Exploration of Parameters

Samuel Burns

1 Introduction

Deep Q-learning's (DQN) performance relies on careful selection of hyperparameters, each playing a crucial role in the learning process. This study explores these hyperparameters through controlled experimentation, focusing on key parameters that influence learning efficiency and final performance. The parameters below were selected to align with common practices in DQN, balancing exploration, exploitation, stability, and learning speed.

2 Learning Rate (α)

The learning rate, (α), controls how much the neural network's weights are adjusted during each update. In DQN a common range of α is between 1×10^{-3} and 1×10^{-4}. I chose to vary α on 3 values within reason: 1×10^{-4}, 5×10^{-4}, and 1×10^{-3}.

- 1×10^{-4}: Very cautious and allows for small, but safe updates. Convergence will be slower.
- 5×10^{-4}: Medium choice which balances stability and learning speed.
- 1×10^{-3}: Quick learning rate, but may be unstable.

I avoid very high rates (like 1×10^{-2}) to prevent likely destabilization in training. I also stayed lower than 1×10^{-4} to keep the training from being too slow. The 3 values create a balance between stability and efficiency, which is common in DQN.

3 Discount Factor (γ)

The discount factor, γ is responsible for the significance of future rewards compared to present ones. I select three values: 0.95, 0.99, and 0.999, which balances weights between short term and long term rewards:

- 0.95: Near-sighted and immediate reward prioritization.
- 0.99: A balance weighing immediate and future rewards more evenly.
- 0.999: Long-term reward prioritization.

The 3 values cover a healthy range without going to extremes (like 0.8 or 0.9999), which might either completely disregard future rewards or weight rewards too far in the future too high and significantly slow down learning.

4 ϵ-Decay Rate (ϵ)

For the ϵ-greedy exploration strategy, the ϵ-decay rate controls how fast exploration reduces over time:

- 0.995: A slower decay that allows for extended exploration, at the risk of slow convergence.
- 0.98: A faster decay that quickly minimizes exploration, allowing for more exploitation but a risk of converging on a sub-optimal solution.

These two rates will be sufficient to observe the importance of exploration time in learning. The small amount of decay values allowed me to keep the experiments timely and computationally manageable, with a focus on typical DQN exploration rates.

5 Soft Update Parameter (τ)

The soft update parameter τ controls how quickly the target network meshes with the local network. Here, $\tau = 1 \times 10^{-3}$:

- A smaller τ (1×10^{-3}) creates a stable target network that updates slowly, reducing divergence and ensuring stable learning.
- Fixing τ simplifies the experiment, allowing focus on more sensitive parameters like (α, γ, and ϵ-decay).

In addition, this value is a common default in DQN that balances stability and reactivity.

Setup 1

- **Network Structure:** 3 fully connected hidden layers ($h_1, h_2, h_3 \in R^{256}$) using ReLU activation (Rectified Linear Unit):

$$f(x) = \max(0, x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases}$$

- **Memory System and Batch Size:** Experience replay buffer of 10^5 sample and batch size of $n = 32$
- **Network Width:** Consistent hidden layer dimensions of ($w = 256$) neurons
- **Number of Episodes Per Set of Parameters:** n = 10,000 (sets a reasonable upper limit to ensure the least efficient parameter configurations do not excel)

Exp 1: $\alpha = 1.0e-04$, $\gamma = 0.950$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 2: $\alpha = 1.0e-04$, $\gamma = 0.950$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 3: $\alpha = 1.0e-04$, $\gamma = 0.990$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)
Exp 4: $\alpha = 1.0e-04$, $\gamma = 0.990$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 5: $\alpha = 1.0e-04$, $\gamma = 0.999$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 6: $\alpha = 1.0e-04$, $\gamma = 0.999$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)
Exp 7: $\alpha = 5.0e-04$, $\gamma = 0.950$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 8: $\alpha = 5.0e-04$, $\gamma = 0.950$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 9: $\alpha = 5.0e-04$, $\gamma = 0.990$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)
Exp 10: $\alpha = 5.0e-04$, $\gamma = 0.990$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 11: $\alpha = 5.0e-04$, $\gamma = 0.999$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 12: $\alpha = 5.0e-04$, $\gamma = 0.999$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)
Exp 13: $\alpha = 1.0e-03$, $\gamma = 0.950$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 14: $\alpha = 1.0e-03$, $\gamma = 0.950$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 15: $\alpha = 1.0e-03$, $\gamma = 0.990$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)
Exp 16: $\alpha = 1.0e-03$, $\gamma = 0.990$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 17: $\alpha = 1.0e-03$, $\gamma = 0.999$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 18: $\alpha = 1.0e-03$, $\gamma = 0.999$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)

Figure 1: Key for Setup 1

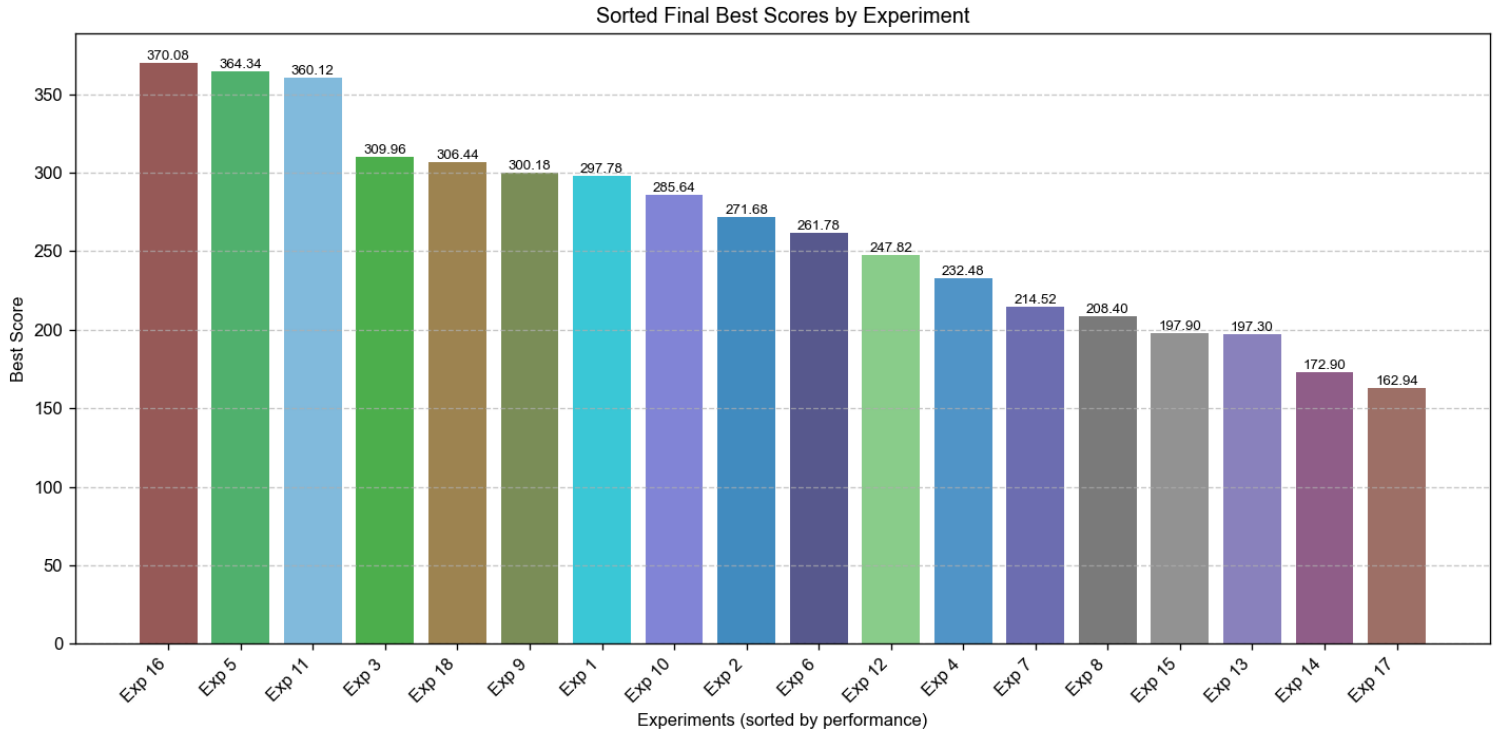

Figure 2

Learning Rate vs Discount Factor for $\epsilon - decay = 0.98$

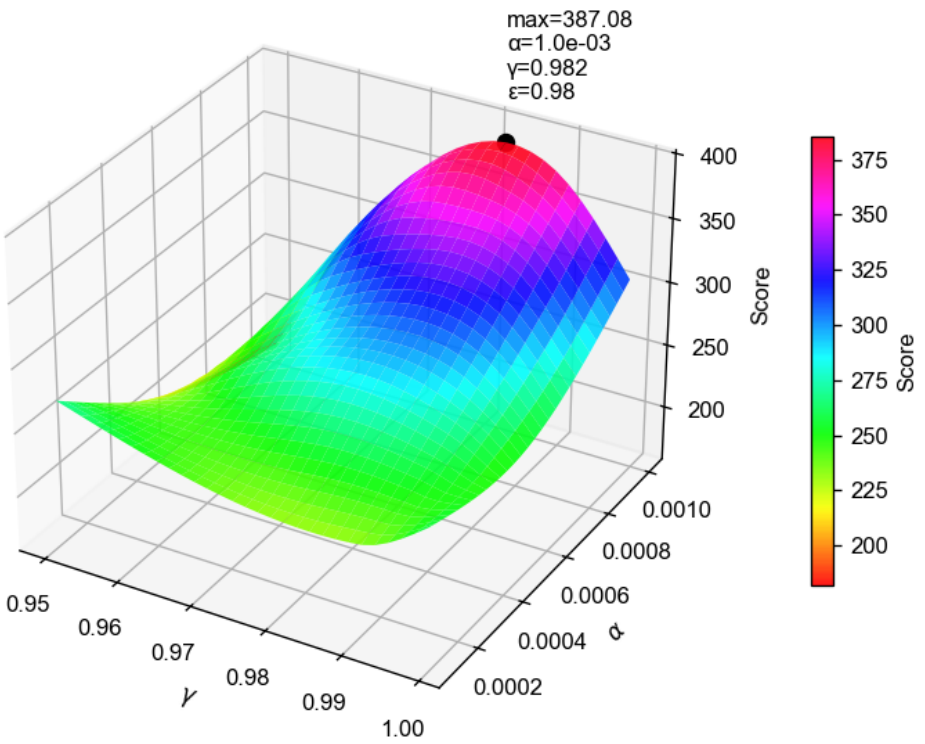

Figure 3

Learning Rate vs Discount Factor for $\epsilon - decay = 0.995$

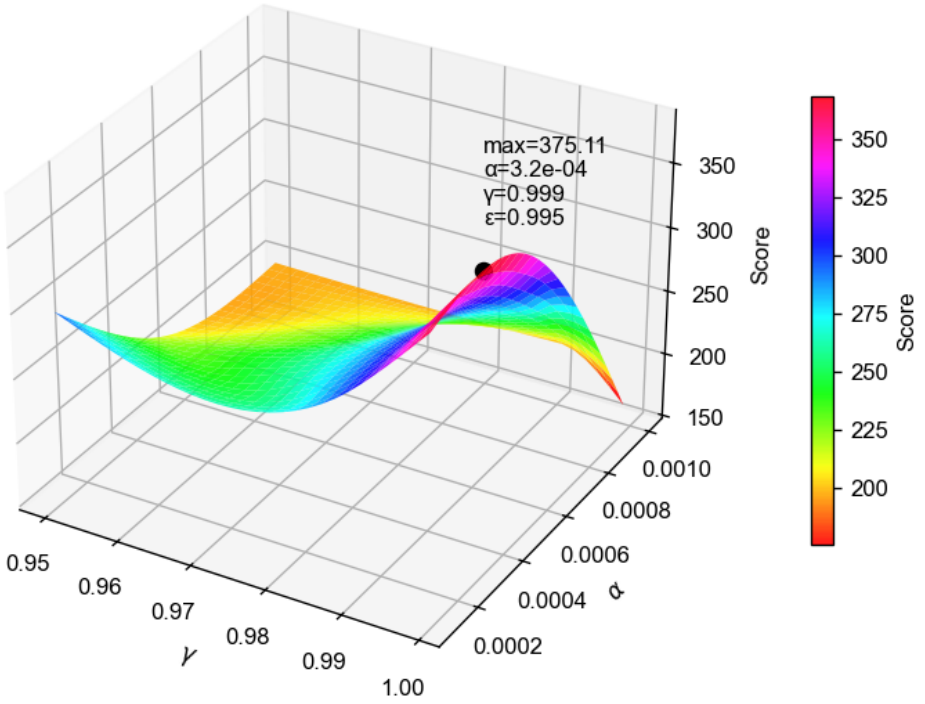

Figure 4

Performance Highlights: Hyperparameter Analysis for 32 Batch Size

Best Configurations

- **Highest Score - Experiment 16:**
  - Score: 370.08
  - Configuration: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.990$, $\epsilon = 0.98$
  - Characteristics: Aggressive learning rate, high discount, faster exploration
- **High Performers:**
  - **Experiment 5:**
    - Score: 364.34
    - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.995$
    - Characteristics: slower/steady learning, maximum discount, slower exploration
  - **Experiment 11:**
    - Score: 360.12 points
    - Configuration: $\alpha = 5.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.995$
    - Characteristics: Moderate learning rate, maximum discount, slower exploration

Surface Analysis for $\epsilon = 0.98$ *(figure 3)*

- **Peak Performance:**
  - Maximum **interpolated** score: 387.08
  - Optimal parameters: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.982$
  - Exceeds actual experiment scores by 4.6%
- **Performance patterns:**
  - Steady improvement as α moves towards 1.0×10^{-3}
  - Best performance at $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.982$
  - Surface forms a clear peak, with decline in all directions from optimal point
  - Sensitivity between α and γ, shown by the peak and steep falloff around $\gamma = 0.982$

Surface Analysis for $\epsilon = 0.995$ *(figure 4)*

- **Peak Performance:**
  - Maximum interpolated score: 375.11
  - Optimal parameters: $\alpha = 3.2 \times 10^{-4}$, $\gamma = 0.999$
  - 3.1% lower than $\epsilon = 0.98$ peak
- **Performance patterns:**
  - Sharp decline when $\alpha > 3.2 \times 10^{-4}$
  - Steady falloff at $\gamma = 0.999$ for $\alpha < 3.2 \times 10^{-4}$
  - Clearly favoring higher γ values, peaking at 0.999

Middle Range Performance Analysis

- **Upper-Middle Tier (300-350 points):**
  - **Experiment 3:**
    - Score: 309.96
    - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.990$, $\epsilon = 0.995$
    - 16.2% below top performer
    - Key feature: Conservative learning rate with high discount
  - **Experiment 18**
    - Score: 306.44
    - Configuration: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.999$, $\epsilon = 0.980$
    - 17.2% below top performer
    - Key feature: Aggressive learning balanced by maximum discount
- **Lower-Middle Tier (250-300 points):**
  - **Experiment 1:**
    - Score: 297.78
    - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.950$, $\epsilon = 0.995$
    - 19.5% below top performer
    - Possible Limitation: Low discount factor *limiting* long-term planning
  - **Experiment 10:** 285.64
    - Score: 285.64
    - Configuration: $\alpha = 5.0 \times 10^{-4}$, $\gamma = 0.990$, $\epsilon = 0.980$
    - 22.8% below top performer
    - Possible Limitations:
      - Learning rate too high for middle range discount factor
      - Aggressive updates undermines moderate future planning

Poor Performance Analysis (< 200 points)

  - **Experiment 17:**
    - Score: 162.94
    - Configuration: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.999$, $\epsilon = 0.995$
    - 56.0% below top performer
    - Parameter analysis:
      - Same α succeeded with lower γ values (Exp 16)
      - Same γ succeeded with lower α values (Exp 11)
      - Suggests incompatibility between high α and high γ in this situation
  - **Experiment 14:**
    - Score: 172.90
    - Configuration: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.950$, $\epsilon = 0.980$
    - 53.3% below top performer
    - Parameter analysis:
      - Same α performed better with higher γ values (Exp 16)
      - Same γ performed better with lower α values (Exp 1)
      - Experiments with very low γ performed poorly

Setup 2

Identical to setup 1 **EXCEPT** sample batch size of $\mathbf{n = 64}$

Exp 1: $\alpha = 1.0e-04$, $\gamma = 0.950$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 2: $\alpha = 1.0e-04$, $\gamma = 0.950$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 3: $\alpha = 1.0e-04$, $\gamma = 0.990$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)
Exp 4: $\alpha = 1.0e-04$, $\gamma = 0.990$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 5: $\alpha = 1.0e-04$, $\gamma = 0.999$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 6: $\alpha = 1.0e-04$, $\gamma = 0.999$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)
Exp 7: $\alpha = 5.0e-04$, $\gamma = 0.950$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 8: $\alpha = 5.0e-04$, $\gamma = 0.950$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 9: $\alpha = 5.0e-04$, $\gamma = 0.990$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)
Exp 10: $\alpha = 5.0e-04$, $\gamma = 0.990$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 11: $\alpha = 5.0e-04$, $\gamma = 0.999$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 12: $\alpha = 5.0e-04$, $\gamma = 0.999$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)
Exp 13: $\alpha = 1.0e-03$, $\gamma = 0.950$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 14: $\alpha = 1.0e-03$, $\gamma = 0.950$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 15: $\alpha = 1.0e-03$, $\gamma = 0.990$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)
Exp 16: $\alpha = 1.0e-03$, $\gamma = 0.990$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)	Exp 17: $\alpha = 1.0e-03$, $\gamma = 0.999$, ε-decay $= 0.995$, $\tau = 1e{-}3$(fixed)	Exp 18: $\alpha = 1.0e-03$, $\gamma = 0.999$, ε-decay $= 0.980$, $\tau = 1e{-}3$(fixed)

Figure 5: Key for Setup 2 (same as Setup 1)

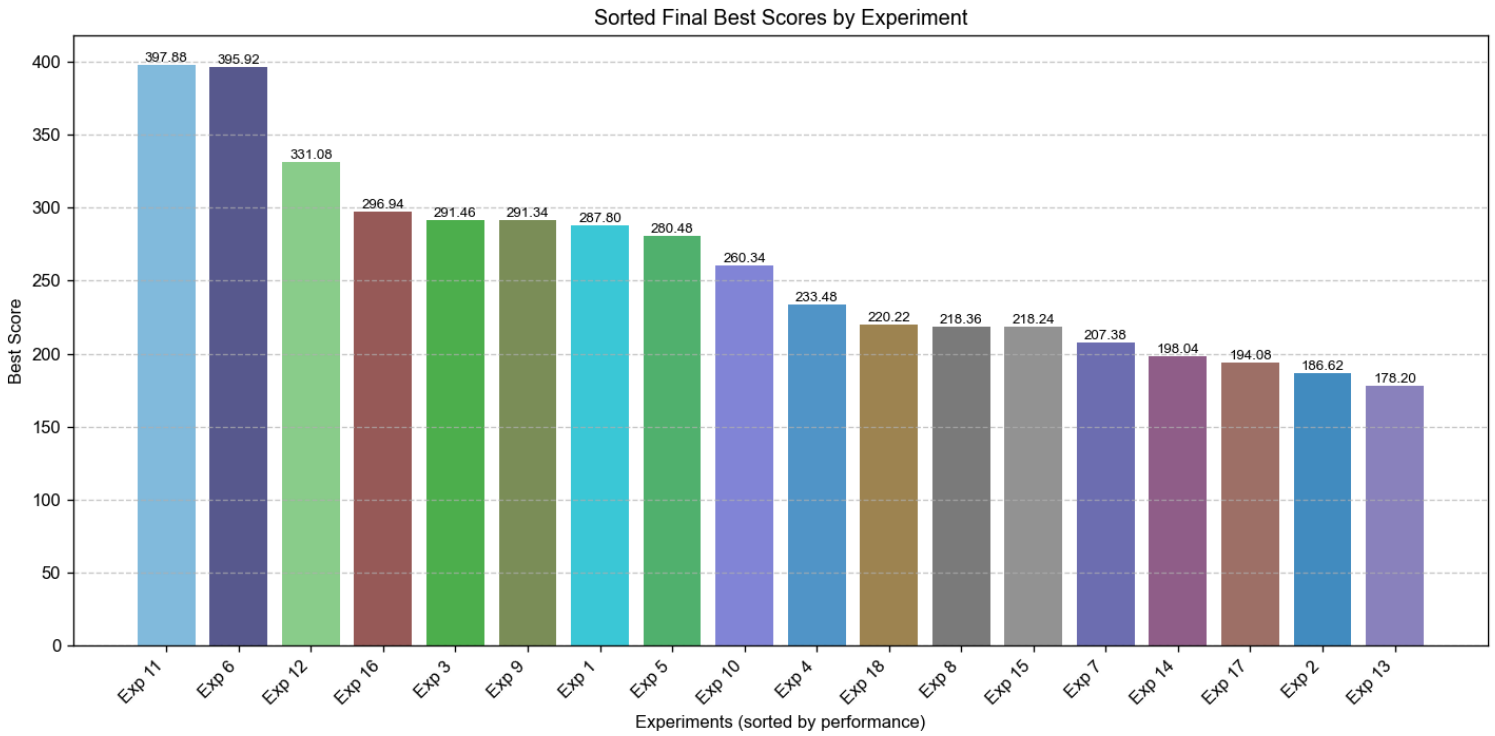

Figure 6

Learning Rate vs Discount Factor for $\epsilon - decay = 0.98$

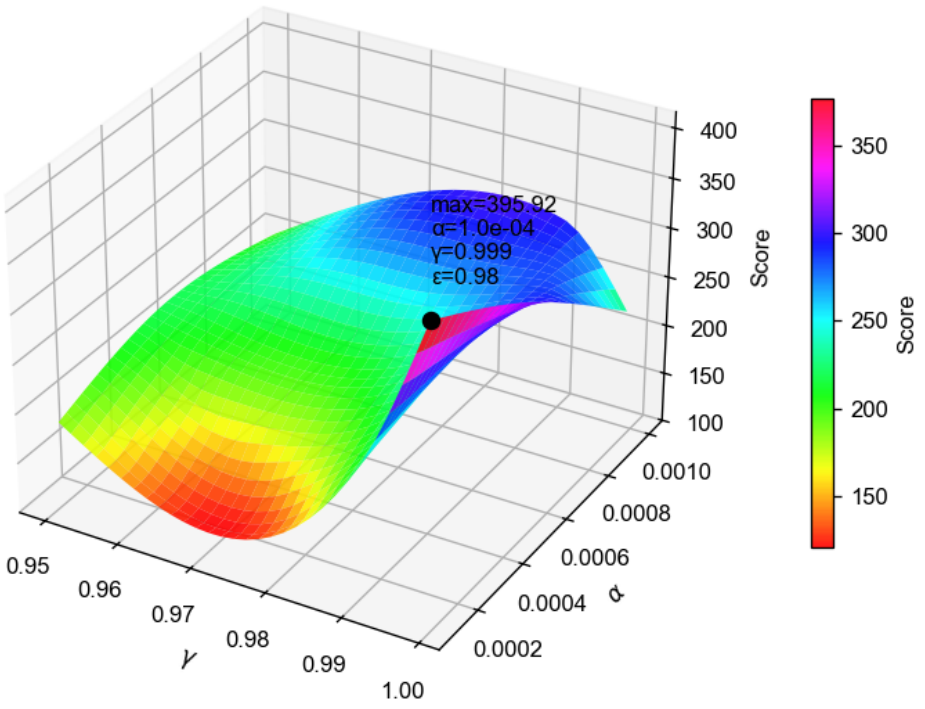

Figure 7

Learning Rate vs Discount Factor for $\epsilon - decay = 0.995$

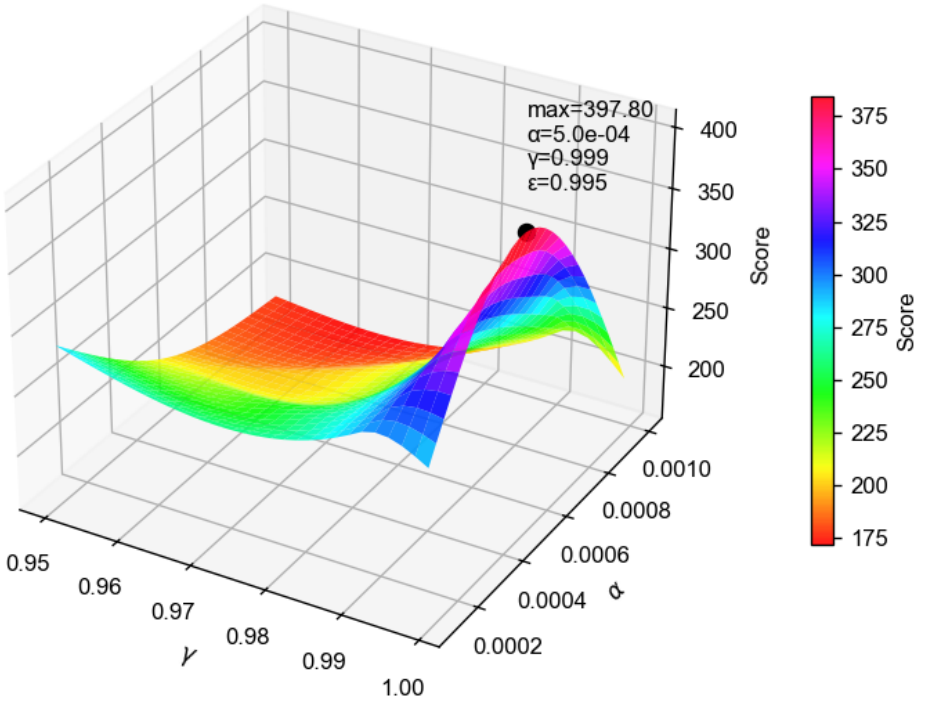

Figure 8

Performance Highlights: Hyperparameter Analysis for 64 Batch Size

Best Configurations

- **Highest Score - Experiment 11:**
  - Score: 397.88
  - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.995$
  - Characteristics: Mixed learning rate between aggression and stability, maximum discount, slower exploration
- **High Performers:**
  - **Experiment 6:**
    - Score: 395.92
    - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.98$
    - Characteristics: Conservative and stable learning, maximum discount, faster exploration
  - **Experiment 12:**
    - Score: 331.08
    - Configuration: $\alpha = 5.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.980$
    - Characteristics: Mixed learning rate between aggression and stability, maximum discount, faster exploration

Surface Analysis for $\epsilon = 0.98$ *(figure 7)*

- **Peak Performance:**
  - Maximum interpolated score: 395.92 (matches exactly with Experiment 6 performance)
  - Optimal parameters: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.999$
- **Performance patterns:**
  - Clear fall off higher alpha values of $\alpha > 5.0 \times 10^{-4}$
  - Peak performance region is narrow at the highest γ values
  - Surface has a clear peak with a steady descent path through high γ values with a more aggressive α. The path continues for lower γ at the most aggressive α.
  - Very sensitive to lowering γ near peak

Surface Analysis for $\epsilon = 0.995$ *(figure 8)*

- **Peak Performance:**
  - Maximum interpolated score: 397.80 (within 0.02% of Experiment 11)
  - Optimal parameters: $\alpha = 5.0 \times 10^{-4}$, $\gamma = 0.999$
- **Performance patterns:**
  - Highest scores all along high γ values, with a near flat plane for $\gamma < 0.99$
  - Extremely sensitive to α changes near the clearly optimal $\alpha = 5.0 \times 10^{-4}$
  - Clearly favors $\gamma > 0.99$
  - Steeper performance fall-offs than $\epsilon = 0.98$

Middle Range Performance Analysis

- **Upper-Middle Tier (290-330 points):**
  - **Experiment 16:**
    - Score: 296.94
    - Configuration: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.990$, $\epsilon = 0.980$
    - 25.4% below top performer
    - Key feature: Best performance with most aggressive learning rate
    - Performance comparisons:
      - Outperforms other higher α configurations
      - Displays stability with moderate discount factor when paired with aggressive learning
      - Shows that faster exploration can be viable
  - **Experiment 3:**
    - Score: 291.46
    - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.990$, $\epsilon = 0.995$
    - 26.7% below top performer
    - Key feature: Slow/steady learning with strong discount factor
    - Performance comparisons:
      - Same α had much higher scores when $\gamma = 0.999$ (Exp 11: 397.88)
      - Slower exploration outperformed similar configurations that had faster exploration (Exp 4: 233.48)
      - Shows viable success with slow/steady learning for slower exploration
      - Same γ value performed slightly better with more aggressive learning rates (Exp 16: 296.94)
- **Lower-Middle Tier (200-290 points):**
  - **Experiment 1:**
    - Score: 287.80
    - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.950$, $\epsilon = 0.995$
    - 27.7% below top performer
    - Key feature: Low discount factor limiting otherwise stable configuration
    - Parameter comparisons:
      - Similar α performed better with higher γ (Exp 11: 397.88, Exp 6: 395.92)
      - Low $\gamma = 0.950$ consistently performed poorly across almost all configurations (Exp 13: 178.20, Exp 2: 186.62)
      - Higher ϵ insufficient to overcome low γ
  - **Experiment 4:**
    - Score: 233.48
    - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.990$, $\epsilon = 0.980$
    - 41.3% below top performer
    - Key feature: Conservative learning undermined by faster exploration
    - Parameter comparisons:
      - Same α succeeded with higher γ (Exp 11: 397.88)
      - Same γ performed better with higher α (Exp 16: 296.94)
      - Fast exploration problematic with slow learning

Poor Performance Analysis (< 200 points)

- **Experiment 2:**
  - Score: 186.62
  - Configuration: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.950$, $\epsilon = 0.980$
  - 53.1% below top performer
  - Parameter analysis:
    - Same α succeeded with higher γ values - Exp 11: 397.88 ($\gamma = 0.999$), Exp 6: 395.92 ($\gamma = 0.999$), Exp 3: 291.46 ($\gamma = 0.990$)
    - Low $\gamma = 0.950$ produced poor results across configurations
    - Exploring quickly ($\epsilon = 0.980$) likely focuses too much on immediate rewards as γ is present-oriented which leads to failure.
- **Experiment 13:**
  - Score: 178.20
  - Configuration: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.950$, $\epsilon = 0.995$
  - 55.2% below top performer
  - Parameter analysis:
    - Same aggressive α achieved 296.94 with higher $\gamma = 0.990$ (Exp 16), showing aggression might be beneficial when paired with future-oriented planning as result of higher γ values
    - Aggressive α + present-oriented γ combination likely causes early exploitation and is completely misses more optimal policies.
    - Slower exploration ($\epsilon = 0.995$) does not help prevent near-sighted behavior
- **Experiment 17:**
  - Score: 194.08
  - Configuration: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.999$, $\epsilon = 0.995$
  - 51.2% below top performer
  - Parameter analysis:
    - Same high α performed better with lower γ (Exp 16: 296.94, $\gamma = 0.990$)
    - Same maximum γ achieved best scores with lower α (Exp 11: 397.88, Exp 6: 395.92)
    - Suggests maximally aggressive learning contradicts the pros of future-oriented planning
    - Even slower exploration ($\epsilon = 0.995$) was unable stabilize learning

Comparative Analysis

Top Performance Analysis

- **Maximum Score Comparison:**
  - Batch-32 maximum: 370.08 (Experiment 16)
  - Batch-64 maximum: 397.88 (Experiment 11)
  - Improvement: 27.80 points (+7.5%)
  - **Statistical Significance:** Interesting, when looking at the averages, the differences across the different experiments for Batch size 32 and 64 is statistically insignificant, which contradicts the improvement, with a two tailed p-value much greater than 0.05 (see Appendix A).
- **Optimal Parameters:**
  - Batch-32 optimal: $\alpha = 1.0 \times 10^{-3}$, $\gamma = 0.990$, $\epsilon = 0.98$
  - Batch-64 optimal: $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.995$
  - Key differences:
    - 10x difference in optimal learning rate
    - Higher discount factor always preferred... $\gamma > 0.990$
    - Faster exploration decay $\epsilon = 0.98$ was less sensitive to parameter shifts compared to slower exploration $\epsilon = 0.995$

Parameter Sensitivity Comparison

Sensitivity Coefficient Definition: The sensitivity coefficient quantifies the change in the parameter (either α or γ). It is calculated as the ratio of the difference in performance to the difference in the parameter between two points.

- **Learning Rate (α) Sensitivity** (see Appendix B):
  - **Batch-32:**
    - $\alpha = 1.0 \times 10^{-4}$, Average Score: 289.67
    - $\alpha = 5.0 \times 10^{-4}$, Average Score: 269.45
    - $\alpha = 1.0 \times 10^{-3}$, Average Score: 234.59
    - **Average Sensitivity Coefficient:** $-60,135$ points per $\Delta\alpha$
    - **Interval 1(1.0×10^{-4} to 5.0×10^{-4}) Sensitivity Coefficient:** $-50,550$ points per $\Delta\alpha$
    - **Interval 2(5.0×10^{-4} to 1.0×10^{-3}) Sensitivity Coefficient:** $-69,720$ points per $\Delta\alpha$
  - **Batch-64:**
    - $\alpha = 1.0 \times 10^{-4}$, Average Score: 279.29
    - $\alpha = 5.0 \times 10^{-4}$, Average Score: 284.39
    - $\alpha = 1.0 \times 10^{-3}$, Average Score: 217.62
    - **Average Sensitivity Coefficient:** $-60,395$ points per $\Delta\alpha$
    - **Interval 1(1.0×10^{-4} to 5.0×10^{-4}) Sensitivity Coefficient:** $+12,750$ points per $\Delta\alpha$
    - **Interval 2(5.0×10^{-4} to 1.0×10^{-3}) Sensitivity Coefficient:** $-133,540$ points per $\Delta\alpha$
  - α **Sensitivity Conclusions:**
    - While the average sensitivity coefficients for both batch sizes are nearly identical, the individual intervals for Batch-64 highlight its higher sensitivity.
    - Batch-32's has a consistent negative sensitivity coefficients implying it is more stable/predictable, making it less prone to major unexpected changes over varying learning rates.
    - Batch-64's sensitivity highlights the need for more careful tuning depending on the current interval, as stability over interval 1 is starkly contrasted with the huge instability in interval 2.

NOTE: It is important to recognize that sensitive does not directly equate to bad scores, but instead shows where volatility might be expected.

- **Discount Factor (γ) Sensitivity** (see Appendix C):
- **Batch-32:**
  - $\gamma = 0.999$, Average Score: 283.91
  - $\gamma = 0.990$, Average Score: 282.61
  - $\gamma = 0.950$, Average Score: 227.20
  - **Average Sensitivity Coefficient:** $+761.85$ points per $\Delta\gamma$
  - **Interval 1**$(0.999$ **to** $0.990)$ **Sensitivity Coefficient:** $+144.44$ points per $\Delta\gamma$
  - **Interval 2**$(0.990$ **to** $0.950)$ **Sensitivity Coefficient:** $+1,385.25$ points per $\Delta\gamma$
- **Batch-64:**
  - $\gamma = 0.999$, Average Score: 303.28
  - $\gamma = 0.990$, Average Score: 265.30
  - $\gamma = 0.950$, Average Score: 212.73
  - **Average Sensitivity Coefficient:** $+2,767.13$ points per $\Delta\gamma$
  - **Interval 1**$(0.999$ **to** $0.990)$ **Sensitivity Coefficient:** $+4,220.00$ points per $\Delta\gamma$
  - **Interval 2**$(0.990$ **to** $0.950)$ **Sensitivity Coefficient:** $+1,314.25$ points per $\Delta\gamma$
- γ **Sensitivity Conclusions:**
  - While the average sensitivity coefficients for both batch sizes indicate that their performances are reactive to changes in γ, Batch-64 displays a much higher sensitivity, especially in the initial interval, which aligns with the vary narrow viable range of high γ. The gradient is much smoother and indicates a more stable (yet worse) score for lower values of γ
  - Batch-32 has a more consistent and moderate sensitivity which suggests it is more stable/predictable with a smoother range of high scoring values for γ. The spike in the 2nd interval displays the potential sweet-spot range which utilizes a more mixed future and present reward bias.
  - Batch-64's high initial sensitivity underscores the need for more careful tuning of γ to optimize performance and find the most lucrative range.

NOTE: Again, it is important to recognize that sensitivity does not directly equate to bad scores, but instead indicates where performance volatility might be expected with parameter adjustments.

Pattern Analysis and Hypotheses

Key Performance Patterns

- **Learning Rate Impact:**
  - **Observation:** Batch-64 performs best with lower learning rates compared to batch-32.
  - **Hypothesis:** Bigger batches creates more reliable estimates by averaging over more examples
  - **Evidence:**
    - **Batch-64 Optimal Performances:**
      - Experiment 11 achieved 397.88 points with $\alpha = 1.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.995$
      - Experiment 6 reached 395.92 points with the same low learning rate.
      - The average score verifies this the benefit of a lower learning rate for Batch-64 as across all experiments with $\alpha = 1.0 \times 10^{-4}$ was 279.29, which drops dramatically to 217.62 when using $\alpha = 1.0 \times 10^{-3}$, showing larger batches are best suited for less aggressive learning
      - Batch-64's averages show high sensitivity, dropping sharply from 279.29 to 217.62 with aggressive learning rates. This increased sensitivity likely is result of the larger batch size, which provides more stable estimates. These more reliable gradients allow for more precise steps that can learn well by taking small and more thought out actions. Aggressive learning rates might overpower the averaging effect of the larger batch size which creates instability.
    - **Batch-32 Optimal Performances:**
      - Experiment 16 needed a much higher learning rate ($\alpha = 1.0 \times 10^{-3}$) to reach its peak of 370.08 points
      - Even when using the same low learning rate that worked for batch-64, Experiment 5 only reached 364.34 points
      - The averages display that batch-32 had a steady performance from 289.67 points at $\alpha = 1.0 \times 10^{-4}$ to 234.59 at $\alpha = 1.0 \times 10^{-3}$, showing it's more tolerant of different learning rates and benefited more from aggression compared to batch-64. This pattern might show as smaller batches create noisier estimates due to the smaller sample size. These noisier gradients might benefit from aggressive learning to try and ignore extra noise, making riskier, but faster and larger steps towards a final policy, rather than using a more relaxed learning rate which might get lost.

- **Discount Factor Impact:**
  - **Observation:** Both batch sizes achieved their highest performance with $\gamma = 0.999$, suggesting the future-oriented discount factor's impact applies to both batch sizes.
  - **Hypothesis:** The environment's reward structure might call for long-term planning, which makes the high discount crucial regardless of how training data is processed.
  - **Evidence:**
    - **High Gamma Performance:**
      - Batch-64 reached its peak of 397.88 points (Experiment 11) with $\gamma = 0.999$, while averaging 303.28 points across all $\gamma = 0.999$ experiments
      - Similarly, batch-32 achieved 370.08 points (Experiment 16) with $\gamma = 0.999$, with an average of 283.91 points for $\gamma = 0.999$ configurations
    - **Performance Degradation:**
      - Batch-64's average performance drops from 303.28 at $\gamma = 0.999$ to 265.30 at $\gamma = 0.990$, and plummets to 212.73 with $\gamma = 0.950$
      - Batch-32 is similar, averaging 283.91 at $\gamma = 0.999$, to 282.61 at $\gamma = 0.990$, and falling to 227.20 at $\gamma = 0.950$. This minute decline between 0.999 and 0.990 (1.3 points) compared to batch-64's drop (38 points) might be described by an interaction between batch size and discount factors. The smaller batch's noisier estimates might help by allowing for the agent to keep steady performance when it becomes slightly more near-sighted. This could mean that the noise in batch-32's might prevent it from becoming too dependent on precise future reward estimates.
      - This consistent fall off across both batch sizes suggests the environment requires long-term planing, as values near ($\gamma = 0.950$) consistently had the worst results.

Stability-Performance Trade-offs

- **Update Stability vs. Learning Speed:**
  - **Observation:** Batch-64 had higher maximum scores but was more sensitivity to parameter choices compared to batch-32.
  - **Hypothesis:** Larger batches allow for more precise learning due to better estimates, but is less tolerant to parameter configurations and might be unable to recover when the estimates are incorrect.
  - **Evidence:**
    - **Peak Performance Analysis:**
      - Batch-64's top experiments (11 and 6) reached record scores of 397.88 and 395.92
      - Small parameter changes led to dramatic drops: Experiment 17 scored only 194.08 when the learning rate became more aggressive.
      - The surface plots confirm this pattern, showing both clear peaks and very steep cliffs for batch-64 configurations
    - **Parameter Interaction Effects:**
      - Batch-64 requires more precise parameter selection, with its best results come only when using slow/steady learning rates (1.0×10^{-4}) with extremely high discount factors.
      - In contrast, batch-32 maintains good performance (300+ points) across more parameter combinations.

Failure Analysis

- **Worst Parameter Combinations:**
  - **Observation:** Some parameter combinations commonly lead to catastrophic failures, with performance dropping by more than 50% compared to the best configurations.
  - **Hypothesis:** The worst failures occur when parameter combinations create compounding negative effects.
  - **Evidence:**
    - **Worst Configurations for Batch-32:**
      - ($\alpha = 1.0 \times 10^{-3}$) with($\gamma = 0.999$)
      - Experiment 17 had this configuration with the slow exploration from $\epsilon = 0.995$and scored 162.94 points, 56% below the batch-32 peak
      - This combination might fail as the aggressive learning magnifies the noisy small batches, while high γ creates poor decisions as it is learning long-term from turbulent data. The slow exploration did not provide enough time to recover the deeply flawed policy
    - **Worst Configurations for Batch-64:**
      - The worst setup combines ($\alpha = 1.0 \times 10^{-3}$) with low discount factors ($\gamma = 0.950$) and fast exploration ($\epsilon = 0.98$)
      - Experiment 13 demonstrates this with only 178.20 points, 55.2% below batch-64's peak
      - This combination might fail as aggressive learning rates shadow the more precise averaging from the larger batch size.The low gamma promotes this near-sighted learning, and fast exploration prevents the model from fully utilizing its better gradient estimates.

Practicality

- **Parameter Selection:**
  - **Batch-32:**
    - Batch-32 is best with more aggressive learning from ($\alpha = 5 \times 10^{-4}$ to 1×10^{-3}) since its noisier gradients benefit from larger update steps. The peak performance of 370.08 was achieved with $\alpha = 1.0 \times 10^{-3}$.
    - Both fast ($\epsilon = 0.98$) and slow ($\epsilon = 0.995$) exploration decay rates produced good results.
    - More gradual performance changes makes it a good choice when parameter optimization is less feasible.
  - **Batch-64:**
    - Batch-64 was suited for slower/steadier learning rates between ($\alpha = 1 \times 10^{-4}$ to 5×10^{-4}). The two best scores (397.88 and 395.92) both had $\alpha = 1.0 \times 10^{-4}$. Rates above $\alpha = 5.0 \times 10^{-4}$ were not terrible within a narrow band of γ values, but performed much worse compared to the peaks.
    - Slower exploration decay ($\epsilon = 0.995$) gave batch-64 more time to learn from diverse experiences. Since batch-64 processes more examples at once, it is generally more reliable. Combined with the extended exploration period, this allowed the agent to build a good understanding of the environment. The highest score of 397.80 had a balanced configuration ($\alpha = 5.0 \times 10^{-4}$, $\gamma = 0.999$, $\epsilon = 0.995$), where the surface plots show smooth transitions around this peak, highlight an optimal and stable sweet spot in the parameter space.
    - The sharp cliffs in surface plots show the importance of parameter tuning Batch-64 can achieve higher peaks (397.88 vs 370.08), but requires a narrow set of parameters to achieve this.
- **Computation:**
  - Memory Requirements:
    - Batch-64: 2x memory per update.
    - Batch-32: More frequent updates needed.
    - Net computational cost similar.

A Statistical Significance Calculation

To determine whether the observed improvement in maximum scores from Batch Size 32 to Batch Size 64 is statistically significant, a two-tailed t-test was performed. Below is a step-by-step calculation using the actual dataset.

a. Sample Data

- **Batch Size 32 Scores**: 370.08, 364.34, 360.12, 309.96, 306.44, 300.18, 297.78, 285.64, 271.68, 261.78, 247.82, 232.48, 214.52, 208.40, 197.90, 197.30, 172.90, 162.94
- **Batch Size 64 Scores**: 397.88, 395.92, 331.08, 296.94, 291.46, 291.34, 287.80, 280.48, 260.34, 233.48, 220.22, 218.36, 218.24, 207.38, 198.04, 194.08, 186.62, 178.20

Sample Means ($\bar{X}$)

$\bar{X}_{32} = \simeq 264.57$
$\bar{X}_{64} = \simeq 260.44$

Sample Standard Deviations (σ)

$\sigma = \sqrt{\frac{\sum(X_i - \bar{X})^2}{n-1}}$

$\sigma_{32} = \simeq 64.89$
$\sigma_{64} = \simeq 66.80$

t-Statistic

$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{\sigma_{32}^2}{n_1} + \frac{\sigma_{64}^2}{n_2}}}$

Where:

$\bar{X}_1$, $\bar{X}_2$: sample means.

σ_{32}, σ_{64}:sample standard deviations.

n_1, n_2 are the sample sizes.

plugging in yields, $\boldsymbol{t = 0.1883}$

p-Value for 2-tailed test via GraphPad

For:

$t = 0.1883$

$df = 34$

two-tailed p-value = 0.8518

Conclusions

- **Critical Value:** The critical t-value is 2.032 for a significance level of $\alpha = 0.05$ with $df = 34$.
- **Hypothesis Testing:** We fail to reject the null hypothesis since the calculated t-statistic (0.1883) is much less than the critical value ($0.1883 \ll 2.032$).
- **p-Value Analysis:** The p-value (0.8518) is significantly greater than the significance level ($0.8518 \gg 0.05$), indicating that the results are not statistically significant.
- **Interpretation of Results:** The negligible difference in average scores between Batch Size 32 and Batch Size 64 suggests that any variations are due to random chance. This finding is interesting as, despite individual parameter sets performing differently across the two batch sizes, their averages remain similar. This consistency indicates that our selected range of parameters was most likely appropriate and effective.

Visualizing the Overlayed Normal Distribution For Similar Average Clarity

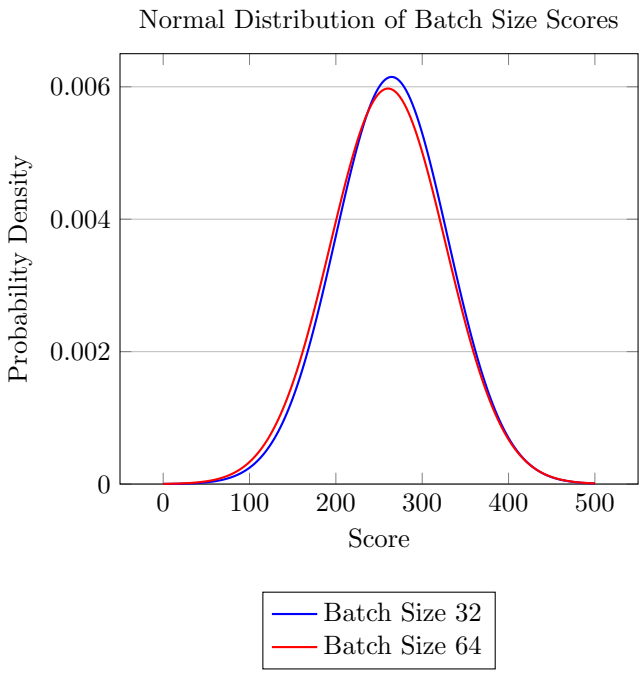

Figure 1A: Normal Distribution Curves for Batch Size 32 and 64

B Sensitivity Coefficient Calculation for Learning Rate α

Sensitivity Coefficient for Batch = 32:

- Interval 1 (1.0×10^{-4} to 5.0×10^{-4}):

$$\frac{269.45 - 289.67}{5.0 \times 10^{-4} - 1.0 \times 10^{-4}} = -50,550 \text{ points per } \Delta\alpha$$

- Interval 2 (5.0×10^{-4} to 1.0×10^{-3}):

$$\frac{234.59 - 269.45}{1.0 \times 10^{-3} - 5.0 \times 10^{-4}} = -69,720 \text{ points per } \Delta\alpha$$

Average Sensitivity Coefficient:

$$\frac{-50,550 + (-69,720)}{2} = -60,135 \text{ points per } \Delta\alpha$$

Sensitivity Coefficient for Batch = 64:

- Interval 1 (1.0×10^{-4} to 5.0×10^{-4}):

$$\frac{284.39 - 279.29}{5.0 \times 10^{-4} - 1.0 \times 10^{-4}} = +12,750 \text{ points per } \Delta\alpha$$

- Interval 2 (5.0×10^{-4} to 1.0×10^{-3}):

$$\frac{217.62 - 284.39}{1.0 \times 10^{-3} - 5.0 \times 10^{-4}} = -133,540 \text{ points per } \Delta\alpha$$

Average Sensitivity Coefficient:

$$\frac{+12,750 + (-133,540)}{2} = -60,395 \text{ points per } \Delta\alpha$$

Visualizing The Sensitivity Coefficient

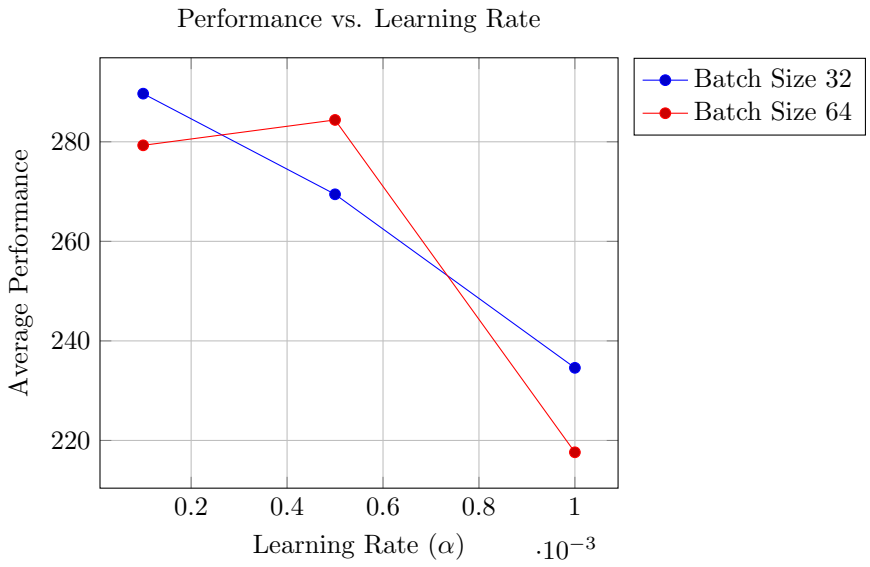

Figure 1B: Average Performance vs. Learning Rate for Batch Sizes 32 and 64

C Sensitivity Coefficient Calculation for Discount Factor (γ)

Sensitivity Coefficient for Batch = 32:

- **Interval 1** (0.999 **to** 0.990)**:**

$$\frac{282.61 - 283.91}{0.990 - 0.999} = \frac{-1.30}{-0.009} = +144.44 \text{ points per } \Delta\gamma$$

- **Interval 2** (0.990 **to** 0.950)**:**

$$\frac{227.20 - 282.61}{0.950 - 0.990} = \frac{-55.41}{-0.040} = +1,385.25 \text{ points per } \Delta\gamma$$

Average Sensitivity Coefficient:

$$\frac{+144.44 + (+1,385.25)}{2} = +761.85 \text{ points per } \Delta\gamma$$

Sensitivity Coefficient for Batch = 64:

- **Interval 1** (0.999 **to** 0.990)**:**

$$\frac{265.30 - 303.28}{0.990 - 0.999} = \frac{-37.98}{-0.009} = +4,220.00 \text{ points per } \Delta\gamma$$

- **Interval 2** (0.990 **to** 0.950)**:**

$$\frac{212.73 - 265.30}{0.950 - 0.990} = \frac{-52.57}{-0.040} = +1,314.25 \text{ points per } \Delta\gamma$$

Average Sensitivity Coefficient:

$$\frac{+4,220.00 + (+1,314.25)}{2} = +2,767.13 \text{ points per } \Delta\gamma$$

Visualizing The Sensitivity Coefficient

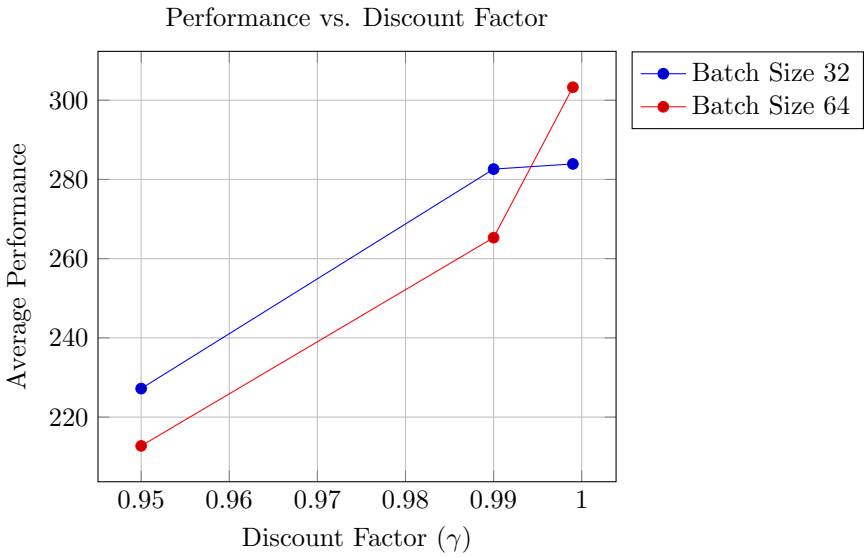

Figure 1C: Average Performance vs. Discount Factor for Batch Sizes 32 and 64