

Neuroforest : Machine Learning methods for trajectory analysis and correlation to ADHD behaviour in a simulated environment

Tom Fizycki, Benjamin Kupecek
tom.fizycki@ensta-paris.fr, benjamin.kupecek@ensta-paris.fr

Abstract

The exploration-exploitation dilemma, a central problem in decision-making, involves balancing the pursuit of novel opportunities against optimizing known rewards. This challenge is particularly relevant in reinforcement learning and in understanding human behavior, including psychiatric conditions like Attention-Deficit/Hyperactivity Disorder (ADHD). ADHD offers a unique perspective on exploration strategies, as individuals may exhibit heightened exploratory behaviors due to impaired reward processing and executive control deficits. Using the Neuroforest dataset, which captures player trajectories in a virtual environment alongside their ADHD assessment score (ASRS), we investigate behavioral markers and computational models to analyze exploration patterns.

Our methodology combines explicit feature extraction and black-box latent spaces for trajectory processing. Based on existing techniques used on trajectory processing, we aim to analyze and distinguish patterns or features that could predict ASRS scores. Overall, we would like to quantify and understand the link between ADHD and the exploration/exploitation dilemma, with a multi-faceted approach.

1. Introduction

1.1. Background on ADHD and the exploration-exploitation dilemma

The exploration-exploitation dilemma is a fundamental decision-making problem in which agents must balance exploring novel options and exploiting known rewards. This dynamic trade-off is modeled extensively in reinforcement learning (RL), where policies dictate whether to prioritize information gathering or resource optimization. In humans, the neural and cognitive mechanisms underlying this trade-off are complex, influenced by both individual differences and psychiatric conditions such as Attention-Deficit/Hyperactivity Disorder (ADHD) [6].

ADHD behavior provides a unique lens to examine ex-

ploration strategies. Individuals with ADHD often exhibit heightened exploratory behaviors, potentially driven by impaired reward processing and reduced executive control [5, 13]. For example, impulsivity—an established characteristic of ADHD—may result in excessive exploration, disrupting optimal decision-making [17]. Computational models in neuroscience reveal that dopaminergic signaling plays a pivotal role in modulating exploration-exploitation dynamics, with altered dopamine levels in ADHD contributing to maladaptive exploration patterns [9].

By incorporating both behavioral markers and RL-inspired computational modeling, this field bridges cognitive neuroscience and psychiatry. These insights not only enhance our understanding of ADHD's neurocognitive basis but also inform the design of interventions tailored to modulate exploration strategies in affected individuals.

1.2. The Neuroforest Dataset

The dataset is divided in two parts, one made in 2022 and one in 2024, with the same experimental protocol. Each player is shown the game, with the instruction to gather as many mushrooms as possible in 20 minutes.



Figure 1. Frame of the neuroforest video game : on the ground, flowers and two mushrooms are visible, and can be caught by user input. On the bottom left, the remaining time. On the bottom right, the number of mushrooms gathered so far.

The player first plays an initial game, with the objective to get used to the game, which will not be used for further

analysis due to bias. Then, he/she plays twice the game, on a "uniform" and on a "patchy" mushroom distribution. Meanwhile, we collect the coordinates on a 2D map of the player's trajectory, sampled with 25 frames per second. The resulting array constitutes the raw data for the player. Each player also answers an ASRS questionnaire (Adult ADHD Self-Report Scale), which is a tool used by psychiatrists to get insights on various ADHD dimensions. Those data are also part of the dataset as a CSV file, reporting each player's answer.

The dataset is composed of two experiment phases : 39 subjects in 2022 and 19 subjects in 2024. The latter were conducted in a similar fashion and context, under the supervision of psychiatrists to ensure data consistency and minimize bias. This results in a total of 116 trajectories, that is 58 across each environment. The influence of the environment over trajectories and behaviours will be discussed in [section 3](#) and [section 6](#). Those trajectories are saved as JSON files with 30,000 points of 2D positions. The dataset also gathers the position of all mushrooms, as well as all those inside the players' current field of vision, as well as the amount of gathered mushrooms.

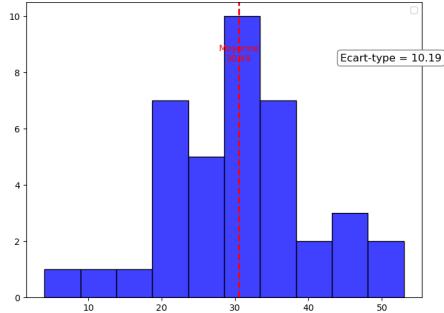


Figure 2. Distribution of ASRS scores among the 2022 dataset

[Figure 2](#) displays the distribution of ASRS scores over test subjects. [Figure 3](#) shows examples of trajectories of a given individual in the two different environments.

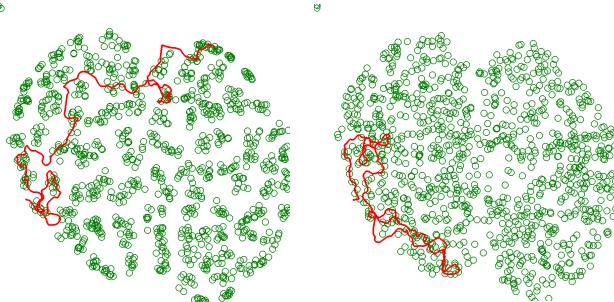


Figure 3. Two trajectories of the same subject. On the left, the patchy environment. On the right, the uniform.



Figure 4. Heatmaps of the two environments. On the left, the patchy environment. On the right, the uniform.

2. Related work

The analysis of behavioral trajectories in constrained environments, particularly in the context of psychiatric conditions like ADHD, is an emerging field with limited publicly available datasets. Most existing studies rely on small-scale datasets due to the challenges associated with data collection, such as the need for controlled environments and participant compliance. It is often applied to studying the wildlife, as the cost and effort of tracking wild animals limits the size of collectible data.

The Neuroforest dataset, with its 58 subjects across two environments, aligns with small-scale studies. Its uniqueness lies in the combination of trajectory data with ADHD-specific metrics (ASRS scores), offering a rare opportunity to study links between movement patterns with psychiatric symptoms.

The DeepHL [10] framework (Deep Learning-assisted Comparative Analysis of Animal Trajectories) provides an innovating approach to trajectory analysis for small datasets. It employs a multi-scale attention mechanism within a neural network to identify and highlight segments of trajectories that are most predictive of specific behaviors or conditions. Its applicability to small datasets, its use of handcrafted and learned features in analysis, as well as the idea of Attention-Based Feature Extraction were key contributions.

In addition, the WildGen [1] and later WildGraph [2] frameworks were the key inspiration for the generation of synthetic trajectories and the analysis of the spacial relations (player position, mushroom density...). Both approaches demonstrate how hybrid methods (deep learning + manual features) can extract meaningful behavioral patterns from minimal data. We adapt this by:

- Using synthetic trajectory generation (WildGen-inspired) to compensate for data scarcity
- Extracting geometric features (curvature, time near bor-

ders) as potential ADHD markers

This synergy between computational ecology and cognitive neuroscience offers novel ways to study psychiatric disorders through virtual movement analysis.

3. Methodology

3.1. Preprocessing

Since Neuroforest is a very new dataset, and is yet to be studied in more detail and to undergo a handful of SOTA techniques on similar tasks, this project sets up its own preprocessing framework.

To cope with the small number of trajectories for training and evaluation for classical Machine Learning algorithms, a first set of hypotheses was established. Its limitations will be discussed in [section 6](#).

1. We first assume that the features encoding ADHD behaviour present in the trajectories are environment-independent on a global scale.
2. Then, we assume that the location of ADHD-correlated behaviour is, in distribution, uniform over time.
3. We assume the measure of ASRS scores follows the results presented in [\[15\]](#).
4. There exists a minimal time interval Δt under which no relevant features can be extracted.

In order to cope with the previously discussed limited data volume, we apply the following augmentation process. Let $s, N_\theta \in \mathbb{N}$, and $T = (x_t, y_t)_{t \in [0, t_{max}]}$ be a given trajectory. For each trajectory, we :

- We select points every Δt ms, the latter being formally introduced in [\(4\)](#). In practice, this value is chosen differently when computing manual feature extraction (angular acceleration for instance), rather than when computing data with an end-to-end model for precision-complexity tradeoffs.
- Cut T into s non-overlapping sections $(T_j)_{j \leq s}$, where, if $T_s = t_{max}/s$, then the section T_j corresponds to $T_j = ((x_{jT_s}, y_{jT_s}), \dots, (x_{jT_s + (T_s - 1)}, y_{jT_s + (T_s - 1)}))$. Here, s represents a "scale" parameter, which rules the length T_s of each section. The hypothesis developed in [\(2\)](#) ensures that we retain the significant characteristics of ADHD present in the trajectory.
- For each section (T_j) , we create N_θ trajectories that are obtained with a rotation of $2\pi/N_\theta$. This is justified by the isotropy coming from hypothesis [\(1\)](#). In addition, we center the obtained trajectories by setting the starting point to $(0, 0)$. This also comes from the same hypothesis made earlier.
- For each segment, we sample a under $\mathcal{G}(\mu, \sigma_i^2)$, where μ is the ASRS score obtained by the patient, and σ is the intraclass standard deviation obtained when mixing ICC scores from [\[15\]](#) and observed sample variances σ_e^2 ,

with the following formula: $\sigma_i^2 = \sigma_e^2 * \frac{1}{ICC} - 1$. This is justified by the variability of test-retest ASRS developed in hypothesis [\(3\)](#).

In practice, different values of the preprocessing parameters are used in the experiments, depending on how realistic the assumptions are in the framework of each.

3.2. Trajectory analysis

To analyze the ADHD-related patterns in player trajectories, we employ a combination of manual feature extraction and deep learning-based comparative methods inspired by DeepHL. This dual approach hopefully allows us to identify both known features of remarkable ASRS behavior and potential new insights from the data.

3.2.1. Primitive Feature Extraction

We start by converting the raw trajectory data into time series of primitive features, including speed, angular acceleration, and distance traveled from the starting point. These primitive features are complemented by handcrafted features, on which regression and simple prediction techniques are performed in order to try to find correlations and to then predict an ASRS score based on those large-scale features.

Also, we noticed that most player's trajectories were biased by the map they played the game on. Namely, we could witness a lot of player following the edges of the map, or going through certain obstacle, which can be used as other features to discriminate between different behaviors. To this end, we choose :

- Mathematical attributes of the trajectory presented in [\[10\]](#). The latter will be developed later on in [section 4](#).
- The time spent on the right side of the map. Considering this part was less explored throughout the different subjects and environments (see the heatmap in [Figure 4](#)), we expect individuals showing higher exploration tendencies to be more likely to cross the middle, as well as potentially spending more time there.
- Distance to the border of the map's convex hull, in an attempt to mimic the limits of the environment. The assumption behind this is that players behaving with a more rational thought process would be tempted to follow this line, exploiting this zone rather than exploring zones in the middle of the map with a higher number of mushrooms in the field of vision of the player. This seems confirmed by [Figure 4](#), seeing as the borders are highly visited.
- Number of mushrooms collected throughout the game.

To this end, we target 2 indicators of ADHD behaviour : sum of points over the questionnaire, as well as the number of answers exceeding the question-specific thresholds defined in the ASRS. Some variants are used for reliability of comparisons. For more information, the questionnaire is

referred in appendix.

3.2.2. Attention-Based Deep Learning Analysis

Following the principles of DeepHL [10], we will try to implement an interpretable neural network with a multi-scale layer-wise attention mechanism to highlight key segments of the trajectories associated with ADHD behaviors. The workflow is described as follows:

1. Preprocessing: Trajectories are normalized for position and rotation invariance. Primitive features and hand-crafted features are fed as input to the network alongside raw coordinates.
2. Network Training: The model is trained on a binary classification task to distinguish trajectories of players with high vs. low ASRS scores. The attention mechanism identifies trajectory segments that are most predictive of ADHD-related behavior. The classification task is used to make the work easier for the network, relative to the low amount of data.
3. Visualization and Interpretation: Attention maps generated by the network are used to color trajectory segments, emphasizing areas of significant divergence in movement patterns. Handcrafted features are correlated with these attention values to elucidate their role in the classification.

This technique and its dedicated architecture allow us to extract trajectory parts that are strongly correlated with a high or low ASRS score.

3.3. Multiplayer simulated environment

In a second step, we try a game theoretic-based approach, taking a new look at our problem.

The first one is to compare the distribution of, that is the number of mushrooms gathered in the Neuroforest game over a given time, amongst the population, and depending on their ASRS questionnaire's answers. Then, we want to simulate the compare the gains in a simulation in which the subject does not know a priori how many players have played before him, and thus gathered mushrooms he will not be able to. Though this observation was not itself conducted in the experiments, we can make the following hypothesis. Considering a given time t , we only study the following parameters : the mushrooms that are close to the subject's position, the trajectory and the associated rewards before time t , and the ASRS score. We thus compute, for a given trajectory, the neighbouring surface with a threshold d , which we call *gathering hull*, representing the area that a player would reasonably cover or reach in a very short time around their position.

Then, given the latter, we re-normalize the quantity of mushrooms in a given area, according to the number of gathering hulls overlapping with mushroom positions. The normalization methods will be developed in section 4. The

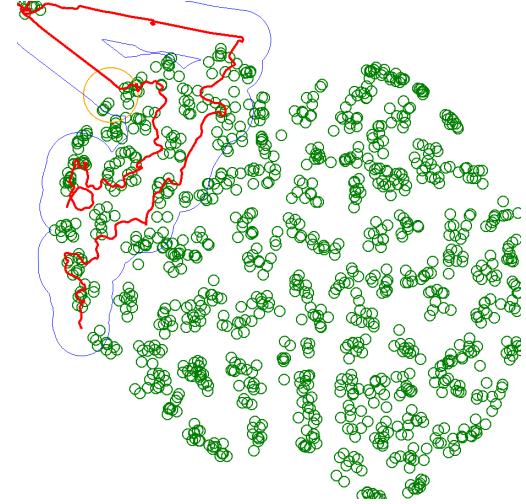


Figure 5. Gathering hull of a given trajectory ($d = 50$). Only the outline of the surface is shown in blue. In orange, a circle of radius d . Here, the point in orange is shown as reachable as it is within distance d of the trajectory.

resulting figure, which we call the *multiplayer simulated reward* (MSR), represents the expected reward the subject can have, not knowing how many persons have played before him. In reality, this hypothesis is not completely true, because subjects may have realized that they play in a single-person environment. Nonetheless, this can strongly be influenced by ADHD, so this remains relevant to us, even under an altered hypothesis. We then wish to compare the MSR over ASRS scores.

3.4. Trajectory Generation

Some research was made [1] on small datasets of trajectories, which often happens when studying wildlife. In this work, we employ a Variational Autoencoder (VAE) to generate synthetic trajectories from a limited dataset of observed trajectories. The VAE is trained to learn a low-dimensional latent representation of the input trajectories, capturing their essential spatial and temporal characteristics. Once trained, the decoder component of the VAE is used to reconstruct trajectories, while the latent space allows for the generation of new, synthetic trajectories by sampling from a learned distribution. This approach is particularly useful for small datasets, where the limited number of observed trajectories may hinder the performance of data-driven models. By generating realistic synthetic trajectories, we augment the dataset, enabling more robust training of downstream models, such as classifiers.

We want to ensure that the generated trajectories keep in some way the primitive features we extract, as we consider

that they are meaningful in the description of a trajectory.

In a second step, we generate trajectories using HMMs, usually used for modeling and generating sequences of observations, such as trajectories. Each state in the HMM represents a distinct phase or behavior of the system, and the transitions between these states are governed by a set of probabilities. The observations, which correspond to the trajectory data, are generated based on the current state according to an emission probability distribution [11].

The process of trajectory generation using HMMs involves several key steps. First, the model parameters, including the initial state probabilities, transition probabilities, and emission probabilities, are estimated from the observed trajectory data. This is typically done using algorithms such as the Baum-Welch algorithm, which is an expectation-maximization (EM) technique [3]. Once the model is trained, new trajectories can be generated by sampling from the HMM. This involves starting from an initial state, sampling subsequent states based on the transition probabilities, and generating observations based on the emission probabilities of the current state.

4. Implementation and experiments

4.1. Manual feature extraction

As a first approach, we had the idea that we could guess some trajectory features that would be correlated to some ASRS values among our subjects. One could, for instance, assume that a high average speed would be correlated to a high bias toward exploration that could have an incidence on ASRS. We also chose to pick different features of higher order, in order to detect more fine correlations.

We therefore extracted the following features :

1. speed, defined as :

$$\mathbf{v}_t = \frac{\mathbf{x}_{t+1} - \mathbf{x}_t}{dt}$$

2. acceleration, defined as :

$$\mathbf{a}_t = \frac{\mathbf{t} - \mathbf{v}_{t-1}}{dt^2}$$

3. angle, defined as

$$\alpha_t = \arccos \left(\frac{(\mathbf{x}_t - \mathbf{x}_{t-1}) \cdot (\mathbf{x}_{t+1} - \mathbf{x}_t)}{\|\mathbf{x}_t - \mathbf{x}_{t-1}\| \|\mathbf{x}_{t+1} - \mathbf{x}_t\|} \right)$$

4. angular speed, defined as :

$$\beta_t = \frac{\alpha_{t+1} - \alpha_t}{dt}$$

5. angular acceleration, defined as :

$$\gamma_t = \frac{\beta_{t+1} - \beta_t}{dt}$$

6. curvature, defined as :

$$\kappa_t = \frac{\Delta x_t \Delta y_{t+1} - \Delta x_{t+1} \Delta y_t}{\sqrt{(\Delta x_t^2 + \Delta y_t^2) \cdot (\Delta x_{t+1}^2 + \Delta y_{t+1}^2)}}$$

with $\Delta x_t = x_t - x_{t-1}$ and $\Delta y_t = y_t - y_{t-1}$

Once those features are extracted, they are reduced because of the high length of the time array, to lower the amount of data. We also calculate the variance and mean of those distributions, in order to try to correlate them with ASRS scores.

4.2. Deep Learning based feature extraction

We present two distinct neural network architectures designed to predict scores based on 2D trajectory data. These models leverage temporal and spatial patterns inherent in sequential input, ensuring robust feature extraction and accurate predictions.

1. LSTM with Attention: The first model employs a Long Short-Term Memory (LSTM) network augmented with an attention mechanism. The LSTM component captures temporal dependencies in the input sequences, while the attention mechanism highlights critical time steps that contribute significantly to the prediction task. Specifically, the architecture comprises a 4-layer LSTM with a hidden dimension of 64, processing 2D input features (e.g., x and z coordinates). The output of the LSTM is passed through a single-head multi-head attention module, emphasizing salient temporal regions. The attention-weighted features are then mapped to a scalar output via two fully connected layers, followed by a sigmoid activation function for score prediction. Dropout regularization with a rate of 0.5 is applied to mitigate overfitting.

2. CNN-Based Encoder: The second model adopts a convolutional architecture tailored for extracting spatial features from sequential trajectory data. This architecture processes 2D input channels (x and z) through three convolutional layers with progressively increasing channel dimensions (8, 16, and 32). Each convolutional layer incorporates batch normalization, max-pooling, and Leaky ReLU activation to enhance feature representation and prevent vanishing gradients. The sequence length is dynamically reduced through the pooling operations, culminating in a feature map that is processed by two fully connected layers. The final output is a sigmoid-activated prediction that represents the score. The model initializes the weights using Kaiming normalization to promote convergence and employs dropout regularization (rate = 0.5) for improved generalization.

4.3. Multiplayer Simulated Rewards

In computing the rewards mentioned in subsection 3.3, we apply the following process. Let c_i be a mushroom, and $N(c_i)$ the number of trajectories getting in range of d of c_i . Let p_{gather} be the probability of a single player taking a single mushroom in his gathering hull.

Let us first assume that p_{gather} is independent of other players. Under these conditions, the odds of c_i having been gathered is $N(c_i) \times p_{gather}$. We hence define $MSR(c_i) = 1/N(c_i)$.

In a second hypothesis, we want to model the interaction between players by setting the player to be aware of previous players' going through a point before him/her. We then model the loss awareness present in human behaviour through various game-theoretic settings as shown in [8, 16], expecting a player to be exponentially more likely to pickup an item that was overlooked or not seen by previous players. Hence, we define $MSR_{exp} = \exp(-N(c_i))$. This reward model is expected to strongly favor going through unexplored zones.

The MSR is parametrized by k the number of mushrooms we consider to compute this figure. Here, we only consider the mushrooms of highest value, in order to get rid of bias due to the start of the trajectory and potential converging points in the environment.

4.4. Trajectory Generation

Given the methodology developed in [1], we develop a set of three experiments. After training, we first sample points in the latent space to generate trajectories, and compare the distribution of some of the previously handcrafted features as benchmarks to evaluate the quality of the decoding process. We will, however, limit ourselves to the trajectory diameter, the time ratio spent on the right side of the island, and the average distance to the hull, because other features (that derive from the movement) are too unstable and will depend a lot on the smoothing filter, which we discuss later.

Secondly, we generate the reconstruction of known trajectories, in order to evaluate through the features how the signal is conserved through the latent space's representation.

To do that, we built an autoencoder whose hyperparameters were chosen based on compromises between the number of parameters and the quality of the reconstruction, both visually and through the MSE loss. We chose a 32 dimension encoder/decoder, with a 3 dimensions latent space, as we observed that higher encoding dimensions tended to lower the loss, but higher latent space dimension tended to make reconstructions more noisy.

We finally fit a gaussian mixture model on the latent space, in order to perform later sampling to generate trajectories on this space.

Looking at the HMM-based trajectory generation, we implement the following experiment. After training a kNN to define environment states, fitting the HMM, we evaluate coverage and exploration on grids of multiple resolutions. The latter are defined as : $\text{cov}_{(t_j), T_i} = \frac{N_{overlap}}{N_{real}}$, $\text{expl}_{(t_j), T_i} = 1 - \frac{N_{overlap}}{N_{gen}}$ where N_{real} is the number of cells of the grid that have been covered by the real trajectories, N_{gen} by the generated ones, and $N_{overlap}$ by both. section 5 also displays a qualitative analysis of generated trajectories.

5. Results

5.1. Manual feature extraction - statistical methods

As stated in subsection 4.1, the naive goal was to guess some correlations between those features and the actual scores. The following correlation matrix gives the following results : as for linear correlation, the strongest link we can establish is between mean speed (i.e total distance) and score, with a correlation coefficient $R^2 = 0.33$.

Variables	Patchy		Uniform	
	Corr	p-value	Corr	p-value
Average speed	0.339	0.035	0.089	0.598
Time on right	0.319	0.047	-0.044	0.793
Avg angle speed	0.293	0.070	-0.337	0.040

Table 1. Correlation and p-values with number of answers over question-specific threshold for variables in patchy and uniform environments.

Table 1 shows the p-values of average speed (v_t), average angular speed (β_t), and the ratio of time spent on the right of the map, using Spearman tests with target ADHD values. In addition, Figure 6 show all the correlations of handcrafted features with targets.

In order to account for potential non-linear relations and interactions between those features that would not be detected by a linear regressor, we used a Random Forest Regressor to predict the score. Even though the amount of data was low, we manage to confirm the idea that speed was really the predominant features that could be looked at on the whole trajectory in order to predict ASRS. We compute the feature importance in Figure 7.

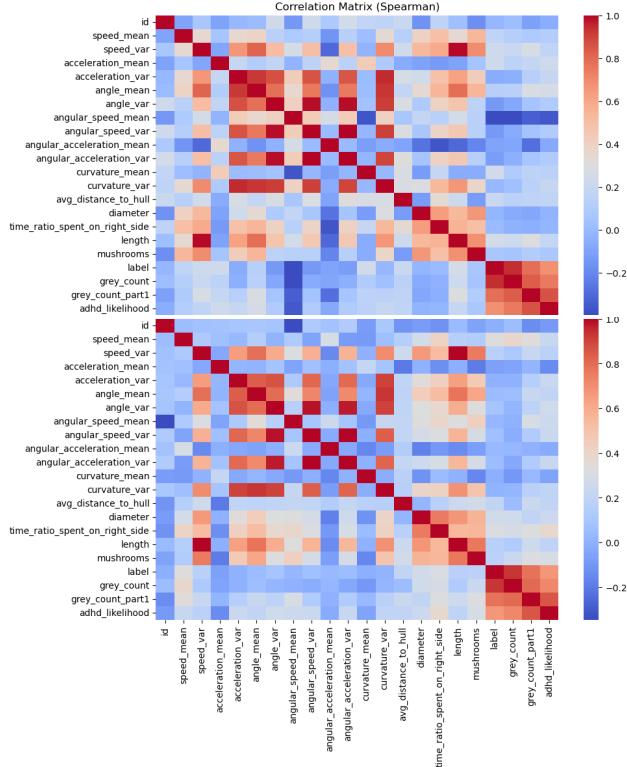


Figure 6. Correlation matrix of handcrafted features in the patch (on top) and uniform (below) environments with a Spearmann test.

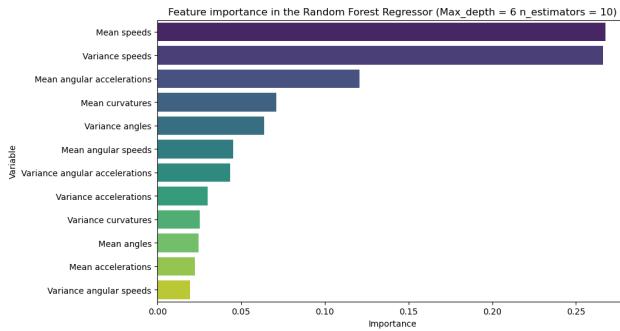


Figure 7. Feature importance with a Random Forest Regressor

5.2. Deep Learning-based extraction

Figure 8 highlights the best predictions that could be achieved using such methods. The distributions of ASRS scores are overall not well conveyed, this throughout different targets, entropy incentives, Kullback-Leibler divergences during the training process. Facing through those limitations, no matter the techniques used to pierce through them, we will be discussing method capabilites later on in section 6.

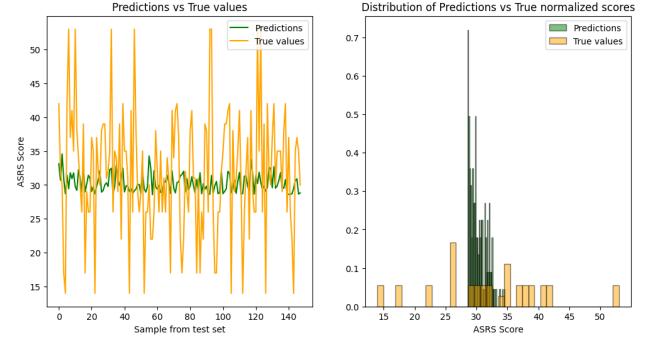


Figure 8. Prediction vs Ground truth using a CNN-based feature extractor as described in section 4. On the left, plot of predictions through the test set. On the right in distribution.

5.3. Multiplayer-simulated environment

Figure 9 shows the correlation scores and p-values of the Spearmann tests computed between ASRS and MSR. Most notable results include : a correlation of $\rho = -0.37$ and $p = 0.022$ for $d = 100$ and $k = 10$ in uniform scenarios, outperforming all handcrafted and mathematical so far in terms of representation. We notice that there is a strong difference between patchy and uniform environments in terms of correlations (also found in Table 2 and Table 3), which will be discussed later on. In addition, MSR_{exp} makes a minimal difference once compared to MSR .

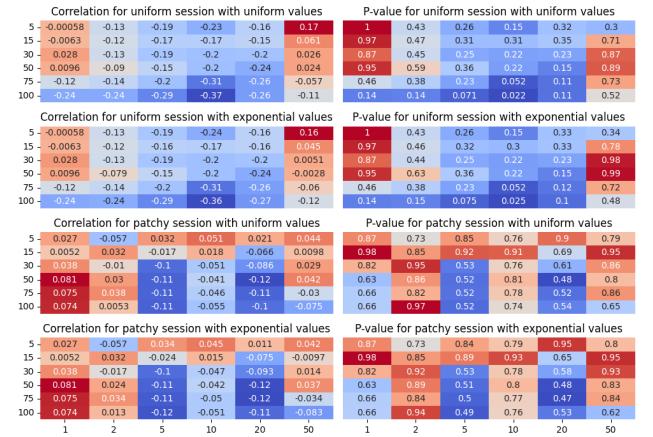


Figure 9. Correlation scores and p-values of the Spearmann tests computed between ASRS scores and MSR. On the horizontal axis, the number k of top-mushrooms that are used for calculation of the MSR. On the vertical axis, the distance d that is used to calculate the hull.

In Table 2 and Table 3, we show the correlation scores and p-values of the same Spearmann test, but update mushroom values based only on their binary gathering by players, and without any notion of hull surrounding the player's trajectory. Upon the notion of d disappearing, we notice

only two significant features, with a best of $\rho = 0.330$ and $p = 0.040$, underperforming compared to Figure 9. The exponential normalization seems to hurt relevance, which could be expected considering that the hull introduces spatial smoothing, and considering only gathered mushrooms when the value normalization may solely highlight pickup biases due to effects in the field of vision.

Threshold	Uniform		Patchy	
	Corr	p-val	Corr	p-val
1	0.159	0.334	0.206	0.214
2	0.303	0.060	0.086	0.608
5	0.256	0.115	-0.020	0.907
10	0.264	0.104	-0.007	0.968
20	0.319	0.048	0.006	0.973
50	0.330	0.040	0.047	0.777

Table 2. Inverse Normalization: Correlation and p-values for different thresholds k under uniform and patchy environments.

Threshold	Uniform		Patchy	
	Corr	p-val	Corr	p-val
1	0.159	0.334	0.206	0.214
2	0.303	0.060	0.086	0.608
5	0.255	0.117	-0.020	0.904
10	0.266	0.102	-0.010	0.953
20	0.294	0.069	-0.013	0.937
50	0.274	0.092	-0.015	0.927

Table 3. Exponential Normalization: Correlation and p-values for different thresholds k under uniform and patchy environments.

It must also be noted that the significant features are here correlated positively with ASRS score, while Figure 9 indicated that the significant features were negatively linked, and that *MSRs* computed with similar parameters ($k = 50$ in the uniform environment, with an inverse normalization), were not clearly positively correlated.

5.4. Trajectory Generation

Figure 10 shows some of the generated trajectories. As discussed in [1], some are considered outliers as they have points that are outside the feasible space, which can be defined here as the island. We found that deleting every trajectory that had any point outside of the hull led to the deletion of the vast majority of trajectories.

We then went on the task of witnessing the reconstruction capabilities of the encoder decoder. As in WildGen, we implemented a Savitzky-Golay smoothing filter [14],

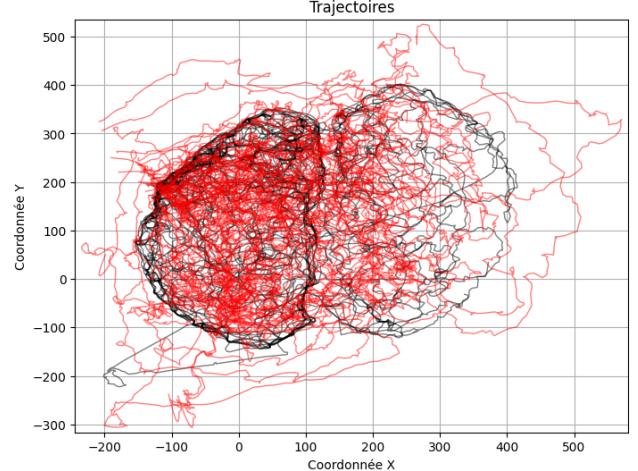


Figure 10. Generated trajectories (in red) based on the sampling in the latent space, and original trajectories (black)

to which we changed the parameters to have a trajectory that most closely resembles the original. In fact, generated and reconstructed trajectories tend to be more noisy before treatment, which makes such a filter useful (Fig. 11).

Seeing that trajectories change as they go through the model, we wanted to quantify this change in relation to the features we extracted previously. Figure 12 shows how these features are preserved when generating on the uniform and on the patchy trajectories. *The following part does not use any smoothing filter, in order to evaluate the model itself.*

We observe that :

- **Average distance to hull** tends to increase on synthetic trajectories. This is expected, as some players followed the shoreline, minimizing their distance, when the generation doesn't know anything about the environment. This behavior is therefore expected
- **Trajectory diameter** tends to be lower : the noisy behavior of the VAE tends to lower the overall diameter, as we generate the same number of points but a lot of them tend to go back and forth.
- **Time spent on one side of the island** is well preserved in generation. Most of the synthetic trajectories understand that most of their points should be on the right side of the map, which is not defined to the model.
- **We observe slight differences** depending on the original distribution. For example, we observe that on uniform trajectories, synthetic trajectories do generate a median ratio of time spent on the left at 0, which is extreme, but fits the input trajectories.

The table 4 contains the measured Kolmogorov-Smirnov test values, which measure the differences in the distribu-

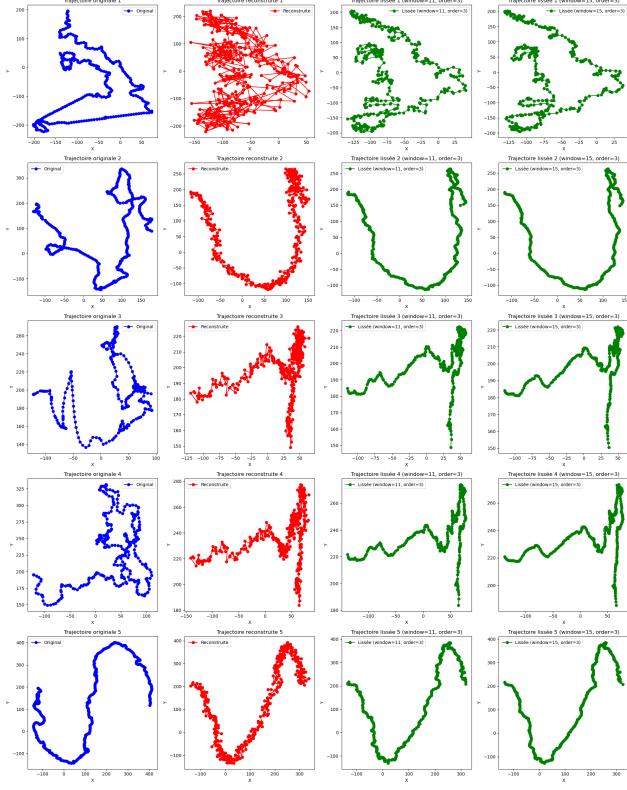


Figure 11. Examples of reconstructions, without filtering and with filtering with different parameters

tions. Applying the smoothening filter lowers the average distance to the hull (by reducing the noisy oscillations), but has no impact on the other two features.

Var	Type	KS Test	p-value	Avg. diff.
1	Patchy	0.3576	0.0082	-22.2
	Uniform	0.4103	0.0025	-23.4
2	Patchy	0.2490	0.1435	37.3
	Uniform	0.2821	0.0897	48.3
3	Patchy	0.1653	0.5979	0.0
	Uniform	0.1026	0.9885	0.0

Table 4. Comparison of the Kolmogorov-Smirnov values of 1. average distance to hull, 2. diameter, 3. ratio of time spent on the left side of the island.

Finally, we wanted to see how the model was able to keep features when reconstructing from the encoded version of the trajectory. We do not measure any difference in distributions, but we focus on the reconstruction of each trajectory.

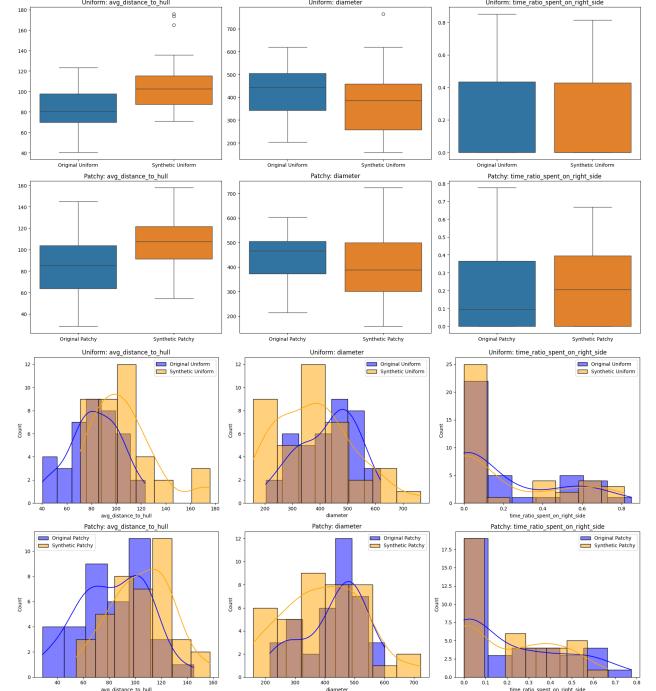


Figure 12. Box plots and distributions of the three extracted features. First it the experiment on uniform dataset, then on patchy. Blue represents the original data

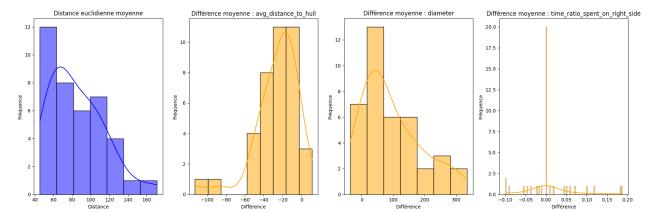


Figure 13. Restitution of the features when decoding the encoded trajectories .

Fig 13 shows the same reconstruction bias as before, with average distance to hull increasing, diameter lowering, and good performances when imitating the time ratio feature. This suggests that the model learns this feature better than the others.

Looking now at HMMs, Figure 14 displays that the definition of the states itself is as import as the transition probabilities between those, as we see a strong correlation between unexplored and state-sparse zones. This is natural considering we learn states based on the centroids of a kNN trained on the real trajectories, which also explore less the zones that are further from the starting point. In addition, we notice that rare real trajectories (like the one exploring the isolated piece of land) are represented in the generated data. Moreover, a more fine-grain analysis on the latter

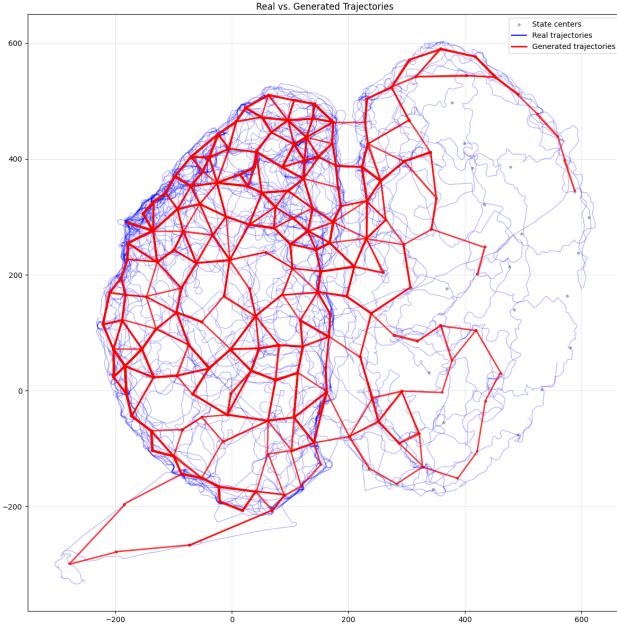


Figure 14. Generated trajectories on a "patchy" environment using HMMs ($n = 200$ states). We show as many generated trajectories as there are of real ones.

show that it exhibits trajectories that go back on their step after reaching certain states, which players tend to avoid, perhaps motivated by the idea that the zones they already explored are less reward-dense.

Resolution	50x50	100x100	200x200	500x500
Coverage	0.1726	0.0651	0.0292	0.0135
Exploration	0.0084	0.0689	0.1881	0.6077
Overlap	237.0	284.0	341.0	357.0
Real Cells	1373.0	4362.0	11697.0	26493.0
Gen Cells	239.0	305.0	420.0	910.0

Table 5. Coverage, exploration, $N_{overlap}$, N_{real} , and N_{gen} of generated trajectories for different resolutions.

Table 5 looks at more quantitative features, such as the aforementioned coverage and exploration metrics, with multiple grid resolutions. One must take into account that high resolutions does not mean more precision, since a lot of variance is explained through the positions of state centers, determined as developed earlier with a kNN. Indeed, because of the sparsity of real trajectories, state centers sometimes occur in the middle of an unexplored zone, representing patches or paths going around those centroids. We are hence unsurprised to see exploration arise as the resolution gets finer. Overlapping cells take a substantial share of all the area covered by the generation, showing a shy risking in out-of-distribution or rare states.

6. Discussion

6.1. Method capabilities

Firstly, it seems important to note that we fail to observe accurate prediction results, especially using deep-learning based approaches. This can be explained by multiple reasons : the small size of the dataset, the non-verifiability of our augmentation hypotheses (which will be discussed in the following), the lack of existing baseline methods to rely on similar tasks.

Most importantly, we realise that we suffer crucially from undertraining, as is highlighted by Figure 8, where the distribution of predicted scores in abnormally close to the mean.

Nonetheless, we manage to highlight significant features for ADHD behaviour, thanks to the significant values shown in Table 1. In addition, this is roughly consistent with Figure 7. One must note the difficult interpretability of the latter, since there is high correlation between variables used by the random forest predictor.

Furthermore, Figure 9 and Table 2 highlight the robustness of game-theoretic approaches to illustrate explorative and ADHD behaviours.

In addition, [1] enables for a handful of experiments that have and could yet be conducted. We thus thank immensely the authors for their work, making for such a contribution on our task.

6.2. Hypothesis limitations

The hypotheses proposed for studying ADHD-related behavior in 2D trajectories provide a structured foundation for exploration, yet several theoretical and methodological limitations warrant critique. Below, we address each hypothesis in turn.

Environment-Independence of ADHD Features (1): The assumption that ADHD-related features are globally environment-independent oversimplifies the interaction between behavioral patterns and environmental contexts. Empirical evidence from reinforcement learning and neurodevelopmental studies suggests that ADHD behaviors are highly sensitive to external stimuli, such as task demands and reward structures [12]. Ignoring environmental variability may result in models that fail to capture context-specific ADHD manifestations, thereby limiting generalizability.

Uniform Temporal Distribution of ADHD Behavior (2): The hypothesis of temporally uniform ADHD behavior assumes equal likelihood of occurrence across time, potentially overlooking the episodic nature of ADHD symptomatology.

tology. Studies show that attentional lapses, hyperactivity, and impulsivity exhibit variability over time, influenced by fatigue, stress, or task complexity [7]. Temporal assumptions should consider periodicity or clustering in behavior, aligning with findings in time-series ADHD modeling [4].

Reliability of ASRS Scores (3): While the reliability of ASRS scores has been validated in prior studies [15], test-retest variability may still arise due to mood, environmental changes, or subjective reporting biases. Relying solely on ASRS scores may obscure dynamic behavioral patterns critical to understanding ADHD trajectories.

Minimum Time Interval for Feature Extraction (4):

The hypothesis of a minimal time interval (Δt) for feature extraction is plausible but underexplored. Neuroscience research indicates that meaningful ADHD-related patterns can emerge at sub-second intervals, particularly in fine-grained motion or reaction-time data [17]. A rigid assumption about Δt may neglect high-frequency behavioral markers detectable with advanced signal processing techniques.

Overall, these hypotheses provide valuable starting points but require refinement through interdisciplinary evidence from reinforcement learning, cognitive neuroscience, and psychiatry.

6.3. Differences between patchy and uniform environments

section 5 and section 6 highlight varying statistical analysis when switching between patchy and uniform environments. Nonetheless, many of our attempts at distinguishing the latter failed to notice relevant differences. This raises the need for models taking into account the environment more strongly to adapt knowledge on trajectory and player behaviour. As developed just above, the assumption that ADHD-related features are environment-agnostic is oversimplified, and must be written off in future work. Furthermore, it might be relevant to notice that the strongest observations that differ between both are when looking at *MSR*, which is a game-theoretic interpretation of expected choices of other players. This indicates that there is further exploration needed in that regard.

7. Conclusion

This work was for us an enriching experience, meeting at the complex intersection between Machine Learning, its wide range of methods, statistical intricacies, and the practical, experimental domain of psychology, medicine, and psychiatry. Though first classical methods struggled to show decisive insights on explaining ADHD behaviour through

the patients' constrained trajectories, we still obtain statistically significant and interpretable variations in ADHD score, be it purely topological or more reward-oriented. Later on, revisiting trajectory generation, we manage to reproduce representative player behaviour, and quantify the quality of our reconstructions. In the future, we aim to keep working on this project, and deeply thank the Neuroforest team.

References

- [1] Ali Al-Lawati, Elsayed Eshra, and Prasenjit Mitra. Wildgen: Long-horizon trajectory generation for wildlife, 2023. [2](#), [4](#), [6](#), [8](#), [10](#)
- [2] Ali Al-Lawati, Elsayed Eshra, and Prasenjit Mitra. Wild-graph: Realistic long-horizon trajectory generation with limited sample size. In *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*, page 247–258. ACM, 2024. [2](#)
- [3] Leonard E Baum, Ted Petrie, Geoffrey Soules, and Neal Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970. [5](#)
- [4] Maria L. Cervantes-Henríquez et al. Machine learning prediction of adhd severity: Association and linkage to adgrl3, drd4, and snap25. *Journal of Attention Disorders*, 2022. [11](#)
- [5] Bowler A. Moses-Payne M. Dubois, M. and J. Habicht. Tabula-rasa exploration decreases during youth and is linked to adhd symptoms. *BioRxiv*, 2020. [1](#)
- [6] M. Dubois and T. U. Hauser. Exploring too much? the role of exploration in impulsivity. *Nature Communications*, 2022. [1](#)
- [7] T.U. Hauser, R. Iannaccone, J. Ball, et al. Role of the medial prefrontal cortex in impaired decision making in juvenile adhd. *JAMA Psychiatry*, 2014. [11](#)
- [8] Maarten de Jong. Game theory of loss aversion, 2018. [6](#)
- [9] Cecchi G. Bouneffouf-D. Lin, B. and J. Reinen. Reinforcement learning models of human behavior: Reward processing in mental disorders. *ResearchGate*, 2020. [1](#)
- [10] Takuya Maekawa, Kazuya Ohara, Yizhe Zhang, Matasaburo Fukutomi, Sakiko Matsumoto, Kentarou Matsumura, Hisashi Shidara, Shuhei J. Yamazaki, Ryusuke Fujisawa, Kaoru Ide, et al. Deep learning-assisted comparative analysis of animal trajectories with deephl. *Nature*, 11(5316), 2020. [2](#), [3](#), [4](#)
- [11] Lawrence R Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989. [5](#)
- [12] Lucy Riglin, Stephan Collishaw, Soren Dalsgaard, et al. Association of genetic risk variants with adhd trajectories in the general population. *JAMA Psychiatry*, 2016. [10](#)
- [13] Leon J. J. Rodriguez-Herrera, R. and P. Fernandez-Martin. Contingency-based flexibility mechanisms through a reinforcement learning model in adults with adhd. *MedRxiv*, 2024. [1](#)
- [14] Abraham. Savitzky and M. J. E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36(8):1627–1639, 1964. [8](#)
- [15] Michael J Silverstein, Samuel Alperin, Stephen V Faraone, Ronald C Kessler, and Lenard A Adler. Test-retest reliability of the adult adhd self-report scale (asrs) v1.1 screener in non-adhd controls from a primary care physician practice. *Family Practice*, 35(3):336–341, 2017. [3](#), [11](#)
- [16] Xiaojia Wen. Loss aversion in asymmetric anti-coordination games. *Southern Economic Journal*, 89(2):475–495, 2022. [6](#)
- [17] A. Wiehler and J. Peters. Dopaminergic modulation of the exploration/exploitation trade-off in human decision-making. *Elife*, 2020. [1](#), [11](#)