Exploring How Computation Can Foster Mathematical Creativity in Linear Algebra Modules

Sarah D. Castle
Michigan State University

The mathematics education community has simultaneously experienced a call for mathematical creativity and the implementation of computing within the classroom. What has not been explored are the ways in which these two calls are complementary. This study investigates how computation, enacted through coding, can bring about mathematical creativity within a series of linear algebra computational notebooks designed using the Understanding by Design framework. Specifically, through the analysis of a group of students, this paper highlights how a computational prediction and reflection cycle enables exploration and student originality while also facilitating connections to multiple representations.

Keywords: Computation, Linear Algebra, Mathematical Creativity, Coding

Mathematics – a field of beauty, creativity, and innovation, yet what is echoed by students is not how mathematics allows students to explore, but rather the ways in which it constrains and does not necessitate creativity (Silver, 1997). Students state mathematics is "a dead subject [with] hundreds of methods and procedures to memorize that [students] will never use, and hundreds of answers to questions that they have never asked" (Boaler, 2022, p. 35). This decline in creativity within school mathematics has coincided with the evolution of computation. Computation is now a scientific pillar (Skuse, 2019) as well as a pedagogical tool within the mathematics classroom (Castle, 2022; Lockwood, 2022). Further, many students associate coding with freedom to create and express themselves (Isomöttönen et al., 2020) which is in sharp contrast with students' views of mathematical creativity. Therefore, a natural question that arises is what if computation could offer mathematics education more, specifically with relation to mathematical creativity. This study is part of a larger research project that aims to address the ways in which computation enacted through coding can aid mathematical understanding and promote mathematical creativity, while preparing students for careers in the upcoming computational world. Within this paper, I seek to answer the following research question: *What are the affordances and constraints that computation, as enacted in coding through a creative environment, provides for students' learning of linear algebra concepts and procedures?*

Theoretical Perspective

Cultural Historical Activity Theory (CHAT) is based on the of the work of Vygotsky, Leontiev, and Engeström (1987). It provides a framework for studying collective activity, specifically emphasizing the role of mediated activity, a key consideration when considering computing. Within CHAT, subjects are the people involved in the activity, whereas the object is the aim of the activity. Subjects' actions are mediated by artifacts or tools. Further, it is imperative that individuals are understood within the context of their actions, and that the social elements must be brought in. Therefore, within this study individuals cannot be divorced from their group and context. There are different levels at which you can analyze the relationship between subject and object: activities, actions, and operations (Leontiev, 1974). Activity is composed of actions, and actions are composed of operations (Batiibwe, 2019). Operations are automatized or routinized behaviors and many times do not require conscious effort (Jonassen & Rohrer-Murphy, 1999). The conditions determine the operations. At the next level up, the goal an individual holds results in actions, but the goal is also affected by the conditions. This is the functional level that uses planning and problem-solving to complete the activities. Finally, it is motive that generates activity, which is composed of actions, and the motive determines the goal. This hierarchical structure is used in order to operationalize activity theory within observations, as there are multiple units of analysis that can be used to build up the overall activity. As the action level is what is of interest within this study, applying this framework necessitates establishing what actions are manifestations of mathematical creativity, thereby providing a way of documenting how the relationship between the students and the aim (mathematical creativity) is mediated by the tool (Jupyter notebooks).

Mathematical Creativity

In order to engage with the notion of mathematical creativity it is important to define what I mean, as there is a plethora of definitions. Based on literature (Haylock, 1997; Savic, 2016; Savic et al., 2017; Silver, 1997; Sriraman et al., 2013) I define mathematical creativity as:

> A process that can be fostered, is context-dependent, and manifests in fluency, flexibility, and originality with respect to the mathematics which results in the connection and abstraction of ideas that bring a new perspective that is of use to the community and is many times the result of prolonged work and reflection.

It is important to note that the focus is on creativity as a process, rather than a trait. This is an important distinction as it means there is the possibility to design opportunities to foster creativity and it is not a stagnant attribute of students. Further, I draw from mathematical creativity rubrics (Blyman et al., 2020; Henriksen et al., 2015; Savic et al., 2017) in order to establish an operationalization of mathematical creativity - specifically six dimensions of creativity: *originality* (ability to try novel or unusual approaches towards a problem), *flexibility* (ability to use multiple methods for solving the same problem), *fluency* (ability to apply the same mathematical idea, concept, or procedure, to a variety of problems and situations), *visualization* (development and use of illustrations, either physical or mental, to clarify or present concepts), *elaboration* (ability to establish meaningful connections between concepts), and *risk* (willing to take action where result is unknown, or a novel approach, to advance problem solving process).

Methodology

This paper reports on data collected from a study investigating the role of computation in developing mathematical creativity. This study consisted of 8 different students recruited from an introduction to computational modeling course (Silvia et al., 2019) who met recurringly with a small group (2-3 members) for a total of 6 weeks. Each week students engaged with a Jupyter Notebook designed to introduce new linear algebra concepts while fostering opportunities for mathematical creativity and understanding for a total of two hours. Afterwards students would complete a reflection about the module. Additionally, I conducted semi-structured pre- and post-study interviews with all participants as well as pre- and post-surveys. For the purposes of this paper, I will focus on a sole group composed of three students: Ivy, Alex, and Kylie, all of whom were non-mathematics STEM majors. This was done for space considerations, allowing an in-depth exploration into their experience across the study and for a more nuanced understanding.

Module Design

An underlying portion of this study centers around the design of the computational modules, accomplished by implementing Understanding by Design framework (Wiggins & McTighe, 2005). This process employs backwards design where the learning goals are first established, then potential evidence for meeting the goals, and finally the activity design itself using the prior work. A textbook, syllabi, and curriculum analysis yielded potential topics and desirable understandings. As this was not an official course, nor a prerequisite, I had latitude in selecting the broad understandings that the modules would center on, specifically for students to develop geometric understandings of linear algebra ideas and properties, as well as how to represent and model systems using their understanding of these concepts. Learning goals were established for each module and the evidence for meeting these goals were written out. Finally, pairing the tasks with opportunities to foster mathematical creativity as well as computational pedagogies such as use-modify-create cycles (Lee et al., 2009) resulted in the final construction of these modules.

Data Analysis

For the results shared within this paper, I transcribed and coded all interviews, observations, and reflections along the mathematical creativity dimensions and whether they were mediated through the notebook. The unit of analysis for coding was individual actions, as determined from the CHAT framing. I then wrote analytical memos detailing the initial themes of mathematical creativity followed by grouping codes along the dimensions of creativity and then along whether the action was mediated, resulting in sets of common elements. These groupings gave rise to two main themes within this set of students' experiences: (1) computation enabled experimentation via prediction and reflection cycles and (2) prediction and reflection facilitated the connection of multiple representations.

Results

Computation Enabling Experimentation Within Mathematics via Prediction and Reflection

During the fifth module, students were introduced to Markov chains. They wrote a function that would take in a state vector, a stochastic matrix, and the number of observations, and return the final state vector. They used the function to predict the population levels of a related suburb and city after 1, 2, and 5 years. The following exchange occurred after writing their functions:

Kylie: I'm just like running them all. Uh, it basically just keeps going in the same direction.

Alex: I think eventually it would, I don't know if we did, like, a thousand that would probably make our computers freeze, but um, would eventually it balance out somewhere.

Ivy: Maybe we can try like 10 and see what happens.

Alex: Maybe balance out, balance out isn't the right word. But eventually it would like 50-50, the directions would've reversed. Yeah.

Ivy: Let's try this. Yeah. It reversed at 10. So, the city is less than the suburb.

Alex: I mean like eventually they would get like eventually a year would go by and the city would increase while the other one would decrease.

Within this example, a key point is that students were never explicitly prompted to explore beyond the 5-year mark. Rather, they were designing a function in order to be able to easily model the Markov chains, and then testing that function to check predictions for a model. After completing the task at hand, the group demonstrated both originality and risk by asking a question that was of interest with a coupled prediction. Note that they had not specifically encountered the concept of steady state yet. After this prediction they tested their hypothesis, using their function, and then reflected on the output itself. Within this example, the computational environment enacted through coding enabled a cycle of prediction and reflection, which ultimately later led to a concept of steady state vectors. The coding enabled function creation, which allowed for an efficient way to engage in experimentation. The students were able to create a prediction and then check their result.

The fourth module focused on linear transformations and matrices as linear transformations. Prior to the discussion, the students had just finished up with figuring out how to take a matrix of points that defined a cat outline and double the cat image in size. They had experimented with this and graphed the resultant set of points. The next prompt within the notebook was *"It will now be your turn to explore! See what happens when you try different matrices. Use the cell below to document your thoughts or whatever you find easiest! Try and state what you think will happen before you run the code".* This was followed by multiple cells designated for students to try different matrices. This portion was specifically designed for students to experiment.

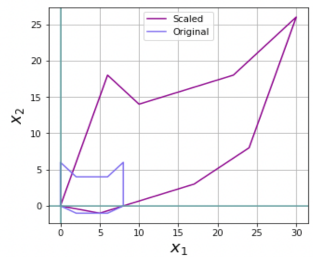

Figure 1. The output from Ivy's Jupyter Notebook when using a matrix transform along a set of points.

Ivy: I wonder if you like had like 3, 1, what would happen like anything that's not a zero.
Alex: Or you wanna do 3, 1, 3, 1. *(Referring to a 2x2 matrix)*
Kylie: Do you think it will work?
Ivy: Um, we can say, we don't think it will work.
Alex: What do you mean? What do you think?
Kylie: I guess it will just change. Like it'll disproportionately change the width and length.
Alex: Uh, I think the X values will be three times X plus Y and the Y values will be, uh, Y plus three X. So I think it, the X and Y will turn out to be the same, if that makes sense…
Ivy: So you think that it's gonna, they're gonna be multiplied by three, but then the value, like a value's gonna be added to them as well…
Kylie: Okay. And then what do we think that's gonna do the graph. It's gonna make it larger and move it?
Alex: I don't think it'll look like a cat anymore.
Kylie: Yeah. It's gonna move him around. So it's gonna like break up the lines. That is correct. It looks really weird. Oh! *[Output graphic shown in Figure 1.]*
Ivy: It kind of looks like it just stretched linearly. Like, kind of like on a line.
Kylie: It's it looks like if you took a side of it and just pulled yeah. You like took one of the cat ears and pulled it.
Ivy: So it's still cat.
Alex: I think it, it kinda looks like you took, like, we took the lines and multiplied them each by like some number, but it looks consistent, you know? You know what I'm saying?
Ivy: Kinda like, it just looks linear to me. I don't know why.
Alex: So for the next one, what do you guys wanna do?
Kylie: What about if we did it like negative numbers?
Alex: Oh, it'll flip it over. Like the X, Y or something?

This process continued on and the students each completed 6-8 different trials, deriving their own novel transformations, and plotting the visualization. From here, students were then asked to define the matrices that would correspond to the points undergoing: reflection across each axis, lines, and through the origin, as well as contractions and expansions, and shears. During each of these phases students then took what they had learned through experimentation to abstract, make meaningful connections, and create general representations of the types of transformations and the corresponding matrix. If they were usure, then they would experiment, using the insight they gained during the first part. Then they used code to validate their approaches and show the transformation on a set of points.

During this activity, there was specific reference to 'exploring' within the prompt. However, it was previously shown that students engaged in experimentation via prediction and reflection even when not prompted. This new excerpt highlights the potential ways in which the prediction and reflection cycle can take place. Specifically, the students were able to predict the ways that a 2x2 transformation matrix would affect an image and how the corresponding vertices would shift. The experimentation highlighted their initial choice in what matrix to start with and followed by different modifications. This allowed for their thoughts to guide the exploration. Further, during this experimentation, students were not simply 'pushing buttons'. Rather, they actively engaged in the sense making and making predictions based on their current understanding of the materials. This exploration piece allows for an inquiry-oriented approach to transformations, and the computational environment is something that enabled this to be fostered, especially the visualization. This was drastically different than previous mathematics experiences for most students. When reflecting on the experience as a whole, Alex stated:

> "I know it was a math, you know, teaching us math, but it didn't feel like math the same way as like solving integral does … it was more so recognizing like what was changing … it felt much more like recognizing patterns... Yeah. Just, just figuring that out. It didn't feel so much like, step by, I move this over here and then simplify this, it felt more like, playing a game, trying to, trying to figure out the way to get to the end goal."

This quote highlights the fact that the experimentation portion felt different to Alex than previous math classes. Although he does not explicitly name experimentation, he discusses this concept of changing pieces and looking for patterns to get to the end goal, which mirrors the notion of prediction and reflection. Within the previous examples from the observations, the group engaged in making changes in order to figure out the corresponding effect, thereby developing a deeper understanding of the system. This experimentation fostered via computation countered Alex's prior notions of mathematics, and specifically gave a framing of exploration rather than following a series of given steps, thereby promoting opportunities for creativity.

Prediction and Reflection Facilitating Connections to Multiple Representations

The second area highlighted during this pilot study was the ways in which the computational environment facilitated students making connections to multiple representations. To start, consider the following excerpt from the group's sixth module. It shows the interplay between group members and the ways in which the development and running of code supports multiple representations. Within this activity, participants were provided a function that takes in three vectors and displays the corresponding parallelepiped. Kylie was sharing her screen with the group at this time, and began by plotting the identity matrix, which represents a unit cube. The group ensured the determinant was 1 since the volume of the cube was 1. They read through the plotting function to ensure they knew what was occurring and how to use the function. This is where the following excerpt picks up. The exchange has actions added within brackets in italics; also, some of the points of general conversation were removed for the sake of space.

Alex: So, let's use the columns of *A* as our vectors. Okay. So, um, I guess that kind of makes sense. Cause the determinant is like doing the, it's kinda like doing the cross products a couple times. Right. And cross products, like the area of a parallelogram. So anyway, um, that's our parallelepiped… Cool. What would cause the determinant of *A*, to be zero? Uh, if it were, how do you get a volume of zero?

Ivy: Would it be like a 2D shape?

Alex: Oh yeah! Yeah that makes sense.

Ivy: But how would we express that?

Alex: We could try different matrices. So v_1, v_2, and v_3 are probably all gonna be in the same plane is my guess. I guess. Yeah but how do we –

Ivy: Should we set one of them to all zeros? No –

Kylie: We could try it! I guess we don't need to plot it right now. We're just looking at the determinant…. Uh okay. So we are trying. Yeah. Here is our original -

Ivy: I'm just gonna change the first value to be zero for all of them. See if that does. Yeah. That is how you can do it… Cause technically you're saying v_1 has like zero value…

[Kylie then sets one of the columns of A to be 0 and uses linAlg.det() to check determinant]

Alex: Okay so that one is like on the same plane? …

[Kylie then runs the plotting code with the new matrix that has a determinant of 0]

Kylie: … How does – no. Cause it's all on the same plane, because one of them is zero. What must be true for one of them to not be zero?

Alex: They must be on the same plane

Ivy: So if we have three vectors that are all on the same plane or that they're all, oh wait, like 2d plane…

Alex: Right. Well there's no volume if they're all on the same.

Kylie: Plane. Yeah okay. No, no, okay, got it. Got it. We're good.

Ivy: Which means that they must be linearly dependent!

Alex: … Oh yeah. So we can say they're linearly dependent - if the determinant is zero.

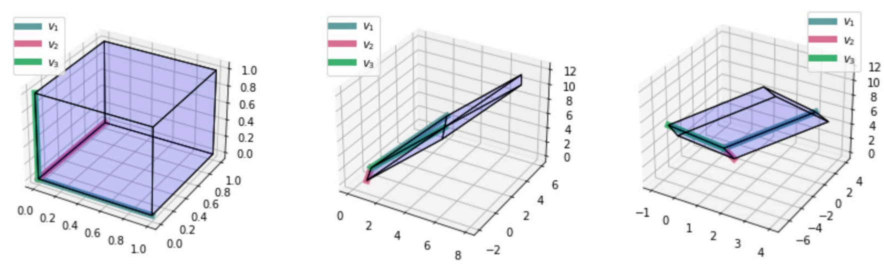

Figure 2. Visualizations Kylie created during the exploration of determinants, trying: (a) unit basis vectors, (b) linearly independent vectors, (c) linearly dependent vectors

The group then goes on to check another set of known linearly independent vectors to verify the determinant is not zero. The three visualizations produced through this experimentation and reflection are shown in Figure 2. The activity does engage the previously discussed prediction and reflection cycle, but also leverages multiple representations as well. This in turn enabled students to discover the relation between the determinant being zero and the linear dependence of vectors, proving opportunities for flexibility and visualization. During the excerpt, the computational representations were the code itself and the actual output of the code, referring both to the visualization of the vectors in 3D as well as the numerical value of the determinant. That is, they were representations that existed via the computer alone. The code itself had students decompose the matrix into a set of vectors in order to calculate the volume to determine the determinant while simultaneously using the full definition of a matrix to calculate the determinant. It was up to the students to determine the test matrices, how the matrix was then used to determine the parallelepiped, and then calculate and visualize the determinant. Besides the scaffolded plotting code, students are enacting all of these concepts, necessitating a switching between resources. Further, when they discussed the notion of determinant, they draw on the algebraic notion alongside the graphical interpretation, leading to the discovery of the impact of linear dependence on the determinant.

Computation enabling prediction and reflection, as previously discussed, means that students are maintaining a constant balance between computational representations. Namely, the code itself and the output, such as Ivy coordinating the input and execution of code with the output value of the determinant. Therefore, since students are already engaging in this process of bridging representations (lines of code and output) then there is a natural way of connecting the tradition of multiple representations, namely students balancing both the graphical, numerical, and algebraic representation. All of these representations were brought into the excerpt, and this facilitated not only flexibility but also the creation of personal visualizations and elaborations.

Discussion and Conclusion

This work highlights the potential that computation has for engaging students in mathematical creativity, specifically when enacted through coding. Within the focal group of students, this was accomplished by computation naturally supporting experimentation through cycles of coding predictions and reflections. Specifically, students adapted their code to follow their own ideas and fluidly check how this altered the result. This cycle of prediction and reflection is not new to linear algebra (Wawro, 2009) nor computation (Lockwood, 2022). However, what this study contributes is specifically how computation enacted through coding can be leveraged to bring about this cycle to promote mathematical creativity. Students naturally explored utilizing the code and engaged in this cycle with and without prompting. These cycles contained multiple dimensions of creativity, especially along originality and risk, which is consistent with research on students' epistemic agency within computational notebooks (Odden et al., 2021). Further, these cycles of prediction and reflection facilitate connections to multiple representations. During the determinant cycle, the group continually bridged and connected their code with the output produced. The modification of the code led to the determination of multiple cases of linear (in)dependence where they then pivoted to make connections between the concepts they had already encountered and the code output. This enabled for a natural bridging between the graphical, algebraic, and numerical understandings of a determinant. A key point within the literature is how students are able to develop great insight and creativity when developing geometric interpretations of key linear algebra concepts (Larson et al., 2008). This is one of the particular strengths of these specific modules but more broadly within computation enacted through coding. Many programming languages have supports built in for visualization which allows students to plot different objects that were previously unattainable by hand. Further, the act of coding coupled with students' control over the representation enable not only originality but also constant maintenance of multiple representations. Within this paper evidence was presented for how computation enacted via coding can lead to opportunities for fostering mathematical creativity through experimentation within computational prediction and reflection cycles, promoting connections to multiple representations. As the relation between computation and mathematical creativity is relatively new, future work should entail focusing on computation as a pedagogical tool for mathematics education in addition to exploring the potentially unique insights that computation offers students.

References

Batiibwe, M. S. K. (2019). Using Cultural Historical Activity Theory to understand how emerging technologies can mediate teaching and learning in a mathematics classroom: a review of literature. *Research and Practice in Technology Enhanced Learning*, 14(1), 1-20.

Blyman, K. K., Arney, K. M., Adams, B., & Hudson, T. A. (2020). Does your course effectively promote creativity? Introducing the mathematical problem solving creativity rubric. *Journal of Humanistic Mathematics*, 10(2), 157-193.

Boaler, J. (2022). Mathematical mindsets: Unleashing students' potential through creative mathematics, inspiring messages and innovative teaching. John Wiley & Sons.

Castle, S. D. (2022, August). If Creativity Return Computing: Exploring the Impact of Computing on Students' Mathematical Creativity in Linear Algebra. In *Proceedings of the 2022 ACM Conference on International Computing Education Research*-Volume 2 (pp. 24-25).

Haylock, D. (1997). Recognizing mathematical creativity in schoolchildren. *ZDM*, 29(3), 68-74.

Henriksen, D., Mishra, P., & Mehta, R. (2015). Novel, effective, whole: Toward a NEW framework for evaluations of creative products. *Journal of Technology and Teacher Education*, 23(3), 455-478.

Isomöttönen, V., Lakanen, A. J., & Nieminen, P. (2020, October). Exploring Creativity Expectation in CS1 Students' View of Programming. In *2020 IEEE Frontiers in Education Conference* (FIE) (pp. 1-8). IEEE.

Jonassen, D. H., & Rohrer-Murphy, L. (1999). Activity theory as a framework for designing constructivist learning environments. *Educational technology research and development*, 47(1), 61-79.

Larson, C., Zandieh, M., & Rasmussen, C. (2008, February). A trip through eigen-land: Where most roads lead to the direction associated with the largest eigenvalue. In Proceedings of the 11th Annual Conference for Research in Undergraduate Mathematics Education.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J. & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, 2(1), 32-37.

Leontiev, A. N. (1974). The problem of activity in psychology. Soviet Psychology, 13(2), 4–33.

Lockwood, E. (2022). Leveraging Prediction and Reflection in a Computational Setting to Enrich Undergraduate Students' Combinatorial Thinking. *Cognition and Instruction*, 1-43.

Odden, T. O. B., Silvia, D., & Malthe-Sørenssen, A. (2021). Using Computational Essays to Redistribute Epistemic Agency in Undergraduate Science. arXiv preprint arXiv:2108.13080.

Savic, M. (2016). Mathematical problem-solving via Wallas' four stages of creativity: Implications for the undergraduate classroom. *The Mathematics Enthusiast*, 13(3), 255-278.

Savic, M., Karakok, G., Tang, G., El Turkey, H., & Naccarato, E. (2017). Formative assessment of creativity in undergraduate mathematics: Using a creativity-in-progress rubric (CPR) on proving. In *Creativity and Giftedness* (pp. 23-46). Springer.

Silver, E. A. (1997). Fostering creativity through instruction rich in mathematical problem solving and problem posing. *ZDM*, 29(3), 75-80.

Silvia, D., O'Shea, B., & Danielak, B. (2019, June). A learner-centered approach to teaching computational modeling, data analysis, and programming. In *International Conference on Computational Science* (pp. 374-388). Springer.

Skuse, B. (2019). The third pillar. Physics World, 32(3), 40.

Sriraman, B., Haavold, P., & Lee, K. (2013). Mathematical creativity and giftedness: a commentary on and review of theory, new operational views, and ways forward. *ZDM*, 45(2), 215-225.
Wawro, M. (2009). Task design: Towards promoting a geometric conceptualization of linear transformation and change of basis. In *Twelfth Conference on Research in Undergraduate Mathematics Education*, Raleigh, NC.
Wiggins, G. P., & McTighe, J. (2005). Understanding by design (2nd ed.). Pearson.