

3-1-2021

## A novel approach to teaching Hidden Markov Models to a diverse undergraduate population

Philip Heller  
*San Jose State University*, [philip.heller@sjsu.edu](mailto:philip.heller@sjsu.edu)

Pratyusha Pogaru  
*San Jose State University*

Follow this and additional works at: [https://scholarworks.sjsu.edu/faculty\\_rsca](https://scholarworks.sjsu.edu/faculty_rsca)



Part of the [Bioinformatics Commons](#), and the [Higher Education Commons](#)

---

### Recommended Citation

Philip Heller and Pratyusha Pogaru. "A novel approach to teaching Hidden Markov Models to a diverse undergraduate population" *Heliyon* (2021). <https://doi.org/10.1016/j.heliyon.2021.e06437>

This Article is brought to you for free and open access by SJSU ScholarWorks. It has been accepted for inclusion in Faculty Research, Scholarly, and Creative Activity by an authorized administrator of SJSU ScholarWorks. For more information, please contact [scholarworks@sjsu.edu](mailto:scholarworks@sjsu.edu).



## Case report

## A novel approach to teaching Hidden Markov Models to a diverse undergraduate population

Philip Heller<sup>\*</sup>, Pratyusha Pogaru

San Jose State University, San Jose, CA 95192, USA

## ARTICLE INFO

## Keywords:

Education  
Bioinformatics  
Hidden Markov Models  
Engagement

## ABSTRACT

Hidden Markov Models (HMMs) are an essential tool for Bioinformatic analysis, with extensive success at finding patterns (e.g. CRISPR arrays or genes of interest) in DNA or protein sequences. HMMs are conceptually intricate, and the algorithms that make use of them are complicated. Thus they present a challenge to Bioinformatics instructors at the undergraduate level, particularly when the students' educational backgrounds are broadly diverse.

At San Jose State University, many undergraduate Bioinformatics students are Biology majors with little or no prior coursework in mathematics, statistics, or programming. For this population a theory-based approach to teaching HMMs would be ineffective. To address this problem we have developed an active learning module that takes advantage of the similarity between HMMs and board games. Our materials include a physical game board for introducing concepts, a software implementation of the game, similar software for visualizing and manipulating HMMs that model proteins, in-class lab exercises, and homework assignments. We have observed high student engagement with these materials over 4 semesters in a diverse undergraduate Advanced Bioinformatics course.

[Here we present our materials, which are freely available to educators.](#)

## 1. Introduction

Undergraduate Bioinformatics education enhances the career paths of Biology and Computer Science majors. Biologists develop programming skills and become familiar with algorithms, while computer scientists are exposed to new problem domains including biomedical research and environmental science. A major challenge for undergraduate instructors is the problem of keeping both constituencies engaged, particularly when teaching complicated algorithms which biologists can be expected to learn more slowly than their more computer-experienced classmates. In this situation biologists may become discouraged and convince themselves that they aren't able to learn mathematical or algorithmic material.

Hidden Markov Models (HMMs) are perhaps the most difficult algorithmic concept of the undergraduate Bioinformatics canon. They are an extension of Markov Chains, which were introduced in 1906 by Andrey Markov [1]. The theoretical basis for HMMs was established in the 1960s, primarily by Baum, e.g. in [2]. They are effective at determining whether a series of observations (e.g. prevailing weather on consecutive days, or chemical units in a DNA or protein sequence) match a pattern represented by an HMM. Their first practical application, in the 1970s,

was in the field of speech recognition [3, 4, 5, 6]. In 1993, Krogh, Haussler, and others proposed using HMMs for classifying genetic sequences [7]. Since then the role of HMMs in Bioinformatics has expanded dramatically, with successful applications to the detection of eukaryotic genes [8, 9], transmembrane proteins [10], chromatin state [11] immune system T-cell receptors [12] and CRISPR-Cas systems [13]. HMMs have also been applied in computer security [14, 15], cryptography [16], neuroscience [17], finance [18], sign-language recognition [19] and many other domains.

Skillful use of Bioinformatic HMMs requires understanding of the underlying concepts. Unfortunately, HMMs are often presented by introducing numerous symbols representing statistical abstractions, and then by presenting algorithms that manipulate the symbols. For example, Bioinformatics textbooks commonly begin their presentation of HMMs with a formal definition, stating that an HMM consists of the following four elements:

- An alphabet of emitted symbols.
- A set of hidden states.
- A matrix of transition probabilities.

<sup>\*</sup> Corresponding author.

E-mail address: [philip.heller@sjsu.edu](mailto:philip.heller@sjsu.edu) (P. Heller).

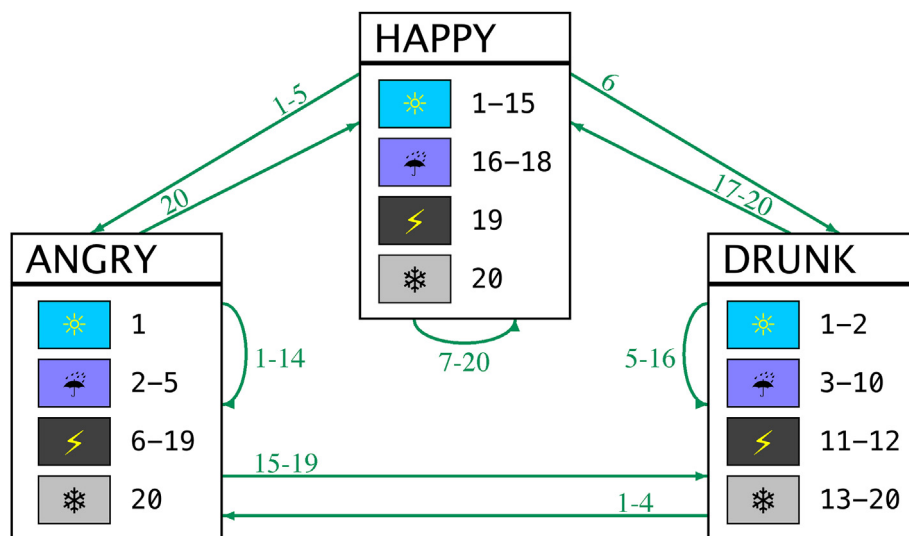


Figure 1. The game board. Number labels on arrows between boxes and on weather icons inside boxes represent transition and emission probabilities respectively, converted to outcomes of rolls of 20-sided dice.

- A matrix of emission probabilities.

Alternatively (for example on the Wikipedia page for Hidden Markov Models [https://en.wikipedia.org/wiki/Hidden\\_Markov\\_model](https://en.wikipedia.org/wiki/Hidden_Markov_model), on May 26, 2020), an HMM is defined as a pair of stochastic processes, of which one is hidden and the other obeys the stationary condition. Neither of these introductions is helpful to a reader who lacks a substantial mathematical and statistical background.

We surveyed 9 Bioinformatics textbooks published from 1997 to 2020 [20, 21, 22, 23, 24, 25, 26, 27, 28]. We found that a theory-first approach to HMMs was used in a third of the sources; theory in context of an example was discussed in another third; and the remainder contained only a brief mention, or no discussion at all, of HMMs.

At San Jose State University, HMMs are presented during the second semester of a 2-semester Bioinformatics course series. The first semester introduces the main public domain tools and databases related to Bioinformatics, along with a few straightforward algorithms; The second semester is more computation-heavy, and students are exposed to more advanced algorithmic topics for the first time. An important learning objective of the second semester is for students to be able to create and apply Profile Hidden Markov Models (an offshoot of HMMs appropriate for modeling proteins) to identify unknown protein sequences. In a typical cohort 20 students are Biology majors and 10 are Computer Science majors. We believe that introducing HMMs by the traditional “theory-first” approach risks alienating the Biology majors [29], whose only mathematics/statistics requirement is a single semester of precalculus, and who may have no programming experience. These students cannot realistically be expected to directly master the abstractions of HMMs. On the contrary, presenting HMMs as concepts beyond their ability can be expected to lower their belief that they can do well, and these lowered beliefs can be expected to lower their learning outcomes [30, 31].

We saw an opportunity to significantly improve learning outcomes compared to a lecture-based approach, using an active learning approach to increase engagement. Academic engagement is crucial to maintaining interest and performance in undergraduate STEM courses, because students have the opportunity to reflect on and process new concepts. The value of active learning in promoting engagement has been well established through numerous large-scale studies [32, 33, 34, 35], including studies of how students learn algorithmic abstraction [29, 36].

In order to present HMMs in a format that encourages participation and promotes learning by a broadly diverse student population, we have developed course materials that present HMMs first as board games, then

as interactive computer visualizations, and lastly as tools for classifying genetic sequences. “Gamification” of algorithmic learning has been shown to expand students’ learning strategies [37] and to evoke positive emotional engagement [38] (in other words, games are fun). Interactive visualization software is an effective tool for learning about algorithms [39]. Our materials include a large portable game board and related accessories, graphical software, homework assignments, and in-class activities. We have used these materials in 4 deliveries of our 2<sup>nd</sup>-semester Bioinformatics course for a mixed population of Biology and Computer Science majors. We have observed high student engagement and strong exam results in all 4 semesters, and a small-scale informal survey indicates positive outcomes. Here we present our approach. All materials will be provided on request to any educator who contacts the corresponding author.

## 2. Materials and methods

### 2.1. Novel course material

We present HMMs in 5 phases: a backstory, a board game, a computer implementation of the game, course material on Profile HMMs, and a computer implementation of Profile HMMs.

#### 2.1.1. The backstory

The backstory is a loose retelling of the Norse mythology of the weather god Thor. In the story, the weather in Scandinavia on any particular day is either sunny, rainy, stormy, or snowing. It is widely believed that the weather is controlled by the moody god Thor, who on any particular day is happy, angry, or drunk. The weather he creates on any day depends on his mood that day, but with an element of chance. For example, when Thor is happy he is likely to create sunny weather, but this is not guaranteed; when Thor is angry he is likely to create thunderstorms; when he is drunk, anything could happen. But no one has ever seen Thor. He might not even exist, and if he does exist nobody can be certain what mood he is in on any day. All we can really know for certain is the weather we observe in Scandinavia from day to day. In other parts of the world, other weather deities are in charge. For example in Armenia and Georgia, the goddess Skadi controls weather. Her moods are happy, angry, and serene. If you’re somewhere between Scandinavia and Georgia – Bulgaria, for example - you might wonder whether Thor or Skadi is controlling the local weather; you would probably form an opinion based on which mythology gave a better explanation of the weather you experience in Bulgaria.

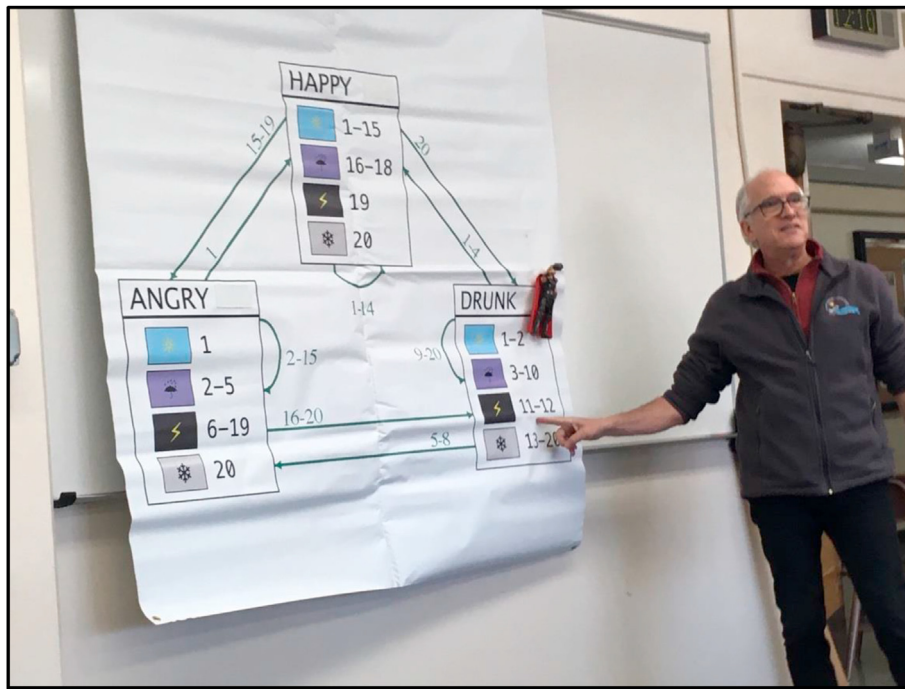


Figure 2. Author Heller using the game board in a class session in March 2020. Thor is drunk; a die roll of 11 or 12 has caused Thor to create a thunderstorm. Photo courtesy of Mike Wu.

### 2.1.2. The game board

The game board is a 3' x 4' vinyl sheet, printed with the image shown in Figure 1. Commercial printing on outdoor-grade vinyl costs less than \$150. Grommets in all 4 corners allow easy hanging in classrooms. A 2" x 4" Velcro™ strip is glued to the center of each of the 3 mood spaces. The game piece is an 8" Thor action figure, with Velcro glued to the back of its cape so it can be attached to any mood space. 2 large (2.5" diameter) 20-sided dice ("d20") are also used, preferably of different colors. Figure 2 shows the board in use in March 2020.

The starting mood (happy, angry, or drunk) is determined by voice vote, and the Thor figure is placed in the corresponding space on the game board. Students are told that they are going to use the game to simulate Scandinavian weather. 3 volunteers - a weather roller, a mood roller, and a scribe – are chosen to help play 10 rounds of the game. A round corresponds to a day, and weather is constant throughout the day. In each round, the weather roller rolls a d20 and reads the weather from the table in the mood space occupied by Thor. The scribe records the mood and the weather. Then the mood roller rolls a d20 to determine

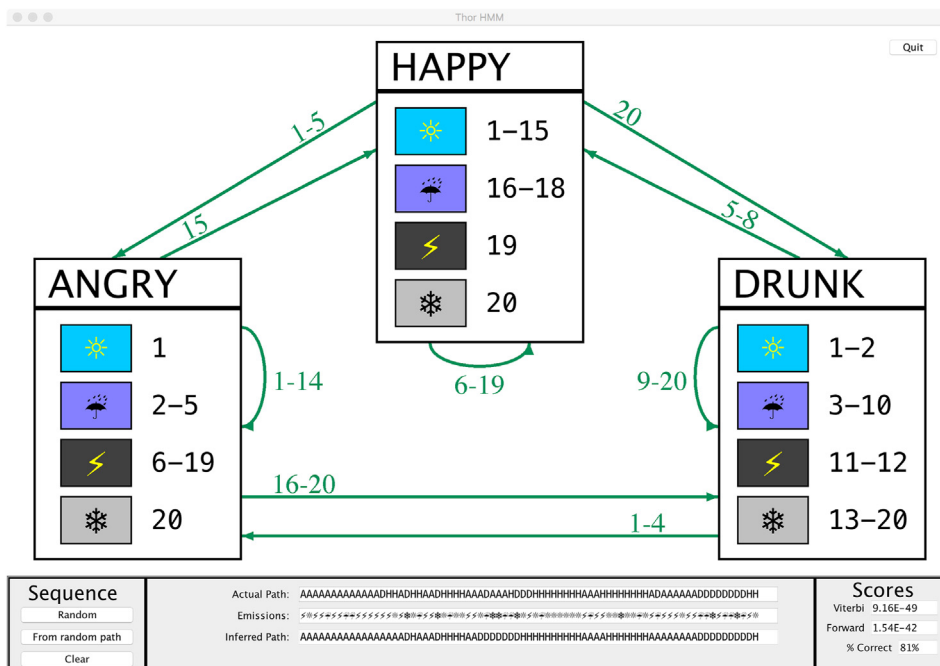
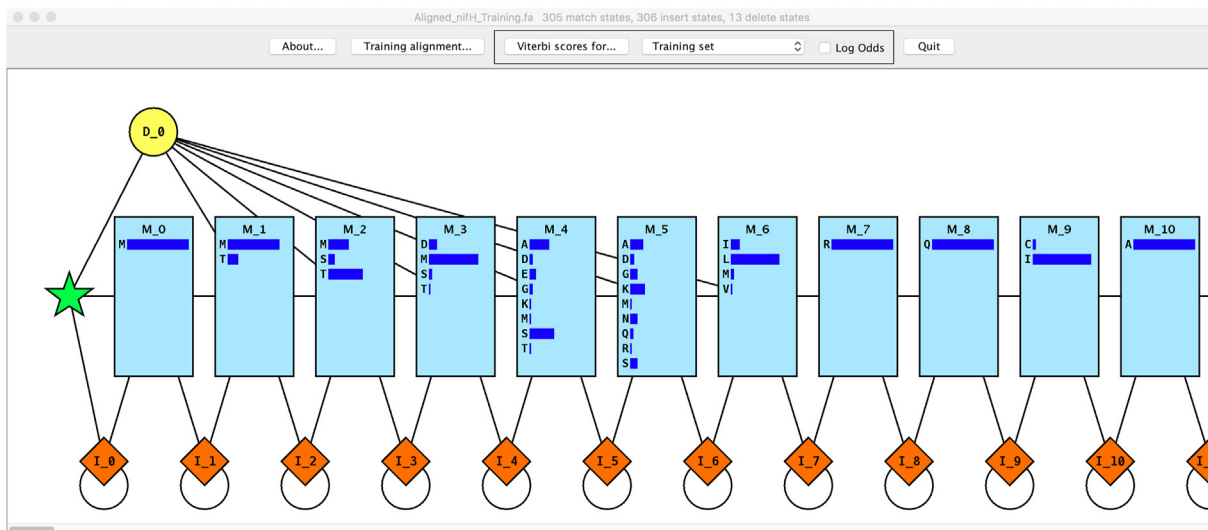


Figure 3. The computer implementation of the game. The user has entered a random sequence of moods ("Actual Path"). The software has simulated a weather sequence from those moods ("Emissions") and then used the Viterbi algorithm to infer the mood sequence given the weather sequence ("Inferred Path").



**Figure 4.** Software visualization of a profile HMM for NifH protein sequences. The software creates and displays the pHMM, and can compute the Viterbi probability of the model generating a putative NifH sequence.

Thor's mood for the next day; arrows pointing to mood spaces are labeled with corresponding die rolls.

After 10 rounds, students are told to discuss the following issues:

- 1) If the game is an accurate representation of Thor, is the resulting weather pattern likely to be typical for Scandinavia?
- 2) If the scribe forgot to record the sequence of moods, could you approximately reconstruct them?
- 3) Given a sequence of weathers, how can you compute the probability that the game produced the sequence?
- 4) If you had a game board for Skadi as well as the one for Thor, and you were given a sequence of weathers, how could you identify which of the 2 game boards probably produced the sequence?

**2.1.3. The computer implementation of the game**

Students are told that the game is an example of a Hidden Markov model. In the vocabulary of HMMs, the moods are hidden states and weather patterns are emissions or observations. They are then shown a software implementation of the board game, as shown in Figure 2. The application, which was written by our research group, supports 2 HMM-related algorithms: generation, and the Viterbi algorithm. Generation is a simulation of the previous exercise: random numbers drive selection of states and emissions. The Viterbi algorithm computes, for any sequence of observations, the most likely sequence of states that generated the observations; this state sequence is called the Viterbi path. The algorithm also computes the probability that the model generates the Viterbi path and the observed sequence.

After a brief demonstration, students download copies of the software and are given hands-on exercises to develop familiarity with the algorithms. The exercises may be done individually or in pairs. The software is written in Java, so it runs identically on any student

laptop whose operating system is Windows, MacOS, or LINUX (see Figure 3).

**2.1.4. Course material on profile HMMs**

The course now transitions from the Thor/weather metaphor to Hidden Markov Models that represent protein sequences (profile HMMs, or pHMMs). This material begins with a detailed presentation of the Viterbi algorithm, which takes about one 75" class session. There follows a presentation of how to create a pHMM representing a protein of interest, given a number of example sequences (the "training set"). This material also takes about one class session.

**2.1.5. Computer implementation of profile HMMs**

Students then do lab exercises in which they download a set of protein sequences from a public database, align the sequences using the public tool Clustal-Omega [40], and create a pHMM from the aligned training set. The pHMM is created and visualized by the same software package used in the previous exercise (Figure 4). Viterbi probabilities are computed for 5 kinds of sequence: members of the training set, instances of the protein that are not in the training set, fictitious instances of the protein simulated by the model, unrelated proteins, and random sequences. In a subsequent homework assignment, students do the same exercise with a different protein; they also use models for different proteins to identify an unknown sequence.

**2.2. Survey**

In the Spring 2019 semester, after all material was presented, all in-class exercises were completed, and all homework exercises were graded and returned, students completed a 5-item questionnaire about their understanding of HMMs. The survey was approved by San Jose

**Table 1.** Survey results.

| Question   | # of students by response |          |         |       |                |
|--|---------------------------|----------|---------|-------|----------------|
|  | Strongly Disagree         | Disagree | Neutral | Agree | Strongly Agree |
| You understood HMMs before they were introduced  | 15                        | 7        | 1       | 4     | 1              |
| You now understand HMMs  | 0                         | 0        | 2       | 16    | 10             |
| The Thor/weather "game board" was an effective way for you to learn about HMMs                   | 1                         | 1        | 1       | 7     | 18             |
| The Thor/weather software and exercises were an effective way for you to learn about HMMs        | 0                         | 0        | 0       | 10    | 18             |
| The Protein Profile HMM software and exercises were an effective way for you to learn about HMMs | 0                         | 2        | 2       | 8     | 16             |

**Table 2.** Correspondence between elements of the backstory and Hidden Markov Model concepts.

| Backstory concept (metaphier)   | HMM concept (metaphrand)  |
|---|---|
| Scandinavia   | A protein, with variation due to evolution  |
| Georgia/Armenia   | A different protein, again with variation   |
| Sunny, rainy, stormy, and snowy weather                               | The 20 amino acids that make up protein sequences   |
| Weather observed over a number of consecutive days                    | Sequences of a protein, determined experimentally   |
| Thor  | An HMM that models the protein  |
| Thor's moods  | The HMM's emission states   |
| We can observe weather  | We can determine the sequence of a protein  |
| ... but we can't know Thor's moods                                    | ... but we don't know what sequence of HMM states produced the sequence                             |
| ... but we can guess Thor's moods                                     | ... but we can use statistical algorithms to identify the most likely states                        |
| ... which helps us predict future weather                             | ... which helps us identify unknown protein sequences   |
| Other weather deities, e.g. Skadi                                     | HMMs for different proteins   |
| Wondering which deity controls the weather in a region, e.g. Armenia  | Computing and comparing the Viterbi probability scores for a protein sequence using various models. |
| Forming an opinion about which deity controls the weather in a region | Choosing the HMM that computes the highest Viterbi probability for a sequence to be identified      |

State University's Institutional Review Board. Response options were “Strongly Disagree”, “Agree”, “Neutral”, “Disagree”, and “Strongly Agree”. Students were asked about their understanding of HMMs before and after the material was presented, and about the effectiveness of the novel approach. Responses are summarized in Table 1. Due to the small sample size, no statistical significance should be inferred.

### 3. Discussion

We have developed instructional materials for teaching Hidden Markov Models in a playful engaging way that avoids theory and algorithms until students are conceptually prepared. Our approach is consistent with extensive research about the value of promoting engagement through “hands-on” active learning activities, particularly among students with little programming or mathematical background.

The backstory gives students a metaphor for thinking about HMMs that does not require a grounding in Statistics or Computer Science. Every element of the story corresponds to a concept related of HMMs (see Table 2). Jaynes [41] explains metaphors as consisting of *metaphrands* (target concepts that need explaining) and *metaphiers* (commonly understood concepts that share attributes with metaphrands). In these terms, the backstory (the metaphor) consists of previously understood metaphiers such as moods and weather, which are useful in explaining the metaphrands embodied in the HMM concepts. The practical benefit of the rich metaphor is that if a student has trouble understanding a technical point, and if an explanation in terms of models, states,

probabilities, etc., is not helpful, then the instructor can explain in terms of weather mythology, which is easier for the student to grasp.

The 4 questions discussed after playing the game pertain to problems that can be solved using HMMs, as shown in Table 3:

The small-scale survey of 18 students (summarized in Table 1), while not statistically significant, suggests that our approach is promising. Most students were initially unfamiliar with HMMs; by the end of the module, all but 2 considered themselves familiar. Nearly all students reported that the game board, software, homework, and labs were effective learning tools.

Our approach required a substantial development effort. The course material consists of 180 PowerPoint™ slides, 2 homework assignments, and 3 in-class lab assignments. The software is approximately 8000 lines of Java source code. We look forward to sharing our materials with educators. Interested parties are invited to contact the corresponding author.

### Declarations

#### Author contribution statement

Philip Heller: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Pratyusha Pogaru: Performed the experiments; Analyzed and interpreted the data; Wrote the paper.

**Table 3.** Correspondence between elements of the game board questions and Hidden Markov Model concepts.

| Game question   | Equivalent question about HMMs   | Relevant HMM algorithm                                       |
|---|--|--|
| If the game is an accurate representation of Thor, is the resulting weather pattern likely to be typical for Scandinavia?   | If an HMM for a protein is accurately trained, are its emissions similar to actual examples of the protein?                      | Sequence simulation  |
| If the scribe forgot to record the sequence of moods, could you approximately reconstruct them?   | Given a protein sequence, can you determine the sequence of states that most probably emitted the sequence?                      | The Viterbi algorithm (traceback path)                       |
| Given a sequence of weathers, how can you compute the probability that the game produced the sequence?  | Given a protein sequence, what is the probability that a protein HMM emitted the sequence  | The Viterbi algorithm (probability)                          |
| If you had a game board for Skadi as well as the one for Thor, and you were given a sequence of weathers, how could you identify which game board probably produced the sequence? | If a sequence were believed to be one of two proteins, and you had an HMM for both proteins, how could you identify the protein? | The Viterbi algorithm (compare probabilities from both HMMs) |



### Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

### Data availability statement

No data was used for the research described in the article.

### Declaration of interests statement

The authors declare no conflict of interest.

### Additional information

No additional information is available for this paper.

### References

- [1] P. Gagnieu, Markov Chains: from Theory to Implementation and Experimentation, John Wiley & Sons, USA, 2017.
- [2] L.E. Baum, T. Petrie, Statistical inference for probabilistic functions of finite state Markov chains, *Ann. Math. Stat.* 37 (6) (1966) 1554–1563.
- [3] J. Baker, The Dragon system - an overview, *IEEE Trans. Acoust. Speech Signal Process.* 23 (1) (Feb. 1975).
- [4] F. Jelinek, L. Bahl, R. Mercer, Design of a linguistic statistical decoder for the recognition of continuous speech, *IEEE Trans. Inf. Theor.* 21 (3) (May 1975) 250–256.
- [5] D. Walker, Speech understanding through syntactic and semantic analysis, *IEEE Trans. Comput. C-25* (4) (Apr. 1976) 432–439.
- [6] M. Gales, S. Young, The application of hidden Markov models in speech recognition, *Foundations Trends® in Signal Processing* 1 (3) (2007) 195–304.
- [7] A. Krogh, M. Brown, I.S. Mian, K. Sjolander, D. Haussler, Hidden Markov models in computational Biology: applications to protein modeling UCSC-CRL-93-32, Aug. 1993, p. 62.
- [8] D. Kulp, D. Haussler, M. Reese, Frank Eckman, A generalized hidden Markov model for the recognition of human genes in DNA, *ISMB-96 Proceedings* (1996) 9.
- [9] W.H. Majoros, M. Pertea, S.L. Salzberg, TigrScan and GlimmerHMM: two open source ab initio eukaryotic gene-finders, *Bioinformatics* 20 (16) (Nov. 2004) 2878–2879.
- [10] E.L.L. Sonnhammer, G. von Heijne, A. Krogh, A hidden Markov model for predicting transmembrane helices in protein sequences, *ISMB-98, Proceedings* 8 (1998).
- [11] J. Ernst, M. Kellis, ChromHMM: automating chromatin-state discovery and characterization, *Nat. Methods* 9 (215) (Feb. 2012).
- [12] V.I. Nazarov, et al., tcR: an R package for T cell receptor repertoire advanced data analysis, *BMC Bioinf.* 16 (1) (Dec. 2015).
- [13] S.S. Abby, B. Néron, H. Ménager, M. Touchon, E.P.C. Rocha, “ MacSyFinder, A program to mine genomes for molecular systems with an application to CRISPR-cas systems, *PLoS One* 9 (10) (Oct. 2014), e110726.
- [14] W. Wong, M. Stamp, Hunting for metamorphic engines, *J. Comput. Virol.* 2 (3) (Nov. 2006) 211–229.
- [15] A. Raghavan, F. Di Troia, M. Stamp, Hidden Markov models with random restarts versus boosting for malware detection, *J. Computer Virol. Hacking Tech.* 15 (2) (Jun. 2019) 97–107.
- [16] R. Vobbilisetty, F. Di Troia, R.M. Low, C.A. Visaggio, M. Stamp, Classic cryptanalysis using hidden Markov models, *Cryptologia* 41 (1) (Jan. 2017) 1–28.
- [17] G. Radons, Analysis, classification, and coding of multielectrode spike trains with hidden Markov models, *Biol. Cybernet.* 71 (1997).
- [18] A. Rossi, G. Gallo, Volatility estimation via hidden Markov models, *J. Empir. Finance* 13 (2) (Mar. 2006).
- [19] C. Vogler, D. Metaxas, “Adapting hidden Markov models for ASL recognition by using three-dimensional computer vision methods,” in 1997, in: *IEEE International Conference on Systems, Man, and Cybernetics*, 1, Computational Cybernetics and Simulation, Orlando, FL, USA, 1997, pp. 156–161.
- [20] D. Gusfield, Algorithms on Strings, Trees, and Sequences - Computer Science and Computational Biology, first ed., Cambridge University Press, 1997.
- [21] R. Durbin, S. Eddy, A. Krogh, G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Amino Acids*, second ed., Cambridge University Press, 1998.
- [22] R. Deonier, S. Tavare, M. Waterman, *Computational Genome Analysis - an Introduction*, first ed., Springer, 2005.
- [23] M. Zvelibil, J. Baum, *Understanding Bioinformatics*, first ed., Garland Science, 2008.
- [24] J. Pevsner, *Bioinformatics and Functional Genomics*, second ed., John Wiley & Sons, 2009.
- [25] C. St. Clair, J. Visick, *Exploring Bioinformatics - A Project-Based Approach*, first ed., Jones & Bartlett Learning, 2015.
- [26] J. Momand, A. McCurdy, *Concepts in Bioinformatics and Genomics*, first ed., Oxford University Press, 2017.
- [27] P. Compeau, P. Pevzner, *Bioinformatics Algorithms - an Active Learning Approach*, third ed., 2018.
- [28] M. Mills, N. Barban, Felix Tropsch, *An Introduction to Statistical Genetic Data Analysis*, first ed., The MIT Press, 2020.
- [29] D. Statter, M. Armoni, Teaching abstraction in computer science to 7th grade students, *J. Educ. Resour. Comput.* 20 (1) (Feb. 2020) 1–37.
- [30] A. Wigfield, J.S. Eccles, Expectancy-value theory of achievement motivation, *Contemp. Educ. Psychol.* 25 (1) (Jan. 2000) 68–81.
- [31] J.A. Fredricks, T. Hofkens, M.-T. Wang, E. Mortenson, P. Scott, “Supporting girls’ and boys’ engagement in math and science learning: a mixed methods study: supporting girls’ and BOYS’ engagement IN math and science, *J. Res. Sci. Teach.* 55 (2) (Feb. 2018) 271–298.
- [32] J.A. Gasiewski, M.K. Eagan, G.A. Garcia, S. Hurtado, M.J. Chang, From gatekeeping to engagement: a multicontextual, mixed method study of student academic engagement in introductory STEM courses, *Res. High. Educ.* 53 (2) (Mar. 2012) 229–261.
- [33] L. Blasco-Arcas, I. Buil, B. Hernández-Ortega, F.J. Sese, Using clickers in class. The role of interactivity, active collaborative learning and engagement in learning performance, *Comput. Educ.* 62 (Mar. 2013) 102–110.
- [34] S. Freeman, et al., Active learning increases student performance in science, engineering, and mathematics, *Proc. Natl. Acad. Sci. Unit. States Am.* 111 (23) (Jun. 2014) 8410–8415.
- [35] R. Christensen, G. Knezek, T. Tyler-Wood, Alignment of hands-on STEM engagement activities with positive STEM dispositions in secondary school students, *J. Sci. Educ. Technol.* 24 (6) (Dec. 2015) 898–909.
- [36] J.D. Serman, Learning in and about complex systems, *Syst. Dynam. Rev.* 10 (2–3) (1994) 291–330.
- [37] M.-B. Ibanez, A. Di-Serio, C. Delgado-Kloos, Gamification for engaging computer science students in learning activities: a case study, *IEEE Trans. Learning Technol.* 7 (3) (Jul. 2014) 291–301.
- [38] S. Volet, C. Seghezzi, S. Ritchie, Positive emotions in student-led collaborative science activities: relating types and sources of emotions to engagement in learning, *Stud. High Educ.* 44 (10) (Oct. 2019) 1734–1746.
- [39] D. Schweitzer, W. Brown, Interactive visualization for the active learning classroom, in: *Proceedings of the 38th SIGCSE Technical Symposium on Computer Science Education*, Mar. 2007, pp. 208–212.
- [40] F. Sievers, et al., Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega, *Mol. Syst. Biol.* 7 (1) (Jan. 2011) 539.
- [41] J. Jaynes, *The Origin of Consciousness in the Breakdown of the Bicameral Mind*, Houghton Mifflin, 1976.