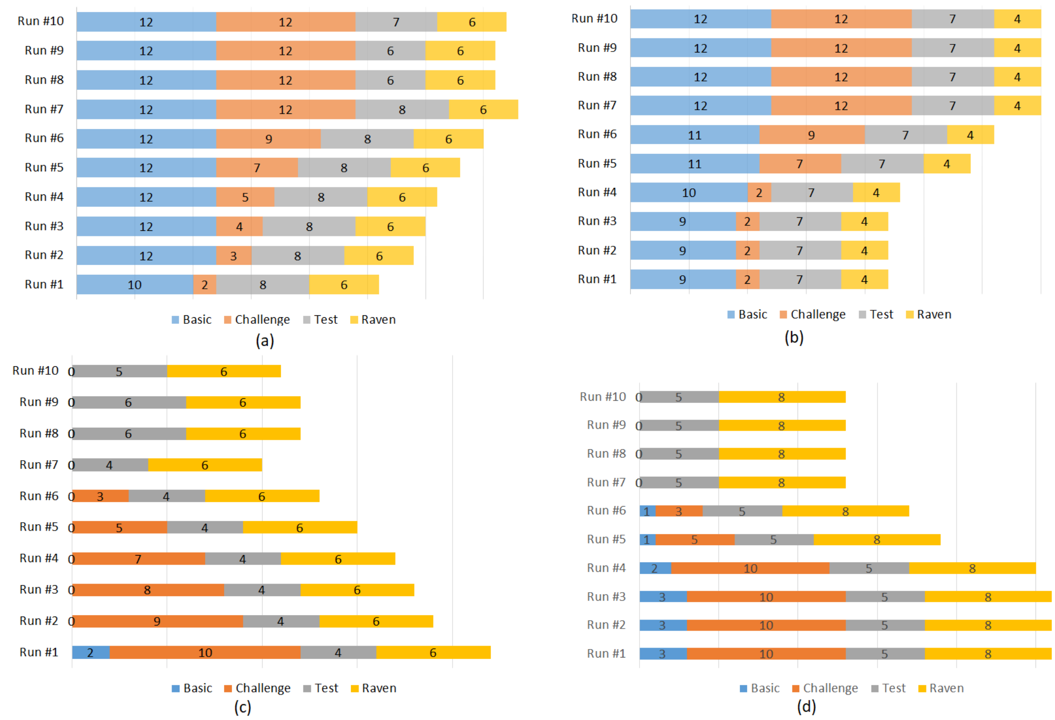


Figure 16—Consolidated results of Correct (a-b) & Incorrect (c-d) across set D & E.

Table 10—Coverage of Rules for Basic/Challenge Problems in Set-D.

		Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8	Rule 9	Rule 10	Rule 11	Rule 12	Rule 13	Rule 14	Rule 15	Rule 16	Rule 17	Rule 18
Basic	D-01	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✓	✗
	D-02	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
	D-03	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
	D-04	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓
	D-05	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓
	D-06	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
	D-07	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
	D-08	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-09	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-10	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓
	D-11	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗
	D-12	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Challenge	D-01	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
	D-02	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗
	D-03	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-04	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-05	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
	D-06	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
	D-07	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
	D-08	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
	D-09	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-10	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-11	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	D-12	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

5.8 Error Categories

Typical problem solving error's by humans could be categorized into four types namely 1) Repetition, 2) Difference, 3) Wrong Principle, and 4) Incomplete Cor-

Figure 17—Efficiency of the Production system on the basic & challenge sets.

relate. In this work, to establish a connection between the humans and the agent with respect to errors, these metrics were analyzed. However all of the error fall

Table 11—Coverage of rules on Basic and Challenge set E problems.

		Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8	Rule 9	Rule 10	Rule 11	Rule 12	Rule 13	Rule 14	Rule 15
Basic	E-01	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-02	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-03	✗	✗	✗	✓	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-04	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-05	✗	✗	✗	✗	✓	✓	✗	✗	✓	✗	✗	✗	✗	✗	✗
	E-06	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-07	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-08	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-09	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗
	E-10	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
	E-11	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
	E-12	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
Challenge	E-01	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-02	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗
	E-03	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗
	E-04	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗
	E-05	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
	E-06	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
	E-07	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗
	E-08	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
	E-09	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
	E-10	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗
	E-11	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗
	E-12	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗

under following two categories. Previously in sections 2.1-2.3, the categories of errors were highlighted. In this section, the definition and examples are presented in line with (Kunda et al., 2016).

Repetition: Repetition (R) errors occur when the chosen agent copies a matrix entry adjacent to the blank space. Choosing an R answer choice may represent some degree of perseveration or fixation on the problem matrix, such that an answer is selected using perceptual matching between the matrix entries closest to the blank space and the available answers. Answer choices 3 and 8 in Figure 18 are examples of Repetition errors.

Difference: Difference (D) errors occur when the chosen distracter is qualitatively different in appearance from the other choices. D kind of answer choices include those that are completely blank, as well as those that have extraneous shapes that are not found anywhere else in the problem. Answer choices 2 and 5 in Figure 18 are its examples.

Wrong Principle: Wrong principle (WP) errors occur when the chosen answer choice is a copy or composition of elements from the problem matrix. A WP answer might be chosen if the agent fails to identify the relationship from the matrix and instead combines the entries according to some other rule or relationship. Answer choices 4 and 6 in Figure 18 are examples of wrong principle

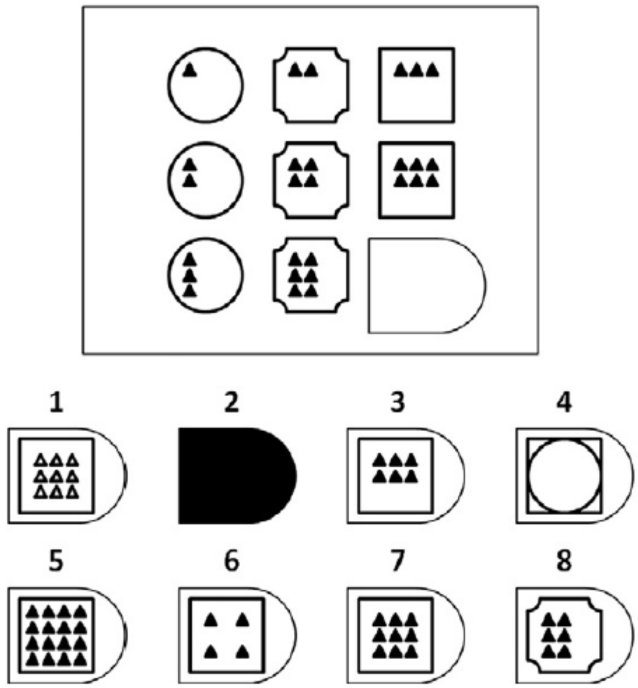


Figure 18—Example of an RPM-like problem. The correct answer is 7. (Kunda et al., 2016)

category of errors.

Incomplete Correlate: Incomplete correlate (IC) errors occur when the chosen answer is almost, but not quite, correct. For example, some IC answer choices represent a rotation or reflection of the correct answer. Answer choice 1 in Figure 18 is an example of an IC. Consolidated error statistics of various errors are as shown in Figure 19.

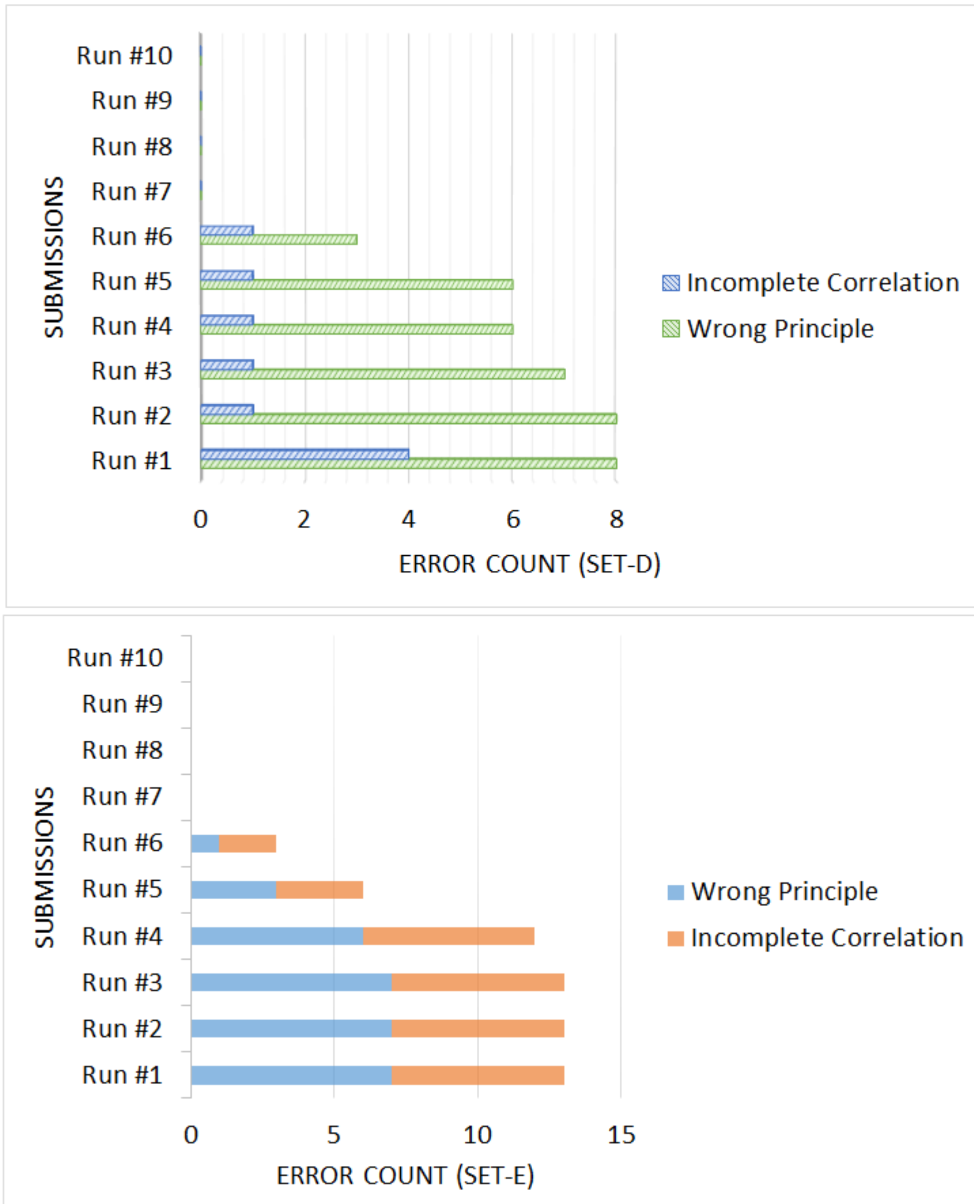


Figure 19—Consolidates error statistics across Basic and Challenge Test sets similar to (Kunda et al., 2016). Set D on top and E on bottom.