Adaptive Posture Regulation with Visual Stimuli

Oussama Jaffal
Simon Gmür
Imane Jennane

EE-568 EPFL

Abstract

This project addresses the issue of poor and static posture caused by prolonged tablet use. We propose an adaptive posture regulation system that uses real-time monitoring and dynamic visual feedback to encourage healthier posture habits. The system continuously observes user posture and provides subtle visual prompts to promote regular posture changes.

We developed and evaluated control learning algorithms within a reinforcement learning framework, creating a simulation environment that mimics real-world scenarios with varying user behaviors and fatigue levels. We implemented and compared two reinforcement learning algorithms: Double Q-Learning and Expected SARSA. Both algorithms demonstrated effective adaptation to user behavior, with Double Q-Learning showing steady reward increases and Expected SARSA handling a highly stochastic environment well.

Our findings illustrate the potential of reinforcement learning techniques in developing adaptive systems for long-term posture management, integrating seamlessly into daily routines and promoting significant health benefits.

Keywords: Posture Regulation, Model-based Reinforcement Learning, Probabilistic Approach.

1 Introduction

In today's digital age, prolonged periods of sitting and working at a computer or tablet are common, leading to an increase in posture-related health issues. Simply not changing posture for extended periods can already cause certain circulation issues, such as decreased blood circulation. Addressing these concerns requires innovative solutions that promote better posture habits in an engaging and effective manner.

This project aims to tackle this challenge by leveraging real-time posture monitoring and dynamic visual feedback to encourage users to change sitting postures at a healthy frequency.

The core concept of the project involves using a sensor and posture estimation models to continuously observe a user's posture as they work on a screen. The interactive element is the window moving around the screen, whcih serves as a visual stimulus designed to subtly prompt the user to change their posture as often as possible. The adaptive nature of this mechanism provides ongoing, personalized guidance. Moreover, this system could encourage the development of better habits over time. The ultimate goal is to create a system that integrates seamlessly into a user's daily routine, promoting long-term health benefits without causing significant disruption to their work

Our contribution to this broader project focuses on comprehensive problem modeling, algorithm development and performance evaluation within a reinforcement learning framework. Initially, we modeled the problem of adaptive posture regulation and implemented various control learning algorithms for comparison. These algorithms were designed to determine optimal policies for moving the visual stimulus based on real-time posture-changes.

We created a robust simulation environment to emulate real-world scenarios where user posture is continuously monitored. To realistically simulate diverse user behaviors, we generated artificial users with varying probabilities of adjusting their posture in response to the visual stimulus and different fatigue rates that affect their responsiveness over time. This approach ensured that our simulation environment accurately reflected the dynamic nature of real-world user interactions.

2 Background and Inspiration

In order to delve into our project, we drew inspiration from several key papers that explore various aspects of reinforcement learning (RL) and its applications.

2.1 Environment setup

The main paper that inspired this proposed study is: **"WriteUpRight: Regulating Children's Handwriting Body Posture by Unobtrusively Error Amplification via Slow Visual Stimuli on Tablets"** [5]. It introduces a system aimed at helping children maintain proper posture while using tablets for handwriting tasks. The system employs two intervention strategies — Error Amplification (EA) and Error Correction (EC) — that utilize slow visual stimuli to either amplify or correct improper postures. A study conducted with 42 children demonstrated that the EA strategy was particularly effective in encouraging self-correction and improving posture quality. The findings suggest that unobtrusive visual stimuli can effectively regulate children's posture, thereby enhancing both their physical well-being and handwriting performance.
This paper served as a backbone to our broader project, and even though the specific methods used were not in the roam of RL, the overall experiment design and execution were adoptable in our context.

2.2 Reinforcement Learning with human feedback

Weber et al. (2018) [3] investigated the use of Reinforcement Learning (RL) to help robots adapt their humor based on user reactions, such as laughter and smiles. Utilizing a Reeti robot, this study demonstrated how implicit feedback can guide a robot to refine its joke-telling over time, thus enhancing its entertainment capabilities. By comparing a learning robot to a non-learning baseline, the research showcased the effectiveness of RL in improving user engagement, offering valuable insights into the potential of adaptive behaviors in socially-aware robotics.

Building on these insights, **Ritschel et al. (2017)** [1] extended the concept of RL to adapt a robot's linguistic style to match user preferences. The goal of their study was to enhance user engagement by dynamically adjusting the robot's personality traits, particularly extraversion, during storytelling. By leveraging RL and real-time social signals, the robot could personalize its interactions, making them more engaging and responsive. This research highlighted the broader potential of adaptive behaviors in enhancing human-robot interaction across various contexts.

For our project's purposes, both papers gave us valuable insights into how to incorporate RL based approaches, and guided us through important model design decisions, such as the choice between episodic or continuing framework.

2.3 Data generation

This is a pioneer study in this field and consequently we aim to verify the feasibility of a model capable of capturing certain patterns in human behavior. We focused on simulating data as close as possible to real world cases. To address this challenge, the paper **"Incorporating Human Fatigue and Recovery into the Learning–Forgetting Process"** [4] introduces the Learning–Forgetting–Fatigue–Recovery Model (LFFRM).

This paper specifically proposes a concrete quantification of fatigue, which we drew inspiration from and included into our simulated user models. We judged aspects like recovery during breaks as irrelevant in our context.

2.4 Stochastic environment

Our environment setup is a discrete space and we focused on the Q Learning and SARSA algorithms and their derivatives. Given the probabilistic nature of our environment, it was crucial to choose algorithm variations that are robust to noise and stochasticity.

Double Q-Learning and Expected SARSA are reinforcement learning algorithms designed to handle the challenges of stochastic environments. Double Q-Learning, introduced by Hado van Hasselt [6], addresses the overestimation bias in traditional Q-learning by maintaining two Q-tables and using them alternately for action selection and evaluation, resulting in more accurate value estimates and improved performance in noisy settings. Expected SARSA, discussed in "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andrew G. Barto [7], enhances the standard SARSA algorithm by using the expected value of future rewards for updates. This method provides more stable learning and is robust to environmental stochasticity, balancing exploration and exploitation more effectively.
Both algorithm derivatives demonstrate significant improvements in environments with high uncertainty and variability and are consequently very suiting for this project's purpose.

3 Approach

In this study, we designed a discrete environment to evaluate the performance of both, the Double Q-Learning and the Expected SARSA algorithms. The environment is structured as a 6x6 grid where the window can navigate through different states, interacting with the environment and receiving rewards based on its actions and the user's reactions.

Environment Setup

The 6x6 grid represents and models the tablet's screen. Each cell in the grid can be in one of the following states:

- **NOTHING** (0): Represents an empty or black screen.
- **WINDOW** (1): Represents the current position of the window.

The window starts at a specific position on this grid and can move within the grid based on the defined actions. The possible actions are:

- **0**: Move up
- **1**: Move down
- **2**: Move left
- **3**: Move right

The table below visualizes a 6x6 grid environment with an example initial setup where the window starts at position 16.

Table 1: Visualization of the 6x6 Grid Environment

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	**1**	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

The window can move within this grid based on the action taken, and the environment is updated accordingly. The state space is represented as a 36-element numpy array because of the nature of Gym environments, where each element can take a single binary value.

Human Fatigue

Our approach to modeling human posture regulation incorporates a nonstationary framework designed to account for the variability in posture changes caused by factors such as fatigue, attention, and individual habits. Central to our model is a fatigue parameter, λ, which influences the probability of posture changes over time. Every generated user receives their own fatigue parameter, randomly attributed, but within a reasonable and representative predefined range [4]. The fatigue is modeled as follows:

$$F(t) = 1 - e^{-\lambda t}$$

Model Design

We adopted a **discrete** problem model because of the nature of our environment setup, having a discrete state space for the position of the window inside the screen. This model also assumes that the user's posture regulation is a process influenced by various external and internal factors over time. Based on the assumption that changing posture is a momentary and instinctive action, we decided to design stateless models, that make decisions based solely on the current state. Stateful models could have been just as interesting to justify and to investigate.

Each iteration within our system lasts one minute, providing frequent opportunities for updates and adjustments based on user posture changes. This interval was chosen to balance the need for timely feedback with the practical considerations of user comfort and sensor limitations. To accurately simulate real-world conditions, we hypothesized that the learning agent would require approximately 2-3 days to adapt to an individual user, with the assumption of two one-hour sessions per day. This duration allows the system to capture a wide range of posture behaviors and adjust accordingly.

Additionally, we introduced a 5% noise range to account for potential sensor errors in posture detection, reflecting the inherent inaccuracies in any real-world measurement system. This noise range helps the model to remain robust against minor fluctuations and ensures that it does not overreact to insignificant changes. We also decided against including a 'no-action' scenario for maintaining the same position, as continuous adaptation turned out to be essential for effective posture regulation.

Expected SARSA

Expected SARSA [7] is a reinforcement learning algorithm that updates the action-value function based on the expected value of the next state's Q-values. This method helps in dealing with stochastic environments.

Algorithm 1 Expected Sarsa

Initialize $Q(s, a)$ arbitrarily for all s, a
loopover episodes
    Initialize s
    repeatfor each step in the episode
        Choose a from s using policy π derived from Q
        Take action a, observe r and s'
        $V_{s'} = \sum_a \pi(s', a) \cdot Q(s', a)$
        $Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma V_{s'} - Q(s, a)\right]$
        $s \leftarrow s'$
    until s is terminal
end loop

Double Q-Learning

Double Q-Learning [6] is an extension of Q-learning that uses two Q-tables to reduce overestimation bias in action-value estimates. This method also helps in handling stochastic environments effectively.

Algorithm 2 Double Q-learning

Initialize Q^A, Q^B, s
repeat
  Choose a, based on $Q^A(s, \cdot)$ and $Q^B(s, \cdot)$, observe r, s'
  Choose with a 0.5 probability either UPDATE(A) or UPDATE(B)
  if UPDATE(A) **then**
    Define $a^* = \arg\max_a Q^A(s', a)$
    $Q^A(s,a) \leftarrow Q^A(s,a) + \alpha(r + \gamma Q^B(s', a^*) - Q^A(s,a))$
  else if UPDATE(B) **then**
    Define $b^* = \arg\max_a Q^B(s', a)$
    $Q^B(s,a) \leftarrow Q^B(s,a) + \alpha(r + \gamma Q^A(s', b^*) - Q^B(s,a))$
  end if
  $s \leftarrow s'$
until end

By alternating updates between Q^A and Q^B:

- **Action Selection:** One Q-function (Q^A) is used to select the best action a^* based on the current state s'.

- **Evaluation:** The other Q-function (Q^B) evaluates this action to update the value of the current state-action pair (s, a).

This mechanism ensures that the evaluation is unbiased, as it is not influenced by the same Q-values used to select the actions. Consequently, Double Q-learning produces more reliable value estimates, leading to better performance in environments with high variability and stochasticity.

4 Results

In this section, we present the performance evaluation of the two reinforcement learning algorithms implemented, Double Q-Learning and Expected SARSA. For the settings of the environment, the fatigue factor was chosen according to the intensity of the work required on the desk [4] from a range of 0.0001 to 0.0002, and was attributed randomly to each user accounting for different fatigue levels depending on users. The transition probabilities from a state to another were also generated randomly, from a range 0.3 to 0.7 for each state-action pair and for each user. We chose these values for the transition probabilities to have an expected outcome of 0 if a random path generator was to chose actions in this space and without accounting for fatigue. We analyzed the effects of both algorithms with and without accounting for the fatigue factor. The environment's stochastic nature poses significant challenges, making it difficult but not impossible to grasp the intricacies of this environment. We fine-tuned the hyper-parameters to showcase a better balance between exploration and exploitation, through an extensive grid-search.

4.1 Double Q-Learning

Without Fatigue

The cumulative rewards and RMSE (Root Mean Squared Error) for all users utilizing Double Q-Learning without fatigue are shown in the figures below. The results indicate a general steady increase in cumulative rewards, showcasing the learning process of the agents. The RMSE plot shows convergence, indicating the stability and accuracy of the learning process.

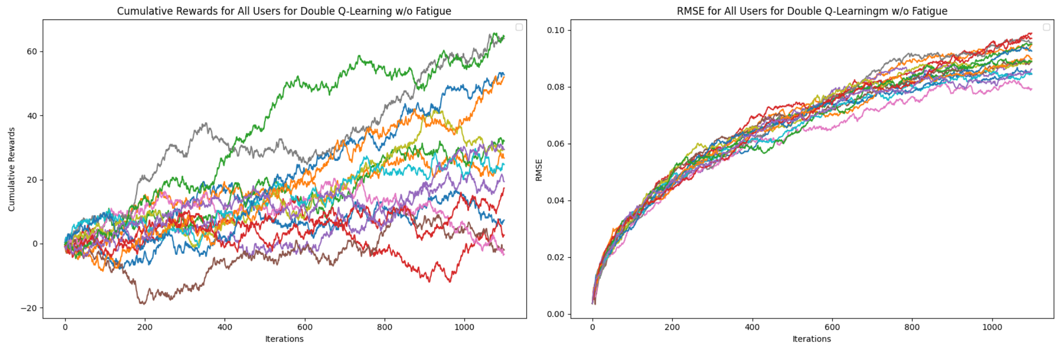

Figure 1: Cumulative reward and RMSE of Double Q-Learning algorithm without Fatigue integration

With Fatigue

When adding the fatigue factor into Double Q-Learning, it presents a more realistic scenario where the agent's performance degrades over time. The cumulative rewards decrease after reaching a peak, demonstrating the impact of fatigue. Despite this, the algorithm shows resilience in adapting to the changes and does not collapse. The RMSE plot similarly shows convergence but with a slower rate compared to the no-fatigue scenario.

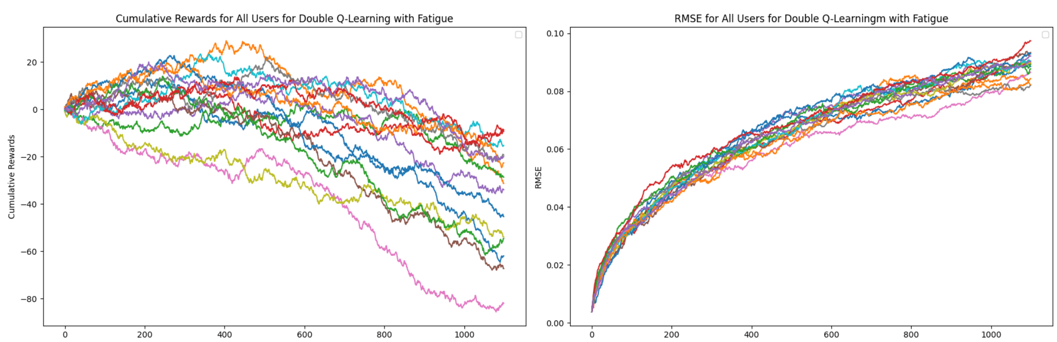

Figure 2: Cumulative reward and RMSE of Double Q-Learning algorithm with Fatigue integration

4.2 Expected SARSA

Without Fatigue

The cumulative rewards for Expected SARSA without fatigue show a steady increase trend, showing effective learning. The algorithm successfully navigates the stochastic environment, gradually improving its policy.

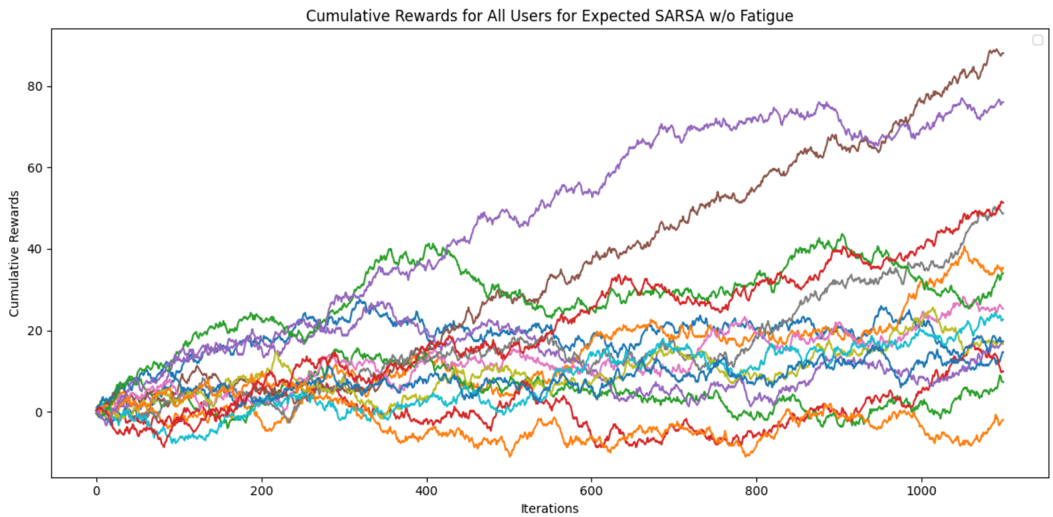

Figure 3: Average reward and RMSE of SARSA algorithm

With Fatigue

When fatigue is introduced, the cumulative rewards for Expected SARSA decline after reaching a certain peak value, highlighting the impact of fatigue on the agent's performance, but without inducing a model failure.

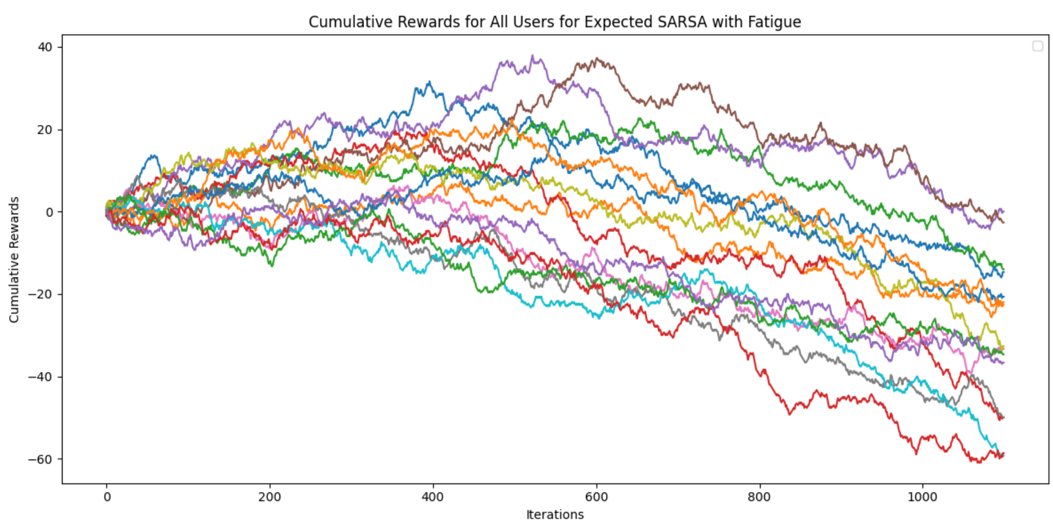

Figure 4: Average reward and RMSE of SARSA algorithm

5 Conclusion

In this project, we attempted to developed an adaptive posture regulation system using reinforcement learning techniques to address static posture during prolonged tablet use. By leveraging real-time posture monitoring and dynamic visual feedback, our system encourages healthier posture habits in an engaging and effective manner. Our approach incorporated a non-stationary framework with a fatigue parameter to dynamically adjust the probability of posture changes over time, ensuring the model's responsiveness to the user's state. To ensure compatibility with the real-world applications, one strategy is the discretisation of continuous spaces, however this can lead to a loss of precision and may not scale well to this problem. Another way to handle this situation is to use Function Approximation to approximate the value function or policy.

In the end, we implemented two reinforcement learning algorithms — Extended SARSA and Double Q-Learning — to determine optimal policies for moving the visual stimulus. The results showed that our system is able to learn and adapt to user behavior, providing consistent performance improvement and accurate posture adjustments. This project illustrates the potential of advanced reinforcement learning techniques in creating robust, adaptive systems for long-term posture management, promoting health benefits without disrupting users' daily routines.

References

[1] Tobias Baur, Hannes Ritschel, and Elisabeth Andre, *Adapting a robot's linguistic style based on socially-aware reinforcement learning*, 2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), 2017.

[2] Michael Widrich, Thomas Unterthiner, Johannes Brandstetter, Sepp Hochreiter, and Jose A. Arjona-Medina, *Rudder: Return decomposition for delayed rewards*, LIT AI Lab, Institute for Machine Learning Johannes Kepler University Linz, Austria, Institute of Advanced Research in Artificial Intelligence (IARAI), 2019.

[3] Ilhan Aslan, Florian Lingenfelser, Elisabeth André, Klaus Weber, and Hannes Ritschel, *How to shape the humor of a robot - social behavior adaptation based on reinforcement learning*, ICMI'18, October 16-20, 2018, Boulder, CO, USA, 2018.

[4] M.Y. Jaber, Z.S. Givi, and W.P. Neumann, *Incorporating human fatigue and recovery into the learning–forgetting process*, Department of Mechanical and Industrial Engineering, Ryerson University, Toronto, ON, Canada, 2013.

[5] B. Bruno, P. Dillenbourg, Chenyang Wang, and D. Carnieto Tozadore, *Writeupright: Regulating children's handwriting body posture by unobstrusively error amplification via slow visual stimuli on tablets*, CHI Conference on Human Factors in Computing Systems Proceedings, Honolulu, HI, USA, May 11–16, 2024, 2024.

[6] Hado van Hasselt, *Double Q-learning*, Advances in Neural Information Processing Systems (NeurIPS), 2010.

[7] Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction" (Chapter on Expected SARSA), Reinforcement Learning: An Introduction, Second Edition, 2018.

A Appendix

A.1 Libraries and Sources citations

The libraries imported and used throughout this project were: gym, pprint, numpy, tensorflow, tensorflow probability, tf agents, random, IPython.display and matplotlib.

A.2 Role of each group member

While we met as a group and worked on the project together, Oussama took the lead on the code and proactively worked on many more algorithms than initially discussed. This proved tremendously helpful later on during the project, when we had to redirect our efforts. Simon took a lead on the poster and on the final report, and finalized it in strong collaboration with Oussama. Imane and Oussama took charge of the discussions with the Chilli lab supervisor. Oussama and Simon thoroughly discussed the redirection of the project and approached supervisors and TAs for advice, while designing a new approach. Overall, all team members contributed greatly, but the special award goes out to Oussama!