1 The Tic-Tac-Toe Problem: Controlling Player Decision

The classical tic-tac-toe game is a two-player, deterministic, turn-based game in which the player and opponent alternately place their respective tokens X and O on a 3×3 grid. The objective of the player is to be the first to align three of their tokens consecutively in a row, column, or diagonal, and likewise for the opponent.

1.1 Encoding

Mathematically, the 3×3 game board is represented by a matrix $\mathbf{M}_{3\times 3}$, where an X token is encoded as -1, an O token as $+1$, and empty cells as 0. For convenience, the matrix $\mathbf{M}$ is vectorized by row into a single 9-dimensional vector $\mathbf{m}$, thereby providing a compact representation of the board state. In the configuration considered, the player is assumed to make the first move, assigned as the O player.

Furthermore, we define a game of tic-tac-toe to be complete at time k, where the player places the O token on the grid at discrete times $\tau_k^{(p)} = 1, \ldots, T_k^{(p)} \leq 5$ (since the player moves first), and the opponent places the X token on the grid at discrete times $\tau_k^{(o)} = 1, \ldots, T_k^{(o)} \leq 4$, while the overall time index for the k^{th} game is given by $\tau_k = 1, \ldots, T_k \leq 9$. Hence $\mathbf{m}^{\tau_k - 1}$ would be the current board state before the player or opponent places their token at time τ_k. Furthermore, we define the set of opponent's decisions/actions at the conclusion of the k^{th} game in sequence $\mathcal{O}_k = \left[a_{\tau_k^{(o)}=1}, \ldots, a_{\tau_k^{(o)}=T_k^{(o)}}\right]'$ for any action $a \in \{1, 2 \ldots, 9\}$ corresponding to available board positions. Furthermore, we define the set of opponent's actions for K games as matrix $\boldsymbol{\mathcal{O}}_K = \left[\mathcal{O}_1, \ldots, \mathcal{O}_K\right]$.

1.2 The game's outcome

At each turn of the game, the current state must be evaluated to determine whether the game has reached a terminal condition - namely, a win, loss, or draw - or whether play should continue. This involves computing the sums of each row, column, and diagonal of $\mathbf{M}$ to check for a winning configuration. Specifically, a sum of -3 or 3 indicates a win for the X or O player, respectively.

We may evaluate the game state by transposing and post-multiplying $\mathbf{m}$ by a state-matrix:

$$\mathbf{S} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}_{9\times 8}$$

hence, if any one of the 8 entries in the game-state evaluation $\mathbf{m}'\mathbf{S}$ is equal to -3, X has won. Likewise, if any of the entries is equal to $+3$, O has won.

To assess whether a draw has occurred, it is necessary to verify that each of the three rows, three columns, and both diagonals of $\mathbf{M}$ contains at least one X and one O. This ensures that no player can achieve a winning alignment in any direction. To do this, let $\mathbf{m}_+ = [\mathbb{I}(m_i = +1)]_{i=1}^9$ where $\mathbb{I}(\cdot)$ is the indicator function, applied element-wise to the entries of $\mathbf{m}$. Hence, $\mathbf{m}_+$ is a 9-dimensional binary vector with ones at positions where the corresponding entries of $\mathbf{m}$ are equal to $+1$, and zeros elsewhere. Similarly, we define the 9-dimensional binary vector $\mathbf{m}_-$ analogously to indicate the positions of entries equal to -1 in $\mathbf{m}$. A draw has then occured if $(\mathbf{v}_+)'_{1\times 8}(\mathbf{v}_-)_{8\times 1} = 8$ where 8-dimensional vector $\mathbf{v}_+ = \left[\mathbb{I}\left(\left(\mathbf{m}_+'\mathbf{S}\right)_i > 0\right)\right]_{i=1}^8$ and $\mathbf{v}_- = \left[\mathbb{I}\left(\left(\mathbf{m}_-'\mathbf{S}\right)_i > 0\right)\right]_{i=1}^8$.

We denote ρ_{τ_k} to denote the value of the winning token or zero (for a draw) if the game is terminal after the player or opponent has placed their token at time $\tau_k = 1, 2 \ldots, T_k$ for the k^{th} tic-tac-toe game. Hence:

$$\rho_{\tau_k} = \begin{cases} +1, & \text{if } \exists\, i \in \{1, \ldots, 8\} \text{ such that } \left((\mathbf{m}^{\tau_k})'\mathbf{S}\right)_i = +3, \\ -1, & \text{if } \exists\, i \in \{1, \ldots, 8\} \text{ such that } \left((\mathbf{m}^{\tau_k})'\mathbf{S}\right)_i = -3, \\ 0, & \text{if } \left(\mathbf{v}_+^{\tau_k}\right)'\left(\mathbf{v}_-^{\tau_k}\right) = 8, \\ \text{NULL}, & \text{if game is not terminal.} \end{cases}$$

1.3 Control

We control player decisions/actions at time $\tau_k^{(p)} = 1, \ldots, T_k^{(p)} \leq 5$ for the k^{th} tic-tac-toe game through the means of the control vector $\mathbf{ct}\left(\mathbf{m}^{\tau_k - 1}, \boldsymbol{\theta}^{\text{Decision}}\right) \in \mathcal{A}_{\tau_k} \subseteq \{1, 2 \ldots, 9\}$, where $\mathcal{A}_{\tau_k}$ represents the subset of available board positions on $\mathbf{m}^{\boldsymbol{\tau_k - 1}}$ before the player opts to play at time τ_k for some parameter configuration $\boldsymbol{\theta}^{\text{Decision}} \in \mathbb{R}^R$ - naturally, once a grid position is occupied by a token, it becomes unavailable for subsequent moves by either the player or the opponent. Hence $\mathcal{A}_{\tau_k} = \left\{i \in \{1, \ldots, 9\} : m_i^{\tau_k - 1} = 0\right\}$. The player action selection is probabilistic and derived from a softmax distribution over logits. Hence, for ℓ_a being the logit score

for any valid action $a \in \mathcal{A}_{\tau_k}$, the probability of selecting that action is:

$$\sigma_L(a \mid \mathbf{m}^{\tau_k-1}, \boldsymbol{\theta}^{\text{Decision}}) = \frac{\exp(\ell_a)}{\sum_{a' \in \mathcal{A}_{\tau_k}} \exp(\ell_{a'})}.$$

The selected action a^* corresponds to the action with the highest probability, that is, $a^* = \text{argmax}_{a \in \mathcal{A}_{\tau_k}} \sigma_L(a \mid \mathbf{m}^{\tau_k-1}, \boldsymbol{\theta}^{\text{Decision}})$. Invalid actions (i.e., $a \notin \mathcal{A}_{\tau_k}$) are assigned $\ell_a = -\infty$, ensuring a zero probability is attributed to that specific invalid action. Now the interface between a model and the player action is undergone through this control vector for which $\mathbf{ct} : (\mathbf{m}^{\tau_k-1}, \boldsymbol{\theta}^{\text{Decision}}) \to \mathbf{model}\left(\boldsymbol{\Omega}\left(\mathbf{m}^{\tau_k-1}\right), \boldsymbol{\theta}^{\text{Decision}}\right) \xrightarrow{\sigma_L(.)} a^* \in \mathcal{A}_{\tau_k}$ where $\boldsymbol{\theta}^{\text{Decision}}$ is fixed throughout all $k = 1, 2, \ldots, K$ tic-tac-toe games for all player turns at times $\tau_k^{(p)} = 1, \ldots, T_k^{(p)} \leq 5$, and all player decisions are based on this fixed parameterization $\boldsymbol{\theta}^{\text{Decision}}$. In the framework of using a neural network as our model, we define $\boldsymbol{\Omega} : \mathbf{m}^{\tau_k-1} \to \mathbf{a}^0 \in \mathbb{R}^{d_0}$ which signifies the vector of input nodes for times $\tau_k^{(p)} = 1, \ldots, T_k^{(p)} \leq 5$. Furthermore, $\boldsymbol{\theta}^{\text{Decision}}$ are the weights and biases of the neural network, $\mathbf{w}^{\text{Decision}} \in \mathbb{R}^R$.

1.3.1 Feature engineering

We construct the feature vector using two inputs: the current board state before the player places a token at time τ_k, $\mathbf{m}^{\tau_k-1}$, as well as the game-state evaluation, $\left(\mathbf{m}^{\tau_k-1}\right)'\mathbf{S}$. The former encodes the spatial configuration of tokens on the board capturing positional information essential to the learning process. The latter provides a structured summary of token alignments across rows, columns, and diagonals, serving as a higher-level representation that facilitates the identification of a win or loss of the player. Hence, our 1^{st} set of input nodes are defined as $\mathbf{a}_1^0 = \mathbf{m}^{\tau_k-1}$ which represents the board state at time $\tau_k - 1$ and the 2^{nd} set of input nodes is given by $\mathbf{a}_2^0 = \left(\mathbf{m}^{\tau_k-1}\right)'\mathbf{S}$ which denotes the game-state evaluation at time $\tau_k - 1$.

1.4 The arbitrary objective

Consider an arbitrary objective where, for a given parameter configuration $\boldsymbol{\theta} = \boldsymbol{\theta}^{\text{Decision}} \in \mathbb{R}^R$ and after playing K number of tic-tac-toe games, we count the number of times the player's token O (encoded as $+1$) won the game, denoted as $\sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right) \in \{0, 1, \ldots K\}$. Hence:

$$\underset{\boldsymbol{\theta}}{\text{argmax}}\, \text{Obj}\left(\boldsymbol{\theta}\right) = \underset{\boldsymbol{\theta}}{\text{argmax}}\, \frac{1}{K} \sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right).$$

By including L2 regularization, our L2 penalized objective becomes:

$$\text{Obj} = \underset{\boldsymbol{\theta}}{\text{argmax}} \left(\frac{1}{K} \sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right) - \nu \|\boldsymbol{\theta}\|^2 \right). \tag{1}$$

Now congruent to Section **??**, $\text{argmax}_{\boldsymbol{\theta}}\, p(\boldsymbol{\theta}|\mathcal{D})$ must be equivalent to maximising the objective function in Equation 1. This is achieved by ensuring the likelihood $p(\mathcal{D} \mid \boldsymbol{\theta})$ is monotonic increasing with respect to $\sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right)$, that is, $p(\mathcal{D} \mid \boldsymbol{\theta}) \propto \sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right)$.

1.4.1 Exponential-based likelihood

Section **??** elucidated that when MCMC is employed primarily as a mode-seeking algorithm - that is, the mode of the conditional $p(\boldsymbol{\theta} \mid \sigma_\theta^2, \mathcal{D})$ - rather than for full Bayesian inference, the necessity of an explicit and well-defined likelihood function linking the parameters $\boldsymbol{\theta}$ to the observed data becomes less critical. In such settings, it suffices to employ any monotonically increasing transformation of the objective function to guide the proposal mechanism of the MH algorithm, thereby biasing the random walk toward regions of high-likelihood (high-valued objective) regions to sample around a dominant mode of the conditional.

Accordingly, we adopt an exponential transformation as the chosen monotonic function, serving as a surrogate for the traditional likelihood, to facilitate efficient exploration of high-valued objective regions in the parameter space. Hence for $\boldsymbol{\theta} \in \mathbb{R}^R$, $\sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right) \in \{0, 1, \ldots, K\}$ hence $\frac{1}{K}\sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right) \in [0, 1]$ and sharpness $\beta \in \mathbb{R}^+$, we have our new likelihood as:

$$p(\mathcal{D} \mid \boldsymbol{\theta}) = \exp\left(\beta \cdot \frac{1}{K} \sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right) \right) \tag{2}$$

with the log of conditional posterior being, noting the prior $\boldsymbol{\theta} \mid \sigma_\theta^2 \sim \mathcal{N}(\mathbf{0}, \sigma_\theta^2 \mathbf{I}_S)$:

$$\begin{aligned} \log\left(p(\boldsymbol{\theta} \mid \sigma_\theta^2, \mathcal{D})\right) &\propto \log\left(p(\mathcal{D} \mid \boldsymbol{\theta})\right) + \log\left(p(\boldsymbol{\theta} \mid \sigma_\theta^2)\right) \\ &\propto \beta \cdot \frac{1}{K} \sum_{k=1}^K \mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right) = +1\right) - \frac{1}{2\sigma_\theta^2}\|\boldsymbol{\theta}\|^2 - \frac{S}{2}\log\left(2\pi\sigma_\theta^2\right). \end{aligned} \tag{3}$$

1.5 Effects of regularization

To train the tic-tac-toe agent, we simulate $K = 100$ tic-tac-toe games in which the model learns to play against an opponent whose behavior is governed by a random decision policy. Specifically, the opponent selects among the available (i.e., unoccupied) grid positions uniformly at random when placing its X token. As a result, the trajectory of each game - and by extension, the opponent's decision-making process - is contingent on a random seed for which we use seed values $\{\omega_i^{\text{Train}}\}_{i=1}^{100}$ corresponding to the $K = 100$ tic-tac-toe games. Now since the player's decisions - determined by the solution $\hat{\boldsymbol{\theta}}_\nu$ - directly influences the set of random opponent decision sequences $\boldsymbol{\mathcal{O}}_{100}^{\text{Train}}$ for the $K = 100$ games (as the opponent may only place their X token on unoccupied grid positions), we cannot assume that the set of random opponent decision sequences remains fixed across all solutions. For example, the solution $\hat{\boldsymbol{\theta}}_{\nu_1}$ will likely induce a different $\boldsymbol{\mathcal{O}}_{100}^{\text{Train}}$ than $\hat{\boldsymbol{\theta}}_{\nu_2}$ for $\nu_1 \neq \nu_2$, even when each of the $k = 1, \ldots, K$ tic-tac-toe games are initialized with the same seed. Consequently, the best we can do to ensure that the random opponent's behavior is both deterministic and reproducible across different solutions, is to control it via a fixed seed, in an attempt to allow for controlled evaluation and consistent comparison of the agent's performance across training runs. We may extend this notion further by observing that during the optimisation process, each candidate solution dictates the set of random opponent decision sequences. As a result, the effective optimisation surface to be maximised is not fixed but changes across iterations under this framework.

Furthermore, we define the test set as consisting of $K = 10{,}000$ simulated tic-tac-toe games, each initialized by a distinct seed value ω_i^{Test} such that $\omega_i^{\text{Test}} \neq \omega_j^{\text{Train}}$ for all i, j. However, this condition alone does not ensure that all random opponent decision sequences in the test set, $\mathcal{O}_k^{\text{Test}}$ for $k \in \{1, \ldots, 10000\}$, are disjoint from those encountered during training, $\mathcal{O}_k^{\text{Train}}$ for $k \in \{1, \ldots, 100\}$. Additionally, to prevent the inflation or degradation of performance due to repeated identical random opponent decision sequences in the test set, we enforce that $\mathcal{O}_k^{\text{Test}} \neq \mathcal{O}_l^{\text{Test}}$ for all $k, l \in \{1, \ldots, 10000\}$ with $k \neq l$ - that is, all random opponent decision sequences used during testing are mutually distinct. Hence, to guarantee that all test games are genuinely out-of-sample (and unique), we iteratively cycle through candidate seed values ω_j^{Test} until we obtain a collection of $K = 10{,}000$ test games whose random opponent decision sequences are distinct from those observed in the $K = 100$ training games (as well as being distinct from each other). Hence $j \geq 10,000$ for our out-of-sample test seed values ω_j^{Test}. In doing so, we ensure that the training and test environments are disjoint, thereby enabling a valid assessment of the agent's generalization performance to previously unseen opponent behaviors. As before, however, since the random opponent decision sequences are governed by the player's decisions - controlled by $\hat{\boldsymbol{\theta}}_\nu$ - each $\hat{\boldsymbol{\theta}}_\nu$ would give rise to a different set of random opponent decision sequences $\boldsymbol{\mathcal{O}}_{10,000}^{\text{Test}}$, hence rendering the $K = 10,000$ out-of-sample games to be somewhat different across solutions derived.

To evaluate the impact of the regularization strength ν, we apply the estimator $\hat{\boldsymbol{\theta}}_\nu^{\text{GA}}$ to the test set and assess both in-sample and out-of-sample performance across a range of ν values, as reported in Table 2. We denote by $\hat{\boldsymbol{\theta}}_\nu^{\text{GA},(\text{I})}$ the solution obtained from $\overset{(\text{I})}{\textbf{model}}$, which solely uses $\mathbf{a}_1^0$ as its feature vector. Likewise, $\hat{\boldsymbol{\theta}}_\nu^{\text{GA},(\text{II})}$ corresponds to $\overset{(\text{II})}{\textbf{model}}$, which incorporates the full feature vector $\mathbf{a}_{(9+8)\times 1}^0 = \left[\left(\mathbf{a}_1^0\right)', \left(\mathbf{a}_2^0\right)'\right]'$. Furthermore, we also apply the estimator $\hat{\boldsymbol{\theta}}_\nu^{\text{RS}}$ on the test set in order to establish a baseline against which the performance of the GA can be compared.

As shown in Table 1, there is a clear trend of decreasing performance, both in-sample and out-of-sample, as the regularization strength increases for both models. This behavior is consistent with underfitting due to excessive regularization - that is, the model becomes overly constrained. Moreover, the results suggest that the use of a GA is necessary to achieve improved performance on the in-sample set, as it consistently outperforms RS at low values of ν. However, this pattern does not persist across all regularization strengths; at high regularization levels, RS appears to yield better in-sample solutions than the GA at times. This observation implies that the fine-tuning capability of the GA is most beneficial when the model is not heavily constrained - that is, under such conditions, the GA's exploitation properties appear to play a critical role in refining existing parent solutions. Finally, we conjecture that the best out-of-sample performance may be attained at regularization strengths corresponding to $\nu \in [10^{-6}, 10^{-4}]$, where the model appears to strike an effective balance between in-sample performance and out-of-sample generalization.

Additionally, we observe, that in the absence of regularization ($\nu = 0$), $\overset{(\text{II})}{\textbf{model}}$ slightly outperforms $\overset{(\text{I})}{\textbf{model}}$ in both the in-sample and out-of-sample sets. Nonetheless, we refrain from making general claims regarding comparative performance across varying values of ν between the two models, as the two models differ in complexity with respect to their number of input nodes - $\overset{(\text{I})}{\textbf{model}}$ utilizing $\mathbf{a}_1^0$ as its feature vector and $\overset{(\text{II})}{\textbf{model}}$ utilizing $\mathbf{a}^0$ as its feature vector. Consequently, a given value of ν cannot be interpreted as exerting the same regularization effect (capturing the goal of making a model less complex) across both models, and direct comparisons of regularization magnitudes should be treated with caution.

Importantly, Table 1 also demonstrates that both models significantly outperform a baseline agent governed by purely random decision-making in cases where the model is not overly constrained - that is, when ν is not excessively large. This baseline agent - playing as the first mover - also faces a random opponent for $K = 10,000$ tic-tac-toe games all of which have distinct random opponent decision sequences ($\mathcal{O}_k \neq \mathcal{O}_l$ for all $k, l \in \{1, \ldots, 10000\}$ with $k \neq l$). The random baseline achieves a normalized win percentage of only 57.16%, which is consistently exceeded by the learned agents under moderate regularization strengths.

Now to investigate the local sensitivity of model performance to small variations in regularization strength, Table 2 reports in-sample and out-of-sample performance across finely spaced values of ν for both $\overset{(\text{I})}{\textbf{model}}$ and $\overset{(\text{II})}{\textbf{model}}$. The results reveal non-monotonic behavior with respect to ν indicating that small increases in regularization do not uniformly lead to performance degradation, and in some cases, produce unexpected improvements - particularly in out-of-sample performance.

	$\overset{(\mathrm{I})}{\mathbf{model}}\left(\mathbf{a}_1^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{GA},(\mathrm{I})}\right)$		$\overset{(\mathrm{II})}{\mathbf{model}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{GA},(\mathrm{II})}\right)$		$\overset{(\mathrm{II})}{\mathbf{model}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{RS},(\mathrm{II})}\right)$	
ν	In-Sample	Out-of-Sample	In-Sample	Out-of-Sample	In-Sample	Out-of-Sample
0.000001	97	61.72	99	83.92	94	67.38
0.00001	99	73.78	99	74.84	96	83.16
0.0001	97	82.96	98	66.96	97	62.98
0.001	92	76.28	93	59.96	68	59.46
0.01	82	77.58	85	59.36	45	47.64
0.1	57	45.92	74	69.10	77	46.92
1	44	47.82	52	46.04	67	54.10

Table 1: The normalized number of O wins, as a percentage $\left(\frac{100}{K}\sum_{k=1}^{K}\mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right)=+1\right)\right)$ for in-sample ($K=100$) and out-of-sample ($K=10,000$) sets across regularization strengths ν using $\overset{(\mathrm{II})}{\mathbf{model}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{GA},(\mathrm{II})}\right)$ and $\overset{(\mathrm{II})}{\mathbf{model}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{RS},(\mathrm{II})}\right)$.

	$\overset{(\mathrm{I})}{\mathbf{model}}\left(\mathbf{a}_1^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{GA},(\mathrm{I})}\right)$		$\overset{(\mathrm{II})}{\mathbf{model}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{GA},(\mathrm{II})}\right)$	
ν	In-Sample	Out-of-Sample	In-Sample	Out-of-Sample
0.0000	98	73.62	99	77.86
0.0001	97	82.96	98	66.96
0.0002	94	64.68	97	76.82
0.0003	98	65.32	95	77.46
0.0004	96	77.20	97	73.72
0.0005	95	78.58	95	73.74
0.0006	97	63.00	95	79.50
0.0007	95	58.96	95	72.86
0.0008	95	73.20	91	62.00
0.0009	96	73.74	94	75.40

Table 2: The normalized number of O wins, as a percentage $\left(\frac{100}{K}\sum_{k=1}^{K}\mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right)=+1\right)\right)$ for in-sample ($K=100$) and out-of-sample ($K=10,000$) sets across regularization strengths ν using $\overset{(\mathrm{I})}{\mathbf{model}}\left(\mathbf{a}_1^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{GA},(\mathrm{I})}\right)$ and $\overset{(\mathrm{II})}{\mathbf{model}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\mathrm{GA},(\mathrm{II})}\right)$.

1.6 MCMC

We exclusively employ Equation 2 as the likelihood function in the MH algorithm presented in Equation **??**, in this section. Although one could reasonably argue for the use of alternative likelihoods - such as the binomial and beta-based forms given in Equations **??** and **??** respectively - we refrain from doing so here as a comprehensive comparison among these three likelihood formulations has already been conducted in Section **??**. The present section is dedicated solely to illustrating how increasing the sharpness of the likelihood (through the parameter $\beta \in \mathbb{R}^+$) may influence the results. Furthermore, we employ $\overset{(\mathrm{II})}{\mathbf{model}}$ in this section.

Illustrated in Table 3 are the normalized number of O wins, expressed as a percentage $\left(\frac{100}{K}\sum_{k=1}^{K}\mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right)=+1\right)\right)$, computed over 100,000 total MCMC iterations, of which the first 20,000 were discarded as burn-in. We observe a clear trend: as the sharpness parameter $\beta \in \mathbb{R}^+$ increases, the proportion of in-sample O wins tends to improve. Section **??** alluded to this phenomemon - the parameter β can be interpreted as a means of amplifying the likelihood ratio in Equation **??**, yielding the modified expression $\left(\frac{p(\mathcal{D}|\boldsymbol{\theta}^*)}{p(\mathcal{D}|\boldsymbol{\theta}^{(j)})}\right)^{\beta}$. Increasing β makes the Markov chain more inclined to accept proposed solutions $\boldsymbol{\theta}^*$ that yield higher objective values, given the proportionality $p(\mathcal{D} \mid \boldsymbol{\theta}) \propto \mathrm{Obj}(\boldsymbol{\theta})$ - in effect, making the MCMC sampler more likelihood-driven.

Furthermore, as elucidated in Section **??**, it was justified that the MCMC sampler should be made more likelihood-driven, as failure to do so could result in an indefinite contraction of $\boldsymbol{\theta}$ toward zero. This phenomenon is corroborated by Figure 1, which depicts the trajectory of $\|\boldsymbol{\theta}^{(j)}\|^2$ across the 80,000 post-burn-in iterations. For small values of β, we observe that the norm $\|\boldsymbol{\theta}^{(j)}\|^2$ exhibits a slow, monotonic decline across iterations - a strong indication that the Markov chain remains in its transient phase and has not yet reached stationarity. We posit that, this monotonic decay may reflect more than just delayed convergence. At low values of β, the pseudo-likelihood - and by extension the conditional posterior $p(\boldsymbol{\theta} \mid \sigma_\theta^2, \mathcal{D})$ in Equation 3 - becomes too diffuse to meaningfully constrain the parameter space. As a result, the data exerts minimal influence over the proposed solution $\boldsymbol{\theta}^{(j)}$, and the conditional posterior is effectively dominated by the prior $p(\boldsymbol{\theta} \mid \sigma_\theta^2)$. In this setting, the likelihood becomes inconsequential, and the sampling dynamics are driven almost entirely by the prior structure. Consequently, the MH acceptance probabilities α_θ favor proposals $\boldsymbol{\theta}^{(j)}$ that reduce the norm $\|\boldsymbol{\theta}^{(j)}\|^2$ - as is evident by Equation 3, where a reduction in $\|\boldsymbol{\theta}^{(j)}\|^2$ results in larger conditional posterior values thereby guiding the Markov chain to search in areas where low $\|\boldsymbol{\theta}^{(j)}\|^2$ values are obtained. This behavior is a direct reflection of what the conditional posterior - being flat and prior-dominated - is prescribing. The sampler is "doing its job": in the absence of strong likelihood information, the proposals $\boldsymbol{\theta}^{(j)}$ are simply contracting toward the origin under the influence of the Gaussian prior.

Additionally, the multivariate effective sample sizes (ESS) reported in Table 3 provide evidence of satisfactory mixing. All reported ESS values exceed the commonly accepted threshold of 100 (as recommended by Vehtari et al. (2021), Section 4), which supports the claim of efficient exploration. However, it is important to emphasize that ESS is a meaningful diagnostic only after convergence has been attained. In particular, for small values of β, we observe - via the continued drift in $\|\boldsymbol{\theta}^{(j)}\|^2$ - that the chain remains in a transient phase, and thus has not yet fully converged to its stationary distribution. Consequently, while we report ESS values at the end of the $80,000$ usable MCMC iterations for completeness, we interpret them with caution in the low-β regime.

Furthermore, for low values of β, where the MCMC samples exhibit non-stationary behavior in $\|\boldsymbol{\theta}^{(j)}\|^2$, the resulting marginal distribution of $\sigma_\theta^2 \mid \mathcal{D}$ deviates from an inverse-gamma form. Since the conditional posterior $\sigma_\theta^2 \mid \boldsymbol{\theta}, \mathcal{D} \sim \text{Inv-Gamma}\big(a+\frac{S}{2},\, b+\frac{\|\boldsymbol{\theta}\|^2}{2}\big)$ where $a, b \approx 0$, we know that if $\|\boldsymbol{\theta}^{(j)}\|^2$ fluctuates around some constant c, the marginal $\sigma_\theta^2 \mid \mathcal{D}$ should also be inverse-gamma distributed with constant shape and scale parameters. That is, if $\|\boldsymbol{\theta}^{(j)}\|^2 \approx c$, then $\sigma_\theta^2 \mid \mathcal{D} \sim \text{Inv-Gamma}\big(a+\frac{S}{2},\, b+\frac{c}{2}\big)$ as illustrated in Figure 1.

Now, Table 3 suggests that, in order to obtain solutions yielding in-sample performance comparable to that achieved by traditional optimisation methods such as the genetic algorithm in Section **??**, the likelihood sharpness parameter β must be increased to sufficiently concentrate the conditional posterior $p(\boldsymbol{\theta} \mid \mathcal{D}, \sigma_\theta^2)$. This ensures that the sampler is more decisively guided by the pseudo-likelihood - that is, made sufficiently likelihood-driven - resulting in proposed solutions $\boldsymbol{\theta}^{(j)}$ whose norms $\|\boldsymbol{\theta}^{(j)}\|^2$ stabilize across MCMC iterations. Such stabilisation indicates convergence toward a dominant mode of the conditional $p(\boldsymbol{\theta} \mid \sigma_\theta^2, \mathcal{D})$. Empirically, for the tic-tac-toe problem studied here, values of $\beta \geq 100$ appear to meet this threshold, yielding both stable posterior behavior and competitive in-sample performance. Furthermore, because the marginals of $\sigma_\theta^2 \mid \mathcal{D}$ are approximately inverse-gamma distributed with nearly identical shape and rate parameters (see the bottom panel of Figure 1) for solutions obtained above said threshold - this suggests that the corresponding solutions impose a comparable degree of regularization inferred to the MAP estimates. However, a practical consideration arises when selecting an appropriate value of β. As β becomes too large, the likelihood - and consequently the conditional posterior - becomes exceedingly sharp, leading to steep gradients around high-valued objective regions. In such cases, the MCMC chain is prone to becoming effectively "trapped" in these narrow peaks, as proposed moves away from the current mode receive vanishingly small acceptance probabilities α_θ. This occurs because the prior no longer exerts sufficient regularizing influence to counterbalance the likelihood's dominance, unlike in regimes where β is moderate and the posterior retains a broader structure. Hence, while increasing β can give rise to solutions with improved in-sample performance, it must be done judiciously to avoid compromising the chain's ability to explore alternative dominant modes of the conditional.

Additionally, we note from Table 3, that the maximum of the log of the conditional posterior, $\log\big(p(\boldsymbol{\theta} \mid \mathcal{D}, \sigma_\theta^2)\big)$, increases as β increases. This behavior is substantiated by Equation 3, where it follows directly that increasing β increases the contribution of the likelihood to the conditional posterior, thereby sharpening the overall posterior landscape.

	$\frac{100}{K}\sum_{k=1}^{K}\mathbb{I}\left(\rho_{T_k}(\boldsymbol{\theta})=+1\right)$			
Sharpness β	In-Sample	Out-of-Sample	max $\log\left(p\left(\boldsymbol{\theta} \mid \sigma_\theta^2, \mathcal{D}\right)\right)$	ESS
0.1	56	50.08	-254.1965	2332.9560
1	70	59.86	-237.8046	2340.9900
10	63	56.68	-243.4494	4222.4970
50	86	72.98	-200.5137	690.1510
100	98	72.62	-162.3267	927.7915
1000	96	68.58	733.5609	927.4778

Table 3: The normalized number of O wins, as a percentage $\left(\frac{100}{K}\sum_{k=1}^{K}\mathbb{I}\left(\rho_{T_k}(\boldsymbol{\theta})=+1\right)\right)$ for in-sample ($K=100$) and out-of-sample ($K=10,000$) sets across likelihood sharpness $\beta \in \mathbb{R}^+$ using $\overset{\text{(II)}}{\textbf{model}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\text{GA,(II)}}\right)$ and $\sigma_{\text{Init}}^2 = 10$ for $100,000$ MCMC iterations and burn-in of $20,000$ iterations.

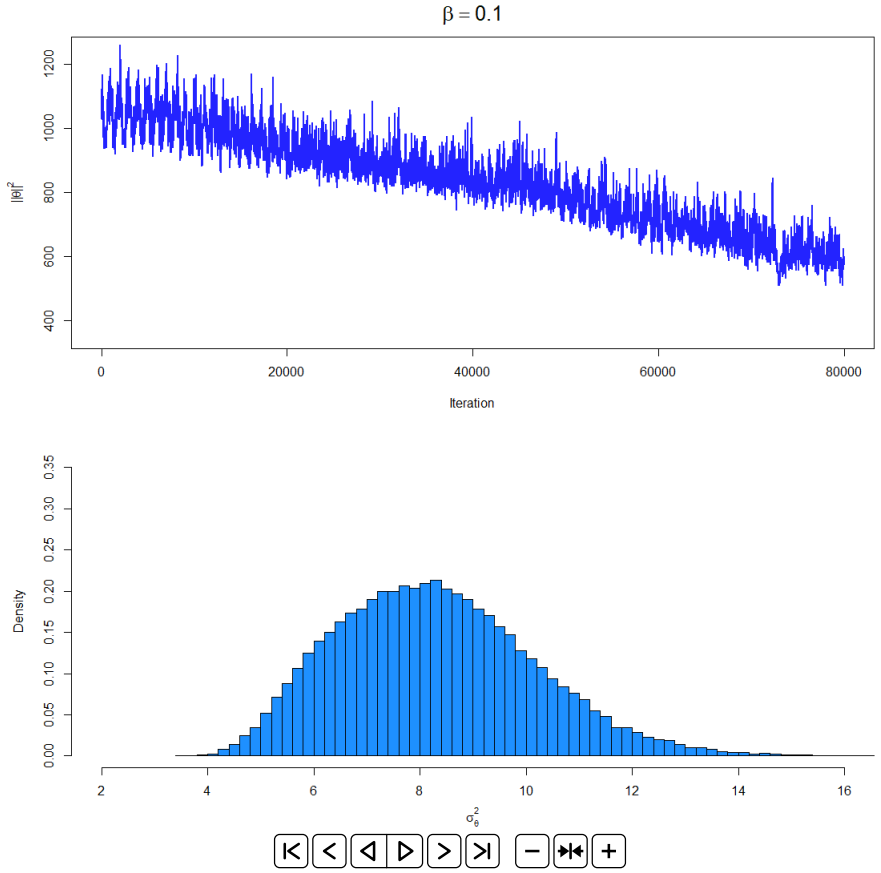

Figure 1: $\|\boldsymbol{\theta}^{(j)}\|^2$ for $j = 1, \dots 80,000$ (post burn-in) with distribution of marginal $\sigma_\theta^2 \mid \mathcal{D}$ for varying likelihood sharpness β using $\underset{\text{model}}{\overset{\text{(II)}}{}}\left(\mathbf{a}^0, \hat{\boldsymbol{\theta}}_\nu^{\text{GA,(II)}}\right)$ and $\sigma_{\text{Init}}^2 = 10$.

With regard to the amount of regularization inferred from the training set - where regularization is represented by the dispersion parameter $\sigma_\theta^2 \propto \frac{1}{\nu}$ - we may draw meaningful conjectures from the marginal distributions of $\sigma_\theta^2 \mid \mathcal{D}$ shown in Figure 1. Specifically, the variation in these distributions across different values of sharpness parameter β, suggests that different β values inherently induce different degrees of regularization, which are in turn reflected in the MAP estimates of the parameters (as discussed in Section **??**, MAP estimates naturally encode a level of regularization inferred by the training set, since the marginal posterior $p(\theta_i \mid \mathcal{D})$ essentially integrates over both the remaining parameters $\boldsymbol{\theta}_{-i}$ and the dispersion parameter σ_θ^2). That is, by examining the marginal distribution of $\sigma_\theta^2 \mid \mathcal{D}$, we gain insight into the implied concentration of the regularization strength ν. Interestingly, we observe that beyond a certain threshold - approximately at $\beta = 100$ for our tic-tac-toe problem - these inverse-gamma marginals appear to converge in shape and scale, as evidenced by the similarity between the distributions for $\beta = 100$ and $\beta = 1000$ in Figure 1. This suggests that the strength of regularization inferred by the training set saturates beyond a certain level of likelihood sharpness.

Even more revealing with respect to the amount of regularization inferred by the training set, is the influence of the initial variance σ_{Init}^2 used to initialize the Markov chain. Recall that the initial proposal $\boldsymbol{\theta}^{(1)}$ is sampled from a multivariate normal distribution, $\mathcal{N}\left(\mathbf{0}_{S\times 1}, \sigma_{\text{Init}}^2 \mathbf{I}_S\right)$. Notably, when fixing $\beta = 100$ to ensure stability of $\|\boldsymbol{\theta}^{(j)}\|^2$, the resulting marginal distributions of $\sigma_\theta^2 \mid \mathcal{D}$ remain inverse-gamma distributed with approximately constant shape parameters but exhibit increasing rate parameters as σ_{Init}^2 increases, as shown in Table 4. This trend is visually supported in Figure 2 (note that the axis scales vary across plots), which displays the inverse-gamma marginals $\sigma_\theta^2 \mid \mathcal{D}$ for increasing values of σ_{Init}^2. It is evident that both the mean and variance of the resulting distributions shift upward - the distributions move to the right and become more compressed. While this "squashing" effect may not be immediately noticeable without paying attention to the axis scales, it highlights an important insight: since $\sigma_\theta^2 \propto \frac{1}{\nu}$, increasing the initial variance σ_{Init}^2 results in the inference of a weaker regularization strength from the training set. In other words, the amount of regularization implicitly inferred by the training set is not only a function of the likelihood sharpness β, but is also influenced by the choice of initial dispersion σ_{Init}^2.

Now as previously argued in Section **??**, the motivation for adopting a hierarchical Bayesian framework - wherein a dispersion parameter σ_θ^2 is introduced via a prior such that $\sigma_\theta^2 \propto \frac{1}{\nu}$ - was to allow the training set to inform the degree of regularization. However, the preceding analysis including Section **??** reveals a tension in this reasoning. Specifically, the psuedo-likelihood form, the likelihood sharpness β and the initial dispersion σ_{Init}^2 are user-specified hyperparameters that exert a substantial influence on the marginal posterior distribution of $\sigma_\theta^2 \mid \mathcal{D}$. This dependence implies that the extent of regularization is not fully inferred by the training set, but is instead strongly shaped by likelihood and prior design choices - particularly the settings of β and σ_{Init}^2. As such, we may question whether the hierarchical structure genuinely facilitates data-driven regularization or whether it merely reintroduces user-defined regularization through a more complex inferential route. From this perspective, the use of a Bayesian hierarchical model for the sole purpose of inferring σ_θ^2 from the training set may appear unnecessary, especially when the same effect could be achieved by explicitly fixing σ_θ^2 (and hence regularization strength ν) to a chosen value. In this light, one might argue that the two-sample MCMC procedure employed here - through which the Bayesian hierarchical model is implemented - is actually just the *user* inferring a specific regularization strength but with "extra steps".

	$\frac{100}{K}\sum_{k=1}^{K}\mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right)=+1\right)$				
Initial Variance σ^2_{Init}	In-Sample	Out-of-Sample	Shape	Rate	ESS
0.1	92	71.18	41.7518	3.1350	414.6554
1	88	65.40	46.2323	41.3117	884.6408
10	98	72.62	44.9264	413.2980	927.7915
100	92	68.92	47.28048	5704.8909	488.9163

Table 4: The Normalized number of O wins, as a percentage $\left(\frac{100}{K}\sum_{k=1}^{K}\mathbb{I}\left(\rho_{T_k}\left(\boldsymbol{\theta}\right)=+1\right)\right)$ for in-sample ($K=100$) and out-of-sample ($K=10,000$) sets across various initial variances σ^2_{Init} accompanied by shape and rate parameters of marginal $\sigma^2_\theta \mid \mathcal{D} \sim$ Inv-Gamma for likelihood sharpness $\beta = 100$.

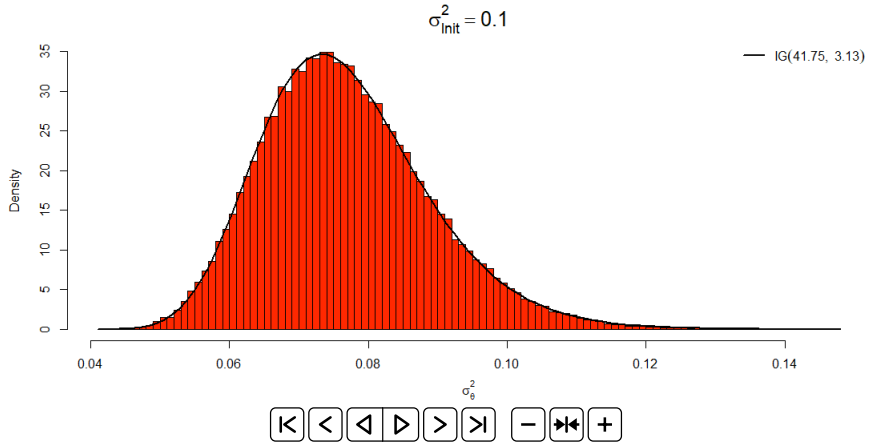

Figure 2: Distribution of marginal $\sigma^2_\theta \mid \mathcal{D} \sim$ Inv-Gamma(Shape, Rate) for varying initial variance σ^2_{Init} (plots are on different x and y scales).

References

Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., and Bürkner, P.-C. (2021). Rank-normalization, folding, and localization: An improved $\hat{R}$ for assessing convergence of mcmc. *Bayesian Analysis*, 16(2):667–718.