FURTHER VARIANCE REDUCTION VIA UNCERTAINTY PROPAGATION AND HEURISTIC PATHOLOGIES IN THE AIVAT FAMILY OF TECHNIQUES

Anonymous authors
Paper under double-blind review

ABSTRACT

How should an agent's performance in a multiagent environment be evaluated when there is a limited sample size or a high cost of running a trial? The AIVAT family of variance reduction techniques was proposed to address this challenge by introducing unbiased low-variance estimators of agents' expected payoffs. An important component of AIVAT is a heuristic value function that discriminates between low- and high-value counterfactual histories. A notable gap in the literature is that there is little to no constraint or guideline on how the heuristic value function should be chosen or how uncertainty in its output should be handled.

In our main contribution, we show how the heuristic uncertainty can be propagated to quantify the uncertainty of AIVAT estimates. It is then possible to further reduce the variance using inverse-variance weighted averaging, but AIVAT's unbiasedness guarantee may have to be sacrificed. In our secondary contribution, we parameterize the heuristic value function to highlight AIVAT's potential vulnerabilities: by directly applying gradient descent/ascent on the sample variance or the test statistic, a) the sample variance can be set pathologically low, and b) one can p-hack to draw a desired statistical conclusion. The key takeaway is that the heuristic value function should be fixed prior to observing the evaluation data! In our experiments, we use a dataset of poker and Goofspiel games to demonstrate our heuristic uncertainty and pathology results, with the former yielding a 43.0% reduction in the number of samples needed to draw statistical conclusions.

1 INTRODUCTION

Evaluating an agent's performance in a multiagent environment can be challenging when running a trial is costly or time-consuming. This is especially true when demonstrating the superhuman capability of an AI agent, requiring human experts to compete against the agent for a long time. For example, the evaluation of the superhuman heads-up poker AI agent Libratus (Brown & Sandholm, 2018) lasted 20 days, morning to evening, and involved four human professionals competing in parallel for a $200,000 prize pool. The resource-intensive process of generating sufficient data to draw statistically significant conclusions cannot be avoided without a low-variance outcome estimator.

The AIVAT (Burch et al., 2018) family of variance reduction techniques was proposed to handle the often high-variance nature of extensive-form games. AIVAT reduces the variance introduced by both nature and player actions in primarily two ways. First, using a heuristic value function, AIVAT evaluates the potential values of counterfactual histories when counterfactual actions are applied to observed histories with known probabilities. Second, AIVAT uses the fact that, regardless of a particular player's hidden information, others would have acted identically as they did in the original observations. Burch et al. (2018) showed that the AIVAT estimator is unbiased, and experimentally demonstrated a reduction in the required number of trials by "more than a factor of 10" to make the same statistical claims as when it is not used. Furthermore, the power of AIVAT increases as the number of players whose strategies are known increases. However, it is also true that its power decreases as fewer player strategies are taken into account, with it being reduced to MIVAT (White & Bowling, 2009) when only chance probabilities are known.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Evaluations of the poker AI agents Pluribus (Brown & Sandholm, 2019) and DeepStack (Moravčík et al., 2017), as well as submissions to the Annual Computer Poker Competitions (ACPC) (Bard et al., 2013) represent the landmark applications of the AIVAT family of techniques. The first application was particularly striking in that, although Pluribus finished the experiment with a negative payoff overall, AIVAT was able to show that Pluribus was, in fact, superhuman.

1.1 Our contributions

We provide two contributions with regard to the AIVAT family of variance reduction techniques. In our *main* contribution, we note that while AIVAT takes account of the uncertainty associated with player actions, in its usage of the heuristic value function, it introduces another source of uncertainty, namely, how certain the heuristic value function is in its outputs. One may be more certain of some outputs of the heuristic value function while being less so about others. This is certainly the case when the value function outputs are nondeterministically approximated game-theoretic values (*e.g.*, Monte Carlo rollouts and/or randomized clustering during abstraction) or are learned from existing data (White & Bowling, 2009) and are predicting from inputs in the low- versus high-density region of the training distribution. We quantify the uncertainty in terms of the variance and demonstrate how the heuristic value function uncertainty can be propagated to the estimate level, obtaining a measure of how uncertain a particular value estimate is. We also show that inverse-variance weighting can be applied, where less weight is given to estimates with more uncertainty and vice versa, to achieve **further variance reduction**, albeit at the risk of incorporating some bias into the estimate. Nonetheless, we a) show the necessary condition for the bias to be zero, b) demonstrate how the bias can be estimated, and c) contend that it is unlikely for game-playing agents to manipulate the bias to appear to play better. Using the gameplay data of Pluribus and Goofspiel, we report a reduction of up to 43.0% in the number of required trials (*i.e.*, poker hands) to reach statistical conclusions. Our findings make multiagent evaluation more scalable.

Our *secondary* contribution is **cautionary**. We expand on the fact that surprisingly little has been said about the constraints on how the heuristic value function can be developed. When introducing the advantage sum, which forms the basis of DIVAT (Billings & Kan, 2006), MIVAT (White & Bowling, 2009), and AIVAT (Burch et al., 2018), Zinkevich et al. (2006) state that "any" and "all" heuristic value functions yield an unbiased estimator of the true value. White & Bowling (2009) suggest learning a linear value function from the sample data, where they optimize for the sample variance as a proxy of the true variance, but offer no additional guidelines. Burch et al. (2018) use self-play values as the arbitrary fixed value function, which was also involved during the data-generating process of the same data on which the technique is applied. In this paper, we highlight AIVAT's potential vulnerabilities by demonstrating its analogue of the well-known phenomenon of overfitting (including adversarial optimization) in supervised learning: it is possible to learn a heuristic value function that a) obtains pathologically low variance or b) p-hacks to falsely draw a desired statistical conclusion about an agent's performance. Using the gameplay of Pluribus and Goofspiel players, we train such a function by parameterizing the heuristic outputs and applying gradient descent/ascent on the desired objective. The key takeaway is that the heuristic value function *should be fixed prior to observing the evaluation data*! (The use of Pluribus data is for demonstration and should not be taken as a criticism of Brown & Sandholm (2019); they were correct.)

2 Notation and background

In this section, we define the notation used throughout the paper and provide the background on extensive-form games and the AIVAT family of variance reduction techniques.

2.1 Extensive-form games

In this paper, we focus our analysis on extensive-form games, but ideas from AIVAT can also be applied to other game representations. An extensive-form game has a finite set of players P (including chance p_c) and histories H. Every history is a sequence of actions played by each player $i \in P$, and is associated with a player $p(h)$ and a set of available actions $A(h)$. $h \cdot a = h'$ denotes that applying $a \in A(h)$ at h leads to h'. If $p(h) = p_c$, then $f_c(h, a)$ gives a fixed probability distribution over each available action $a \in A(h)$. Each terminal history $z \in Z \subseteq H$ has a utility $u_i(z)$ for every player i.

The imperfect information setting is represented by information sets $\mathcal{I}_i$: a partition of histories belonging to a non-chance player $i \in P \setminus \{p_c\}$. Player i cannot distinguish between $h, h' \in I \in \mathcal{I}_i$. Therefore, $A(h) = A(h')$, and we denote the set of available actions at an information set by $A(I)$. Each player i plays with a strategy $\sigma_i(I)$ which assigns a probability distribution over $A(I)$. Then a strategy profile σ is defined as a tuple of all player strategies. We use $\pi(h)$ to represent the probability of reaching h given players follow σ. The contribution of player i to $\pi(h)$ is $\pi_i(h)$.

By definition, extensive-form games assume discrete action spaces. In continuous settings, the action space can be discretized, or AIVAT can be trivially extended to perform Monte Carlo sampling on actions.

2.1.1 Agent evaluation

Agent evaluation in extensive-form games typically requires obtaining an estimate of the expected utility of a particular player i on a given strategy profile σ:

$$\mathbb{E}_{z \in Z}[u_i(z)|\sigma] = \sum_{z \in Z} \pi(z) u_i(z).$$

Before the advent of variance reduction techniques for extensive-form games, Monte Carlo samples $z_1, \dots z_T$ were drawn independently to calculate the mean player utility,

$$\bar{u}_i = \frac{1}{T} \sum_{t=1}^{T} u_i(z_t),$$

which is an unbiased estimator, *i.e.*,

$$\mathbb{E}[\bar{u}_i|\sigma] = \mathbb{E}_{z \in Z}[u_i(z)|\sigma],$$

and has the following variance:

$$\mathrm{Var}[\bar{u}_i|\sigma] = \frac{1}{T}\mathrm{Var}[u_i(z)|\sigma].$$

The multiagent environment's stochasticity and desired statistical significance level influence the choice of T. One is therefore limited by the cost of running each trial, and it may be far too expensive to draw a statistically significant conclusion using the Monte Carlo method.

2.2 Variance reduction techniques

The AIVAT family of variance reduction techniques specializes the control variates method for agent evaluation in extensive-form games to give a low-variance estimate of any function $v(z)$, an example of which is player utility. The current state-of-the-art variance reduction technique is AIVAT (Burch et al., 2018), which can be thought of as a combination of its two predecessors: imaginary observations (Bowling et al., 2008) and the advantage sum (Zinkevich et al., 2006). The AIVAT family of techniques yields unbiased estimators of the value function.

2.2.1 Control variates

The control variates method (Glasserman, 2004, Ch. 4) is a standard way to reduce variance in Monte Carlo methods. We begin by giving a brief description in the language of agent evaluation. In this setting, we seek to estimate $\mathbb{E}_{z \in Z}[v(z)|\sigma]$. Suppose the existence of another value function $w(\cdot)$ where $\omega = \mathbb{E}_{z \in Z}[w(z)|\sigma]$ is known. Then, the following is an unbiased estimator of $\mathbb{E}_{z \in Z}[v(z)|\sigma]$:

$$\hat{v}(z) = v(z) - c(w(z) - \omega)$$

for any choice of constant c. Its variance is as follows:

$$\mathrm{Var}(\hat{v}(z)) = Var(v(z)) + c^2 Var(w(z)) - 2c\mathrm{Cov}(v(z), w(z)). \tag{1}$$

When $c^2 Var(w(z)) - 2c\mathrm{Cov}(v(z), w(z)) \leq 0$, variance reduction is achieved. The optimal choice of c can be derived by differentiating (1) with respect to c:

$$c^* = \frac{\mathrm{Cov}(v(z), w(z))}{\mathrm{Var}(w(z))},$$

which results in

$$\mathrm{Var}(\hat{v}(z)) = (1 - \mathrm{Corr}(v(z), w(z))^2)\mathrm{Var}(v(z)).$$

2.2.2 ADVANTAGE SUM

The advantage sum technique uses control variates with a heuristic value function to reduce the variance of the estimate and is of the following form:

$$\hat{v}(z) = v(z) - \hat{v}_c(z),$$

where $\hat{v}(\cdot)$, $v(\cdot)$, and $\hat{v}_c(\cdot)$ denote the estimate, value function, and correction term, respectively. The correction term, which can be thought of as control variates, is defined as follows:

$$\hat{v}_c(z) = \sum_{h \cdot a \in K(z)} \left(v'(h \cdot a) - \sum_{a' \in A(h)} f_c(h, a') v'(h \cdot a') \right),$$

with $v'(\cdot)$ the heuristic value function and $K(z)$ the set of histories preceding z where the probability distribution of available actions at the immediate parent history is known. MIVAT (White & Bowling, 2009) represents a special case of the advantage sum where no player's action probabilities, except those of nature, are known. In this particular context, the subtraction of the correction term can be thought of as canceling the effect of luck (or lack thereof). No matter the choice of the heuristic value function $v'(\cdot)$, Zinkevich et al. (2006) showed that the advantage sum (and hence MIVAT) is an unbiased estimator of the expected value, *i.e.*, $\mathbb{E}_{z \in Z}[\hat{v}(z)|\sigma] = \mathbb{E}_{z \in Z}[v(z)|\sigma]$.

White & Bowling (2009) proposed using a linear function as the heuristic value function, which is to be trained on existing sample data. In doing so, they proposed to minimize the sample variance as a proxy for the true variance, hence the following optimization problem for the mean-squared error:

$$\textbf{Minimize:}_{\hat{v}: Z \mapsto \mathbb{R}} \sum_{t=1}^{T} \left(\hat{v}(z_t) - \frac{1}{T} \sum_{t'=1}^{T} \hat{v}(z_{t'}) \right)^2. \tag{2}$$

They derived a closed-form formula for an optimal linear value function given a feature engineering function and applied it to heads-up fixed-limit, heads-up no-limit, and 6-max fixed-limit poker, observing reductions of up to about 62%, 23%, and 18%, respectively, in the standard deviation.

2.2.3 IMAGINARY OBSERVATIONS

Imaginary observations do not use a heuristic value function or control variate terms. Instead, given a trial outcome z, it directly applies the value function to a subset of terminal nodes and obtains an estimate of the expected payoff using importance sampling. It is suited for estimating the expected payoff of alternative strategies (which they dubbed the off-policy case). The imaginary observation estimator is unbiased in both the on-policy and off-policy cases with full information; however, this guarantee is lost with partial information (Bowling et al., 2008). Nevertheless, imaginary observations in the partial information case remain a useful tool as they can yield a low-variance estimate.

2.2.4 AIVAT

AIVAT, visualized in Appendix A, is a generalization of the advantage sum where imaginary observations are applied to each considered history. It is of the following form:

$$\hat{v}(z) = \hat{v}_b(z) + \hat{v}_c(z), \qquad \hat{v}_b(z) = \frac{\sum_{z' \in U(z)} \pi(z') v(z')}{\sum_{z' \in U(z)} \pi(z')},$$

$$\hat{v}_c(z) = \sum_{h \cdot a \in K(z)} \left(\frac{\sum_{a' \in A(U(h))} \sum_{h' \in U(h)} \pi(h' \cdot a') v'(h' \cdot a')}{\sum_{h' \in U(h)} \pi(h')} - \frac{\sum_{h' \in U(h)} \pi(h' \cdot a) v'(h' \cdot a)}{\sum_{h' \in U(h)} \pi(h' \cdot a)} \right),$$

where $\hat{v}_b(z)$ and $\hat{v}_c(z)$ are the base and correction terms, respectively. $U(h)$ is defined as a set of histories differing from h only by the private information belonging to $p(h)$. $A(U(h))$ is the common set of actions available to that player. Burch et al. (2018) showed that AIVAT is an unbiased estimator regardless of $v'(h)$ and proposed using an AI agent's self-play values as the outputs of the heuristic value function, which can also be used to generate the very data being evaluated. More recently, Li et al. (2026) proposed the use of confidence sequences with AIVAT in streaming settings.

While AIVAT is a useful tool in an agent evaluator's toolkit, the reduction in variance depends on how good the chosen heuristic value function is. In general, one may only have access to poor heuristic value functions, in which case applying AIVAT can be **counterproductive**, *i.e.*, the estimates might be of even higher variance than before. This presents the need to improve upon AIVAT.

3 FURTHER VARIANCE REDUCTION USING HEURISTIC UNCERTAINTY

We are now ready to present our *main* contribution, which is a new way of obtaining **additional** variance reduction on top of AIVAT. We approach the AIVAT family of variance reduction techniques from a previously unexplored angle of utilizing the uncertainty of heuristic outputs. For convenience, we simplify the expression for the AIVAT estimate $\hat{v}(\cdot)$ by rewriting it as an affine function of the outputs of the value function $v'(\cdot)$, as follows:

$$\hat{v}(z) = b(z) + \sum_{h \in H} \mathbf{c}(z)_h v'(h), \tag{3}$$

where $b(z)$ is the affine shift and $\mathbf{c}(z) \in \mathbb{R}^H$ is the vector of coefficients of each heuristic output $v'(\cdot)$. Depending on the game, the size of H may be huge; however, for our purpose, we only need to consider the histories and counterfactual histories h encountered during agent evaluation (*i.e.*, where $\mathbf{c}(z)_h \neq 0$), which is usually a small fraction of the original game tree size if the game is large. The detailed derivation process, as well as how $b(z)$ and $\mathbf{c}(z)$ are defined, is relegated to Appendix B.

The affine structure of AIVAT allows the uncertainty of the estimates to be cleanly derived. In this paper, we use variance to quantify uncertainty. Defining the covariance matrix $\mathbf{\Sigma}(z)$ where, for each $(h_1, h_2) \in H \times H$, $\mathbf{\Sigma}(z)_{h_1,h_2} = \mathrm{Cov}(v'(h_1), v'(h_2))$,

$$\mathrm{Var}(\hat{v}(z)) = \mathrm{Var}\left(b(z) + \sum_{h \in H} \mathbf{c}(z)_h v'(h)\right) = \mathrm{Var}\left(\sum_{h \in H} \mathbf{c}(z)_h v'(h)\right) = \mathbf{c}(z)^\top \mathbf{\Sigma}(z) \mathbf{c}(z). \tag{4}$$

Note that, in practice, $\mathbf{c}(z)$ is sparse (see Appendix B), and hence $\mathbf{\Sigma}(z)$ can be implemented as sparse, setting irrelevant rows and columns to zeros. One can simplify the expression further, as in the following, by assuming that the heuristic value outputs are uncorrelated:

$$\mathrm{Var}(\hat{v}(z)) = \sum_{h \in H} \mathbf{c}(z)_h^2 \mathrm{Var}(v'(h)). \tag{5}$$

That said, the above assumption was **not** made in our main experimental results, presented later.

Just as we are given T independent trials $z_1, \ldots, z_T$ and assume the values $v(z_1), \ldots, v(z_T)$ are independent and identically distributed (i.i.d.), we make the standard (Brown & Sandholm, 2019) assumption that the AIVAT estimates are also i.i.d. Note that in typical settings where our technique could be used, *e.g.*, hypothesis testing during agent evaluation, making such an assumption is a common practice. Thus, in many practical settings where our technique can be used, invoking its theoretical guarantees would not require introducing any more assumptions. Then, the variance of the arithmetic mean of AIVAT estimates $\bar{v} = \frac{1}{T} \sum_{t=1}^{T} \hat{v}(z_t)$ is as follows:

$$\mathrm{Var}(\bar{v}) = \frac{\sum_{t=1}^{T} \mathrm{Var}(\hat{v}(z_t))}{T^2}. \tag{6}$$

We can treat the variance of the mean as a proxy objective to be minimized. It is possible to improve from (6) by assigning a weight w_t to each estimate to obtain the weighted average, *i.e.*,

$$\bar{v}^* = \frac{\sum_{t=1}^{T} w_t \hat{v}(z_t)}{\sum_{t=1}^{T} w_t}.$$

Consider inverse-variance weighting (IVW) (Hartung et al., 2011, Ch. 4), which puts less weight on those with higher uncertainty and vice versa: $w_t = \frac{1}{\mathrm{Var}(\hat{v}(z_t))}$. Then, the variance is

$$\mathrm{Var}(\bar{v}^*) = \frac{1}{\sum_{t=1}^{T} \frac{1}{\mathrm{Var}(\hat{v}(z_t))}}. \tag{7}$$

Proposition 1. *Assuming that the AIVAT estimates are independent, their IVW average yields the minimum variance amongst all weighted averages.*

The proof is in Appendix D.2. It follows from a well-known result in statistics; cf. Hartung et al. (2011, Ch. 4). Thus, IVW improves upon uniform weighting. When the estimates being averaged over are all equally uncertain, (6) and (7) are equivalent. Ideally, we would like the IVW average to also be an unbiased estimator. This is true when the weights are independent of the data being averaged over, but not in general, as shown in the following proposition:

Proposition 2. *When the weights are independent of the data being averaged over, the weighted average of AIVAT estimates is unbiased, i.e.,* $\mathbb{E}[\bar{v}^*|\sigma] = \mathbb{E}_{z\in Z}[v(z)|\sigma]$. *But when the weights are correlated with the data being averaged over, the weighted average of AIVAT estimates is, in general, biased, i.e.,* $\mathbb{E}[\bar{v}^*|\sigma] \neq \mathbb{E}_{z\in Z}[v(z)|\sigma]$. *Assuming that the data points are i.i.d., the asymptotic bias is* $\mathrm{Cov}(w, \hat{v}(z)|\sigma)/\mathbb{E}[w]$, *where* w *and* $\hat{v}(z)$ *are random variables representing the unnormalized weights and estimated expected utilities.*

The proof is in Appendix D.3. To address this, we can **estimate** the bias, and our estimator can remain useful if the estimated bias is much smaller than the reduction in the standard error. While it can also be used to offset the IVW mean, this would add a highly non-trivial term to the variance.

It is important to note that our estimator does **not** inherently introduce bias. Indeed, whether or not our estimator is biased depends on the specific learning algorithm used to train the heuristic value function, since this determines the AIVAT estimates we obtain and their corresponding weights. Additionally, fixing the specific learning algorithms can help us derive even stronger theoretical bounds. In our experiments, the model we utilized as the heuristic value function is *Gaussian process regressor (GPR)* (Williams & Rasmussen, 2006), which relies on Gaussian assumptions, of which one of the fundamental statistical properties is that the mean and the variance are structurally independent. The estimated uncertainty depends only on the input space, so the output and estimated uncertainty have zero covariance under Gaussian priors, so our estimator is unbiased. The same holds for other models in Appendix H, which operate under the same foundational statistical assumptions.

Besides, it is unclear how a player could even play to manipulate such a weighting scheme, since the bias depends on the correlation between the IVW weights and the expected utility estimates. It would require them to play differently while predicting the confidence of the heuristic value function chosen by the evaluator in its outputs for different histories. A similar bias-variance tradeoff was previously explored in the context of imaginary observations (a predecessor of AIVAT) in the case of partial information. We also demonstrate this tradeoff in our experiments below.

3.1 Experiments on heuristic uncertainty propagation

We conducted experiments on the publicly available poker hand history data from the Pluribus experiment (Brown & Sandholm, 2019) (poker rules are in Appendix E). Since Pluribus's action distribution is not known to us, our implementation of AIVAT reduces to MIVAT. In learning the heuristic value function for our MIVAT estimator, we chose the *Gaussian Process Regressor (GPR)* (Williams & Rasmussen, 2006) with Dot Product and White kernels. We refer to this estimator as MIVAT-GPR. For feature engineering, we followed White & Bowling (2009) (see Appendix F). The Pluribus data consists of 10,000 poker hands. We applied 10-fold cross-validation to calculate the estimates for the entirety of the data. In each fold, a heuristic value function was trained using all history-payoff pairs in the training set, and then the trained estimator was evaluated on the test set. While training MIVAT-GPR, 1,000 data points were subsampled in each fold due to computational constraints.

Our objective is not to draw statistical conclusions about Pluribus's superhuman performance, which, realistically, would require access to Pluribus's strategies. Instead, we seek to demonstrate that IVW gives rise to an estimate that has a lower standard error of the (weighted) mean than uniform weighting (more details are in Appendix G). Here, we also include the results for AIVAT reported by Brown & Sandholm (2019). It would be unfair to compare our performance with theirs, as they had access to Pluribus's action probabilities, which vastly increase AIVAT's variance-reduction power compared to when only the chance probabilities are known (which is what we know). Additionally, the results using MIVAT with the linear heuristic value function obtained through the steps introduced by White & Bowling (2009) were included, which we denote as MIVAT-WB. In Appendix I, we derive it for the more general case of AIVAT, but we also critique its design.

Table 1 shows the performance of our MIVAT-GPR estimator, along with that of the baselines. When taking the simple mean, MIVAT-GPR does worse than MIVAT-WB, but this is understandable, as we used far less data to train GPR due to computational constraints. Also, under uniform averaging, MIVAT-GPR performs worse than even when no estimator is applied (see 'Raw'). It is clear that, due to the much smaller training data used to train its heuristic value function, GPR is quite inadequate in calculating the heuristic value of some counterfactual situations. However, under IVW averaging, our MIVAT-GPR estimator is also observed to far outperform itself under uniform averaging and MIVAT-WB. Indeed, MIVAT-GPR notably achieved approximately 24.5% reduction in the standard

Table 1: Results on variance reduction via heuristic uncertainty propagation in milli-big blinds per hand (mbb/h). SE stands for the standard error of the (weighted) mean. Our results should not be compared with results (starred) of Brown & Sandholm (2019), as we lacked access to Pluribus.

Estimator	Weighting	Win rate	SE	Est. bias
MIVAT-GPR	Uniform	-25	99	–
	IVW (ours)	-22	**75**	3
Raw	Uniform	-70	88	–
MIVAT-WB	Uniform	-98	85	–
AIVAT (Brown & Sandholm, 2019)	Uniform	48	*25**	–

error of the (weighted) mean compared to when uniform averaging was used, corresponding to roughly 43.0% reduction in the number of hands required to reach the same statistical significance. Additionally, the estimated bias of the IVW estimate is about an order of magnitude smaller than the reduction in the standard error of the weighted mean. This is evidence that IVW, which puts lower weights on outputs with higher uncertainty, can help achieve a low-variance estimate of expected player utilities in games even when heuristics are sometimes poor.

Also, it is notable that MIVAT-WB (White & Bowling, 2009) performs worse than MIVAT-GPR under IVW and is not much better than when no estimator is used. At first glance, this is surprising, as their machine learning procedure is supposed to produce a linear heuristic value function that is optimal in the sense that it minimizes the sample variance of the training set. In Appendix I, we note the estimator's lack of regularization and suspect that it can be prone to overfitting, which hurts its performance. To our knowledge, we are the first to note that much simpler procedures of learning the heuristic value function can outperform the value function of White & Bowling (2009).

The results also clearly show that our technique can allow poor heuristic functions (that are counterproductive in AIVAT) to be fruitfully used in AIVAT. Indeed, MIVAT-GPR yields estimates of even higher variance when vanilla MIVAT is used compared to the raw estimates, but applying IVW noticeably reduces the variance. In practice, this is important because reliable expected utility estimators are not always easy to implement. While feature engineering techniques for certain domains like poker are well-studied, this is not the case for many imperfect-information games encountered in the real world. Thus, one may only have access to poor value functions, whose variance reduction power can be **boosted** by our technique. Additional experiments, including those for Goofspiel, for which our implementation of AIVAT does **not** reduce to MIVAT, are detailed in Appendix H.

4 Heuristic Pathologies

We continue by presenting our *secondary* result, which is a **cautionary tale**, highlighting AIVAT's potential vulnerabilities. We do so by demonstrating its analogue of the well-known phenomenon of overfitting (including adversarial optimization) in supervised learning: one can learn a heuristic value function to a) obtain pathologically low variance or b) p-hack to falsely draw a desired statistical conclusion about an agent's performance. Our results underscore the need to fix the heuristic value function before evaluation. We continue from the simplified expression shown in (3). We define a vector $\boldsymbol{\theta} \in \mathbb{R}^H$ to parameterize each heuristic output as follows:

$$v'_{\boldsymbol{\theta}}(h) = \theta_h. \tag{8}$$

By parameterizing as such, we give the heuristic value function the maximum expressive power. Again, the size of H can be huge. However, we only need to parameterize the (counterfactual) histories encountered during agent evaluation. We can further simplify the AIVAT estimate expression:

$$\hat{v}_{\boldsymbol{\theta}}(z) = b(z) + \sum_{h \in H} \mathbf{c}(z)_h v'_{\boldsymbol{\theta}}(h) = b(z) + \langle \mathbf{c}(z), \boldsymbol{\theta} \rangle. \tag{9}$$

Optimizing for the sample variance. Using (9), the proxy objective of optimization problem in (2) can be refined as follows:

$$\textbf{Minimize:}\ C(\boldsymbol{\theta}) = \sum_{t=1}^{T} \left(\hat{v}_{\boldsymbol{\theta}}(z_t) - \frac{1}{T} \sum_{t'=1}^{T} \hat{v}_{\boldsymbol{\theta}}(z_{t'}) \right)^2 = \sum_{t=1}^{T} \left(\left(b(z_t) - \bar{b} \right) + \langle \mathbf{c}(z_t) - \bar{\mathbf{c}}, \boldsymbol{\theta} \rangle \right)^2,$$

where $\bar{b} = \frac{1}{T}\sum_{t=1}^{T} b(z_t)$ and $\bar{\mathbf{c}} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{c}(z_t)$. (More details are given in Appendix C.)

Proposition 3. *On a given set of trials, there exists an optimal parameter vector $\boldsymbol{\theta}^*$ that achieves the lowest possible sample variance of the estimates.*

The proof is in Appendix D.1. It follows from the fact that finding the pathological heuristic outputs that minimize the sample variance of a given dataset is a least-squares problem.

Optimizing for the *t*-statistic. Optimizing for the sample variance may not be very interesting, as we do not explicitly control what can be said about the given data. Indeed, one of the main purposes of using variance reduction techniques is to draw statistical conclusions about an agent's performance. This is usually done via a one-sided *t*-test from which a p-value is computed and compared against a desired statistical confidence level. We show that the p-value can be hacked by either minimizing or maximizing the *t*-statistic, as shown in the optimization problem below.

$$\textbf{Optimize:} \quad \underset{\boldsymbol{\theta}\in\mathbb{R}^H}{} \frac{\bar{v_{\boldsymbol{\theta}}} - \mu_0}{s_{\boldsymbol{\theta}}/\sqrt{T}},$$

where $\bar{v_{\boldsymbol{\theta}}} = \sum_{t=1}^{T} \hat{v}_{\boldsymbol{\theta}}(z_t)$ and $s^2_{\boldsymbol{\theta}} = \frac{\sum_{t=1}^{T}(\hat{v}_{\boldsymbol{\theta}}(z_t) - \bar{v_{\boldsymbol{\theta}}})^2}{T-1}$. We apply gradient descent/ascent on the parameters to minimize or maximize our objective.

4.1 Experiments on heuristic pathologies

We conducted our pathology experiments on the Pluribus data (Brown & Sandholm, 2019).

4.1.1 Optimizing for the sample variance

First, we aimed to produce a result of extremely low variance that does not pertain to reality. To do so, we optimized on the sample variance as the proxy objective (2). As Pluribus's action distribution is not known, our AIVAT implementation reduces to MIVAT.[1] We trained a heuristic value function, outputting an estimate of the expected utilities at every player position. Although the least-squares solution could theoretically have been solved, we could not fit the transformed data matrix $[(\mathbf{c}(z_t) - \bar{\mathbf{c}})^\top]_{t\in\{1,\dots,T\}}$ in memory, so we applied gradient descent for 250 iterations with the Adam optimizer ($\eta = 100, \beta_1 = 0.9, \beta_2 = 0.999, \lambda = 0$). The unusually high η was chosen, as conventional values (*e.g.*, 0.001) converged too slowly.

Using the learned parameters, we were able to achieve **pathologically** low variance estimates of all players' expected utilities, as shown in Table 2. Here, the results show that all players won by extremely high margins. Considering that win rates of 100 mbb/h are typically characterized as immense, these values are unrealistic. They also violate the zero-sum constraint; this is because we did not enforce such a constraint. This experiment shows that, when the data is fixed and the heuristic value function is directly optimized using the proxy objective (2), given by White & Bowling (2009), one can produce a nonsensical result that does not pertain to reality, especially with enough degrees of freedom to be exploited.

Table 2: Results on heuristic pathologies minimizing the sample variance (in mbb/h). The full table is shown in Appendix K. SE stands for the standard error of the mean.

Player	Win rate	SE
Pluribus	2062	25
Budd	2059	50
MrWhite	2043	37
MrOrange	2053	29
Hattori	2045	68
MrBlue	2062	26
⋮	⋮	⋮

4.1.2 Optimizing for the *t*-statistic

Continuing, we explored how to make false claims about the given data by p-hacking via gradient descent/ascent on the *t*-statistics. We again trained a parameterized heuristic value function except, this time, players' *t*-statistics were being optimized. When a player's *t*-statistic is being minimized, the null hypothesis is that the player did not lose. Conversely, when a player's *t*-statistic is being maximized, the null hypothesis is that the player did not win. The goal was to obtain a small enough p-value for each hypothesis test to support the claim that every player both won and lost. The parameters were optimized for 10 iterations with the Adam optimizer using the same hyperparameters.

[1] The reduction decreases the degrees of freedom that we can exploit, so it actually strengthens our results.

Table 3: Results on heuristic pathologies optimizing the t-statistic. Full results are in Appendix K.

Player	Losing		Winning	
	t-statistic	p-value	t-statistic	p-value
Pluribus	-95.050	$<10^{-1399}$	$+92.900$	$<10^{-1353}$
Budd	-49.538	$<10^{-373}$	$+55.012$	$<10^{-433}$
MrWhite	-76.612	$<10^{-816}$	$+75.638$	$<10^{-802}$
MrOrange	-95.291	$<10^{-1294}$	$+89.609$	$<10^{-1188}$
Hattori	-37.584	$<10^{-212}$	$+39.327$	$<10^{-226}$
MrBlue	-87.433	$<10^{-1207}$	$+93.654$	$<10^{-1336}$
⋮	⋮	⋮	⋮	⋮

The results are shown in Table 3. For every player, it was possible to train separate heuristic value functions, one for winning and another for losing, using which one could show that they both won and lost on the same data with overwhelmingly low p-values. The implication of this is that although the AIVAT family of variance reduction techniques is guaranteed to be unbiased, one can still craft a heuristic value function that supports whatever conclusion one wants to draw when the data being evaluated is known a priori. One interesting interpretation of this result is that, when an **adversary** controls the heuristic value function and is aware of the data being evaluated, the adversary can make the evaluator draw conclusions that are nonsensical, contradictory, or incorrect. Additional experiments on the synthetic data of 10,000 near-optimal gameplays in variants of Goofspiel (for which our implementation does **not** reduce to MIVAT) is shown in Appendix K.

Here, we have shown how severely the analogue of the overfitting behavior (to the game data) manifests in AIVAT's affine structure, as we could draw false or arbitrary conclusions with extremely high confidence. One interpretation of our results is that, if an adversary controls the heuristic value function, the adversary can manipulate the agent evaluation process to appear to be stronger than it really is, which is highly undesirable, as AIVAT's goal is to provide an accurate low-variance estimator. These pathologies shed light on the need to **fix** the heuristic value function **prior** to evaluation. There are many ways to do this correctly. For example, one can use self-play values to learn the evaluation function or one can use a part of the actual data for training and the rest for testing (*e.g.*, using k-fold cross-validation, as we have done in our heuristic uncertainty experiments).

Note that, in using their data, we are not criticizing the application of AIVAT by Brown & Sandholm (2019); they were correct. We are using their dataset to demonstrate how invalid conclusions can be drawn due to heuristic pathologies, that is, not using the test data to tune the heuristic value function. While we are unaware of existing AIVAT applications that violate this, to our knowledge, we are the first to point out such vulnerabilities in the AIVAT family of techniques.

5 Conclusions and Future Research

We studied the roles played by the underspecified heuristic value function in the AIVAT family of variance reduction techniques. In our main contribution, we showed that the uncertainty of the heuristic value function outputs, if known, can be propagated to find the uncertainty of AIVAT estimates. Our contribution on heuristic uncertainty is particularly notable in that it leverages extra-game-theoretical considerations to achieve additional variance reduction on top of AIVAT. We demonstrated that IVW further reduces the variance and yields a 43.0% reduction in the number of samples required to draw statistical conclusions. While this may come at the cost of possible bias, we can estimate this bias, and fixing the learning algorithm for the heuristic value function can yield stronger theoretical guarantees. In our secondary contribution, we developed a heuristic value function to pathologically lower the variance and draw desired statistical conclusions about an agent's performance. Our results showed that, when the data is fixed, an adversary can craft a heuristic value function that can lead to drawing nonsensical, misleading, or contradictory statistical conclusions. Thus, the heuristic value function should be fixed *prior to evaluation*!

A possible avenue of future work is developing methods to quantify the uncertainty of self-play values approximated using stochastic game-solvers, which can then be used by AIVAT with IVW.

AI USE STATEMENT

In this paper, generative AI tools were not used for any of the tasks with required disclosure, such as generating datasets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or creating software code, designing or providing feedback on research methodology or experiments, implementing methods, assisting with translation, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, and interpreting results. For tasks with recommended disclosure, we have used generative AI tools for analyzing existing/relevant literature and searching for information. In general, generative AI did not play any role in the paper write-up, code creation, or figure/table generation.

REFERENCES

Nolan Bard, John Hawkin, Jonathan Rubin, and Martin Zinkevich. The annual computer poker competition. *AI Magazine*, 34(2):112–114, 2013.

Darse Billings and Morgan Kan. A tool for the direct assessment of poker decisions. *ICGA Journal*, 29(3):119–142, 2006.

Michael Bowling, Michael Johanson, Neil Burch, and Duane Szafron. Strategy evaluation in extensive games with importance sampling. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2008.

Noam Brown and Tuomas Sandholm. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418–424, 2018.

Noam Brown and Tuomas Sandholm. Superhuman AI for multiplayer poker. *Science*, 365(6456): 885–890, 2019.

Neil Burch, Martin Schmid, Matej Moravcik, Dustin Morill, and Michael Bowling. AIVAT: A new variance reduction technique for agent evaluation in imperfect information games. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

Paul Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, 2004.

Joachim Hartung, Guido Knapp, and Bimal K. Sinha. *Statistical Meta-Analysis with Applications*. John Wiley & Sons, 2011.

James W. Kirchner. Data analysis toolkit #12: Weighted averages and their uncertainties, 2006.

Boning Li, Yu Chen, and Longbo Huang. AV-AIVAT: 74x cheaper agent evaluation with certified anytime-valid stopping in imperfect-information games, 2026.

David J. C. MacKay. Bayesian non-linear modeling for the prediction competition. In *Proceedings of the International Workshop on Maximum Entropy and Bayesian Methods of Statistical Analysis*, 1996.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. DeepStack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

Oskari Tammelin. Solving large imperfect information games using CFR+, 2014.

Michael E. Tipping. Sparse Bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 1:211–244, 2001.

Martha White and Michael Bowling. Learning a value analysis tool for agent evaluation. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2009.

Christopher K. I. Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. MIT Press, 2006.

Martin Zinkevich, Michael Bowling, Nolan Bard, Morgan Kan, and Darse Billings. Optimal unbiased estimators for evaluating agent performance. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2006.

A VISUALIZATION OF AIVAT

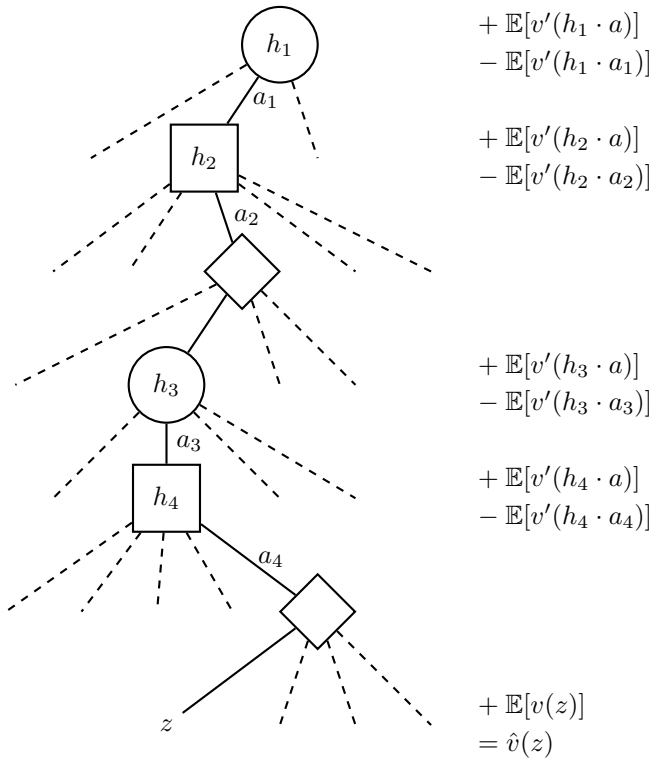


Figure 1: A diagram representing AIVAT. Circles, squares, and diamonds denote chance nodes, player-1 nodes, and player-2 nodes, respectively. This diagram illustrates a situation where only the action probabilities of the nodes belonging to nature and Player 1 are known. The right side of the diagram shows the terms involved in the calculation of the AIVAT estimate. Here, $U(h) = \{h\}$ for the sake of simplicity.

A simple visualization of AIVAT is given in Figure 1.

B SIMPLIFYING THE EXPRESSION FOR AIVAT

We simplified the expression for the AIVAT estimate $\hat{v}(\cdot)$ by rewriting it as an affine function of the outputs of the value function $v'(\cdot)$ in Section 4. The detailed derivation process is given below:

$$
\begin{aligned}
\hat{v}(z) &= \hat{v}_b(z) + \hat{v}_c(z) \\
&= \frac{\sum_{z' \in U(z)} \pi(z') v(z')}{\sum_{z' \in U(z)} \pi(z')} + \sum_{h \cdot a \in K(z)} \left(\frac{\sum_{a' \in A(U(h))} \sum_{h' \in U(h)} \pi(h' \cdot a') v'(h' \cdot a')}{\sum_{h' \in U(h)} \pi(h')} \right. \\
&\quad \left. - \frac{\sum_{h' \in U(h)} \pi(h' \cdot a) v'(h' \cdot a)}{\sum_{h' \in U(h)} \pi(h' \cdot a)} \right) \\
&= \sum_{z' \in U(z)} \frac{\pi(z') v(z')}{\sum_{z'' \in U(z)} \pi(z'')} \\
&\quad + \sum_{h \cdot a \in K(z)} \sum_{a' \in A(U(h))} \sum_{h' \in U(h)} \frac{\pi(h' \cdot a')}{\sum_{h'' \in U(h)} \pi(h'')} v'(h' \cdot a') \\
&\quad + \sum_{h \cdot a \in K(z)} \sum_{h' \in U(h)} \left(- \frac{\pi(h' \cdot a)}{\sum_{h'' \in U(h)} \pi(h'' \cdot a)} \right) v'(h' \cdot a).
\end{aligned}
$$

For the sake of simplicity, define the affine shift $b(z)$,

$$b(z) = \sum_{z' \in U(z)} \frac{\pi(z')v(z')}{\sum_{z'' \in U(z)} \pi(z'')},$$

the sets $S(z)$ and $S'(z)$ of H (note that $S'(z) \subseteq S(z)$),

$S(z) = \{h' \cdot a' : \exists h \cdot a \in K(z), a' \in A(U(h)), h' \in U(h)\}, \quad S'(z) = \{h' \cdot a : \exists h \cdot a \in K(z), h' \in U(h)\}$,

and $\mathbf{c}(z) \in \mathbb{R}^H$, the vector of coefficients of each $v'(\cdot)$. Explicitly, for each $h \cdot a \in S(z) \setminus S'(z)$,

$$\mathbf{c}(z)_{h \cdot a} = \frac{\pi(h \cdot a)}{\sum_{h' \in U(h)} \pi(h')},$$

for each $h \cdot a \in S'(z)$,

$$\mathbf{c}(z)_{h \cdot a} = \frac{\pi(h \cdot a)}{\sum_{h' \in U(h)} \pi(h')} - \frac{\pi(h \cdot a)}{\sum_{h' \in U(h)} \pi(h' \cdot a)}, \tag{10}$$

and finally for the remaining $h \notin S(z)$,

$$\mathbf{c}(z)_h = 0.$$

Putting it all together, we obtain the following simplified expression:

$$\hat{v}(z) = b(z) + \sum_{h \in H} \mathbf{c}(z)_h v'(h). \tag{11}$$

C Detailed refinement of the optimization problem for sample variance

The detailed derivation process for the refinement of the proxy objective of the optimization problem given by White & Bowling (2009) in (2) is as follows:

$$\begin{aligned}
\textbf{Minimize: } C(\boldsymbol{\theta}) &= \sum_{t=1}^{T} \left(\hat{v}_{\boldsymbol{\theta}}(z_t) - \frac{1}{T} \sum_{t'=1}^{T} \hat{v}_{\boldsymbol{\theta}}(z_{t'}) \right)^2 \\
&= \sum_{t=1}^{T} \left(b(z_t) + \langle \mathbf{c}(z_t), \boldsymbol{\theta} \rangle - \frac{1}{T} \sum_{t'=1}^{T} \left(b(z_{t'}) + \langle \mathbf{c}(z_{t'}), \boldsymbol{\theta} \rangle \right) \right)^2 \\
&= \sum_{t=1}^{T} \left(b(z_t) + \langle \mathbf{c}(z_t), \boldsymbol{\theta} \rangle - \left(\bar{b} + \langle \bar{\mathbf{c}}, \boldsymbol{\theta} \rangle \right) \right)^2 \\
&= \sum_{t=1}^{T} \left(\left(b(z_t) - \bar{b} \right) + \langle \mathbf{c}(z_t) - \bar{\mathbf{c}}, \boldsymbol{\theta} \rangle \right)^2,
\end{aligned}$$

where $\bar{b} = \frac{1}{T} \sum_{t=1}^{T} b(z_t)$ and $\bar{\mathbf{c}} = \frac{1}{T} \sum_{t=1}^{T} \mathbf{c}(z_t)$.

D Omitted proofs

This section contains several proofs that were omitted in the main body of the paper.

D.1 Proof of Proposition 3

Proof. The cost function $C(\boldsymbol{\theta}) = \sum_{t=1}^{T} \left(\left(b(z_t) - \bar{b} \right) + \langle \mathbf{c}(z_t) - \bar{\mathbf{c}}, \boldsymbol{\theta} \rangle \right)^2$ can be rewritten in the standard least-squares form as follows: $C(\boldsymbol{\theta}) = \|\boldsymbol{y} + \boldsymbol{X}\boldsymbol{\theta}\|_2^2$, where $\boldsymbol{X} = \left[(\mathbf{c}(z_t) - \bar{\mathbf{c}})^\top \right]_{t \in \{1, \ldots, T\}} \in \mathbb{R}^{T \times H}$ and $\boldsymbol{y} = \left[b(z_t) - \bar{b} \right]_{t \in \{t, \ldots, T\}} \in \mathbb{R}^T$. When expanded, we have $C(\boldsymbol{\theta}) = \boldsymbol{y}^\top \boldsymbol{y} + 2\boldsymbol{y}^\top \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{\theta}^\top \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\theta}$. Since $\boldsymbol{X}^\top \boldsymbol{X}$ is positive semidefinite, we have that $C(\boldsymbol{\theta})$ is convex. Therefore, it has a global minimum, and there exists a vector $\boldsymbol{\theta}^* \in \mathbb{R}^H$ that reaches it, as required. □

D.2 Proof of Proposition 1

Proof. Let $\hat{v}(z_1), \ldots, \hat{v}(z_T)$ be independent unbiased AIVAT estimates with variances $\mathrm{Var}(\hat{v}(z_1)), \ldots, \mathrm{Var}(\hat{v}(z_T))$. The weighted average $\bar{v}^* = \sum_{t=1}^T w_t^* \hat{v}(z_t)$ has variance $\mathrm{Var}(\bar{v}^*) = \sum_{t=1}^T (w_t^*)^2 \mathrm{Var}(\hat{v}(z_t))$. It is a classical result that this variance can be minimized by setting each weight w_t to be proportional to $\frac{1}{\mathrm{Var}(\hat{v}(z_t))}$; cf. Hartung et al. (2011, Ch. 4). □

D.3 Proof of Proposition 2

Proof. We have that

$$\begin{aligned}
\mathbb{E}[\bar{v}^*|\sigma] &= \mathbb{E}\left[\sum_{t=1}^T w_t^* \hat{v}(z_t) \middle| \sigma\right] \\
&= \sum_{t=1}^T \mathbb{E}[w_t^* \hat{v}(z_t)|\sigma] \\
&= \sum_{t=1}^T (\mathbb{E}[w_t^*|\sigma]\mathbb{E}[\hat{v}(z_t)|\sigma] + \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma)) \\
&= \sum_{t=1}^T \mathbb{E}[w_t^*|\sigma]\mathbb{E}[\hat{v}(z_t)|\sigma] + \sum_{t=1}^T \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma) \\
&= \mathbb{E}_{z\in Z}[v(z)|\sigma] \sum_{t=1}^T \mathbb{E}[w_t^*|\sigma] + \sum_{t=1}^T \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma) \\
&= \mathbb{E}_{z\in Z}[v(z)|\sigma]\mathbb{E}\left[\sum_{t=1}^T w_t^* \middle| \sigma\right] + \sum_{t=1}^T \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma).
\end{aligned}$$

We assume normalized weights, so $\mathbb{E}[\bar{v}^*|\sigma] = \mathbb{E}_{z\in Z}[v(z)|\sigma] + \sum_{t=1}^T \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma)$. If $\sum_{t=1}^T \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma) = 0$, then $\mathbb{E}[\bar{v}^*|\sigma] = \mathbb{E}_{z\in Z}[v(z)|\sigma]$, as required. And if $\sum_{t=1}^T \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma) \neq 0$, then $\mathbb{E}[\bar{v}^*|\sigma] \neq \mathbb{E}_{z\in Z}[v(z)|\sigma]$, as required. Let $S = \sum_{t=1}^T w_t$ and let w and $v(z)$ be random variables for w_t and $v(z_t)$. Since we assume the data points are i.i.d., we have that $\mathbb{E}[w_t] = \mathbb{E}[w]$, $\mathbb{E}[v(z_t)] = \mathbb{E}[v(z)]$, and $\mathbb{E}[S] = T\mathbb{E}[w]$. Then, we have that

$$\begin{aligned}
\text{bias} &= \sum_{t=1}^T \mathrm{Cov}(w_t^*, \hat{v}(z_t)|\sigma) \\
&= \sum_{t=1}^T \mathrm{Cov}\left(\frac{w_t}{S}, \hat{v}(z_t)\middle|\sigma\right) \\
&\approx \sum_{t=1}^T \left(\frac{1}{\mathbb{E}[S]}\mathrm{Cov}(w_t, \hat{v}(z_t)|\sigma) - \frac{\mathbb{E}[w_t]}{\mathbb{E}[S]^2}\mathrm{Cov}(S, \hat{v}(z_t)|\sigma)\right) \\
&= \frac{T}{\mathbb{E}[S]}\mathrm{Cov}(w, \hat{v}(z)|\sigma) - \frac{T\mathbb{E}[w]}{\mathbb{E}[S]^2}\mathrm{Cov}(S, \hat{v}(z)|\sigma) \\
&= \frac{T}{T\mathbb{E}[w]}\mathrm{Cov}(w, \hat{v}(z)|\sigma) - \frac{T\mathbb{E}[w]}{(T\mathbb{E}[w])^2}\mathrm{Cov}(S, \hat{v}(z)|\sigma) \\
&= \frac{1}{\mathbb{E}[w]}\mathrm{Cov}(w, \hat{v}(z)|\sigma) - \frac{1}{T\mathbb{E}[w]}\mathrm{Cov}(S, \hat{v}(z)|\sigma) \\
&\approx \frac{\mathrm{Cov}(w, \hat{v}(z)|\sigma)}{\mathbb{E}[w]},
\end{aligned}$$

as required. □

E RULES OF TEXAS HOLD'EM

In Texas hold'em, the game begins with every player being dealt two private cards. This is followed by a preflop betting round, where players take turns betting, matching (*i.e.*, checking or calling), or giving up (*i.e.*, folding) based on their hidden cards and beliefs about what cards others hold. After the betting round, the dealer reveals three public cards as part of the flop, which is then followed by another betting round. After that, a single card is revealed as the turn card, and another betting round ensues. Finally, the river card is revealed publicly, and the final betting round takes place, after which the game terminates. The game can also end when only one player remains in the pot (as others have folded). In no-limit settings, players are allowed to bet or raise whatever amount they want (capped by their number of chips) however many times they want, whereas this amount and the maximum number of bets are determined by the rules of the game in fixed-limit settings.

F FEATURE ENGINEERING OF STATES IN 6-MAX NO-LIMIT TEXAS HOLD'EM

For the feature engineering of states in 6-max no-limit Texas hold'em, we took an approach similar to that of White & Bowling (2009) in their evaluation of 6-player limit Texas hold'em data. Specifically, we extracted the pot amount and each player's hand strength and hand strength squared values. The hand strength is the expected probability of beating another random hand, while the hand strength squared is the expected squared probability of beating a random hand. All hand strength (squared) values were also multiplied by the pot amount and exponentiated by the number of non-folded players.

G STANDARD ERROR OF THE (WEIGHTED) MEAN

In this section, we provide the definition of the standard error of the (weighted) mean. When unweighted, the standard definition in the field of statistics is

$$SE = \frac{s}{\sqrt{N}},$$

where s is the sample standard deviation and N is the sample size. Note that

$$s^2 = \frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N-1},$$

where $x_1, \ldots, x_N$ are the sample values and $\bar{x}$ is the sample mean. Under inverse-variance weighting, Kirchner (2006) gives the following definition:

$$SE^* = \frac{s^*}{\sqrt{N}},$$

where s^* is the weighted sample standard deviation and $w_1, \ldots, w_N$ are the weights corresponding to the sample values $x_1, \ldots, x_N$, respectively. Here,

$$(s^*)^2 = \left(\frac{\sum_{i=1}^{N} w_i(x_i - \bar{x}^*)^2}{\sum_{i=1}^{N} w_i}\right)\left(\frac{N}{N-1}\right),$$

where $\bar{x}^* = \frac{\sum_{i=1}^{N} w_i x_i}{\sum_{i=1}^{N} w_i}$ is the weighted sample mean.

H ADDITIONAL EXPERIMENTAL RESULTS ON HEURISTIC UNCERTAINTY

We continue our experiments on heuristic uncertainty. In learning the heuristic value function for our MIVAT estimator, we chose two more models that allow the calculation of output variance: *Bayesian ridge (BR)* (Tipping, 2001) and *automatic relevance determination (ARD) regression* (MacKay, 1996). We refer to the two new MIVAT estimators as follows: MIVAT-BR and MIVAT-ARDR, respectively. Due to the inherent limitation of the models, we assumed for the two estimators that the heuristic value outputs are uncorrelated, *i.e.*, (4) simplifies to (5). This limitation is why we

Table 4: Results in milli-big blinds per hand (mbb/h). SE stands for the standard error of the (weighted) mean, and 'Est.' is an abbreviation for 'Estimated'. Our results should not be compared with results (starred) of Brown & Sandholm (2019), as we lacked access to Pluribus.

Estimator	Weighting	Win rate	SE	Est. bias
MIVAT-BR	Uniform	-80	83	–
	IVW (ours)	0	**74**	80
MIVAT-ARDR	Uniform	-80	83	–
	IVW (ours)	0	**74**	80
MIVAT-GPR	Uniform	-25	99	–
	IVW (ours)	-22	**75**	3
Raw	Uniform	-70	88	–
MIVAT-WB	Uniform	-98	85	–
AIVAT Brown & Sandholm (2019)	Uniform	48	*25**	–

relegated the use of these models to the appendix. Except for subsampling, the training procedure for both estimators was identical to that of MIVAT-GPR.

Table 4 shows the performance of both MIVAT-BR and MIVAT-ARDR, in addition to MIVAT-GPR and our baselines. When taking the simple mean, both MIVAT-BR and MIVAT-ARDR outperform MIVAT-GPR, but this is understandable, as we used far less data to train GPR due to computational constraints. IVW allowed our MIVAT estimators to yield the best performance (save for AIVAT). The difference in the uncertainty achieved between uniform averaging and IVW averaging is smaller for MIVAT-BR and MIVAT-ARDR than for MIVAT-GPR. In general, for all our MIVAT estimators under IVW, the bias is approximately equal to the difference between the estimates under different weighting schemes. Indeed, offsetting the IVW estimates using the estimated bias recovers an estimator that is in line with using uniform averaging, which is unbiased. Also, the IVW win rates for MIVAT-BR and MIVAT-ARDR are not exactly zero but are shown so purely due to rounding coincidence. Before rounding, the values were approximately -0.233 for MIVAT-BR and -0.217 for MIVAT-ARD. Note that the values for MIVAT-BR and MIVAT-ARDR are similar because both BR and ARD are Bayesian linear regression models.

In addition to using the Pluribus data, we also conducted experiments on the synthetic data of 10,000 near-optimal gameplays in variants of (simultaneous) Goofspiel with five, six, and seven cards. For each Goofspiel variant, we generated 5,000 extra gameplays to train the GPR value function. Unlike for Pluribus, our implementation of AIVAT for Goofspiel does **not** reduce to MIVAT, and we denote the AIVAT estimator using the GPR value function as AIVAT-GPR.

Table 5 shows the performance of our MIVAT-GPR estimators for the Goofspiel data. In every case, there is a noticeable reduction in the standard error achieved by AIVAT with IVW compared to both when vanilla AIVAT is used and when AIVAT is not used at all. Additionally, the bias induced by IVW is small enough for our AIVAT estimator to be helpful, *i.e.*, it reduces the mean squared error. This is additional evidence that IVW can help achieve a low-variance estimate of expected player utilities in games, even when heuristics are poor. We have also demonstrated that our results on further variance reduction via uncertainty propagation holds outside the domain of poker.

I Linear value function

Define a feature engineering function $\boldsymbol{\phi} : H \mapsto \mathbb{R}^d$ that maps a state h to a respective vector of features. When linear, we require the value function to be of the following form:

$$v'_{\boldsymbol{\theta}}(h) = \boldsymbol{\phi}(h)^\top \boldsymbol{\theta}, \tag{12}$$

where $\boldsymbol{\theta} \in \mathbb{R}^d$ is the parameter vector. Then,

$$\hat{v}_{\boldsymbol{\theta}}(z) = b(z) + \sum_{h \in H} \mathbf{c}(z)_h v'_{\boldsymbol{\theta}}(h) = b(z) + \sum_{h \in H} \mathbf{c}(z)_h \boldsymbol{\phi}(h)^\top \boldsymbol{\theta}$$

Table 5: Results on the synthetic Goofspiel data. SE stands for the standard error of the (weighted) mean, and 'Est.' is an abbreviation for 'Estimated'.

Game	Estimator	Weighting	Win rate	SE	Est. bias
Goofspiel-5	AIVAT-GPR	Uniform	-5.23×10^{-3}	7.47×10^{-3}	–
		IVW (ours)	-2.57×10^{-3}	$\mathbf{1.62 \times 10^{-3}}$	-2.65×10^{-3}
	Raw	Uniform	1.90×10^{-3}	6.04×10^{-3}	–
Goofspiel-6	AIVAT-GPR	Uniform	-9.57×10^{-3}	9.07×10^{-3}	–
		IVW (ours)	-3.90×10^{-3}	$\mathbf{3.64 \times 10^{-3}}$	-5.68×10^{-3}
	Raw	Uniform	2.80×10^{-3}	7.66×10^{-3}	–
Goofspiel-7	AIVAT-GPR	Uniform	-5.83×10^{-3}	1.03×10^{-2}	–
		IVW (ours)	-5.05×10^{-3}	$\mathbf{5.32 \times 10^{-3}}$	-7.86×10^{-4}
	Raw	Uniform	8.90×10^{-3}	8.45×10^{-3}	–

$$= b(z) + \left(\sum_{h \in H} \mathbf{c}(z)_h \boldsymbol{\phi}(h) \right)^\top \boldsymbol{\theta} = b(z) + \boldsymbol{\psi}(z)^\top \boldsymbol{\theta},$$

where $\boldsymbol{\psi}(z) = \sum_{h \in H} \mathbf{c}(z)_h \boldsymbol{\phi}(h)$.

I.1 Closed-form solution

For MIVAT, White & Bowling (2009) derived a closed-form formula for the optimal linear function parameter $\boldsymbol{\theta}^*$ for the optimization problem shown in (2). The steps they provide can be closely followed to yield an analogous solution for AIVAT. With $\bar{b} = \frac{1}{T} \sum_{t=1} b(z_t)$ and $\overline{\boldsymbol{\psi}} = \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\psi}(z_t)$, the parameter of the linear function that optimizes (2) is as follows:[2,3]

$$\boldsymbol{\theta}^* = \left(\left(\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\psi}(z_t) \boldsymbol{\psi}(z_t)^\top \right) - \overline{\boldsymbol{\psi}}\, \overline{\boldsymbol{\psi}}^\top \right)^{-1} \left(\bar{b}\, \overline{\boldsymbol{\psi}} - \left(\frac{1}{T} \sum_{t=1}^{T} b(z_t) \boldsymbol{\psi}(z_t) \right) \right). \tag{13}$$

We can simplify this further to

$$\boldsymbol{\theta}^* = \mathbf{X}^{-1} \mathbf{y}, \tag{14}$$

where $\mathbf{X} = \left(\left(\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\psi}(z_t) \boldsymbol{\psi}(z_t)^\top \right) - \overline{\boldsymbol{\psi}}\, \overline{\boldsymbol{\psi}}^\top \right)$ and $\mathbf{y} = \left(\bar{b}\, \overline{\boldsymbol{\psi}} - \left(\frac{1}{T} \sum_{t=1}^{T} b(z_t) \boldsymbol{\psi}(z_t) \right) \right)$. Although left unstated by White & Bowling (2009), for the above closed-form solution to exist, $\mathbf{X}$ must be invertible. We provide the conditions required for this to be true.

Theorem 1. *If no hyperplane in $\mathbb{R}^d$ contains all of $\{\boldsymbol{\psi}(z_t)\}_{t=1}^T$, then $\mathbf{X}$ is invertible.*

Proof. We prove the contrapositive: if $\mathbf{X}$ is non-invertible, then there is a hyperplane in $\mathbb{R}^d$ containing all of $\{\boldsymbol{\psi}(z_t)\}_{t=1}^T$.

If $\mathbf{X}$ is singular, then $\exists \mathbf{a} \in \mathbb{R}^d$ such that $\mathbf{a} \neq \mathbf{0}$ and $\mathbf{X}\mathbf{a} = \mathbf{0}$. The latter implies $\mathbf{a}^\top \mathbf{X} \mathbf{a} = 0$. Using

$$\mathbf{X} = \left(\left(\frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\psi}(z_t) \boldsymbol{\psi}(z_t)^\top \right) - \overline{\boldsymbol{\psi}}\, \overline{\boldsymbol{\psi}}^\top \right) = \frac{1}{T} \left(\sum_{t=1}^{T} (\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}})(\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}})^\top \right),$$

we obtain

$$0 = \mathbf{a}^\top \mathbf{X} \mathbf{a} = \mathbf{a}^\top \frac{1}{T} \left(\sum_{t=1}^{T} (\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}})(\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}})^\top \right) \mathbf{a} = \frac{1}{T} \sum_{t=1}^{T} ((\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}})^\top \mathbf{a})^2,$$

implying, $\forall t \in \{1, \ldots, T\} : (\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}})^\top \mathbf{a} = 0$, and hence $\boldsymbol{\psi}(z_t)^\top \mathbf{a} = \overline{\boldsymbol{\psi}}^\top \mathbf{a}$. Note that $\overline{\boldsymbol{\psi}}^\top \mathbf{a} = c$ is a constant and, defining $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{x}^\top \mathbf{a} = c$ is an equation of a hyperplane. □

[2] A careful reader may notice that the latter factor in (13) is negated in the equivalent formulation of White & Bowling (2009). This is because we hide the negative sign inside $\mathbf{c}(z)_{h' \cdot a}$; see (10).

[3] The steps we took to obtain this are isolated in Appendix J.

This prohibits one from applying common feature engineering techniques such as using separate linear coefficients for different stages in the game (*e.g.*, in poker: preflop, flop, turn, and river). On top of this, it lacks any regularization, which makes it prone to overfitting. It is also unclear how one can estimate the variance of its outputs.

J Derivation of the closed-form solution

This appendix section contains the steps to obtain the optimal linear value function parameter for the optimization problem given by White & Bowling (2009) optimization problem stated in (2). Note that this is for the case of the linear heuristic value function shown in (12), not the parameterized version in (8). Restating the optimization problem,

$$\begin{aligned}
\underset{\boldsymbol{\theta}\in\mathbb{R}^d}{\textbf{Minimize:}}\ C(\boldsymbol{\theta}) &= \sum_{t=1}^{T}\left(\hat{v}_{\boldsymbol{\theta}}(z_t) - \frac{1}{T}\sum_{t'=1}^{T}\hat{v}_{\boldsymbol{\theta}}(z_{t'})\right)^2 \\
&= \sum_{t=1}^{T}\left(\left(b(z_t) + \boldsymbol{\psi}(z_t)^\top\boldsymbol{\theta}\right) - \frac{1}{T}\sum_{t'=1}^{T}\left(b(z_{t'}) + \boldsymbol{\psi}(z_{t'})^\top\boldsymbol{\theta}\right)\right)^2 \\
&= \sum_{t=1}^{T}\left(\left(b(z_t) - \frac{1}{T}\sum_{t'=1}^{T}b(z_{t'})\right) + \left(\boldsymbol{\psi}(z_t) - \frac{1}{T}\sum_{t'=1}^{T}\boldsymbol{\psi}(z_{t'})\right)^\top\boldsymbol{\theta}\right)^2 \\
&= \sum_{t=1}^{T}\left(\left(b(z_t) - \bar{b}\right) + \left(\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}}\right)^\top\boldsymbol{\theta}\right)^2.
\end{aligned}$$

They then took the derivative of C with respect to $\boldsymbol{\theta}$ and set it to zero:

$$0 = \left.\frac{\partial C}{\partial\boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} = \sum_{t=1}^{T} 2\left(\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}}\right)\left(\left(b(z_t) - \bar{b}\right) + \left(\boldsymbol{\psi}(z_t) - \overline{\boldsymbol{\psi}}\right)^\top\boldsymbol{\theta}^*\right).$$

Simplifying and solving for $\boldsymbol{\theta}^*$, the following closed-form solution was obtained:

$$\boldsymbol{\theta}^* = \left(\left(\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{\psi}(z_t)\boldsymbol{\psi}(z_t)^\top\right) - \overline{\boldsymbol{\psi}}\,\overline{\boldsymbol{\psi}}^\top\right)^{-1}\left(\bar{b}\,\overline{\boldsymbol{\psi}} - \left(\frac{1}{T}\sum_{t=1}^{T}b(z_t)\boldsymbol{\psi}(z_t)\right)\right).$$

K Additional experimental results on heuristic pathology

We continue with our experiments on heuristic pathologies. First, the full results on heuristic pathologies optimizing the sample variance and the t-statistic are shown in Tables 6 and 7, respectively. As an extension to our second experiment, where we explored how to falsely make desired claims about the given data by p-hacking via gradient descent/ascent to either minimize or maximize the t-statistics, we also conducted experiments on the synthetic data of 10,000 near-optimal gameplays in variants of (simultaneous) Goofspiel with five, six, and seven cards. We trained these agents by running CFR$^+$ (Tammelin, 2014) for 1,000 iterations. Unlike for Pluribus, our implementation of AIVAT for Goofspiel did **not** reduce to MIVAT. Here, we again trained a parameterized heuristic value function following the procedure shown in the main content of the paper. The results are shown in Table 8. Even in the domain of Goofspiel, for each statistical conclusion, we were able to p-hack AIVAT into producing a t-statistic that corresponds to extremely low p-values. Thus, we have demonstrated that our pathology results hold outside the domain of poker.

L Testbench specification

Our testbench contains an AMD Ryzen 9 3900X 12-core, 24-thread desktop processor and 128 GB of memory.

Table 6: Results on heuristic pathologies minimizing the sample variance (in mbb/h). SE stands for the standard error of the mean.

Player	Win rate	SE
Pluribus	2062	25
Budd	2059	50
MrWhite	2043	37
MrOrange	2053	29
Hattori	2045	68
MrBlue	2062	26
MrPink	2048	32
Joe	2064	64
Bill	2061	30
MrBlonde	2040	49
Eddie	2049	34
MrBrown	2035	67
Gogo	2006	113
ORen	2034	90

Table 7: Results on heuristic pathologies optimizing the t-statistic.

Player	Losing		Winning	
	t-statistic	p-value	t-statistic	p-value
Pluribus	-95.050	$<10^{-1399}$	$+92.900$	$<10^{-1353}$
Budd	-49.538	$<10^{-373}$	$+55.012$	$<10^{-433}$
MrWhite	-76.612	$<10^{-816}$	$+75.638$	$<10^{-802}$
MrOrange	-95.291	$<10^{-1294}$	$+89.609$	$<10^{-1188}$
Hattori	-37.584	$<10^{-212}$	$+39.327$	$<10^{-226}$
MrBlue	-87.433	$<10^{-1207}$	$+93.654$	$<10^{-1336}$
MrPink	-69.132	$<10^{-767}$	$+68.549$	$<10^{-757}$
Joe	-44.570	$<10^{-278}$	$+42.285$	$<10^{-259}$
Bill	-70.778	$<10^{-814}$	$+70.991$	$<10^{-818}$
MrBlonde	-70.923	$<10^{-606}$	$+68.642$	$<10^{-582}$
Eddie	-66.739	$<10^{-710}$	$+69.562$	$<10^{-756}$
MrBrown	-42.189	$<10^{-249}$	$+44.325$	$<10^{-266}$
Gogo	-22.699	$<10^{-77}$	$+19.617$	$<10^{-63}$
ORen	-20.375	$<10^{-73}$	$+20.807$	$<10^{-76}$

Table 8: Results on the Goofspiel data using pathological heuristic value functions for each t-statistic. "Goofspiel-n" denotes the game of Goofspiel played with n cards.

Variant	P1 losing		P1 winning	
	t-statistic	p-value	t-statistic	p-value
Goofspiel-5	-15.429	$<10^{-52}$	$+15.394$	$<10^{-52}$
Goofspiel-6	-39.312	$<10^{-313}$	$+39.344$	$<10^{-314}$
Goofspiel-7	-82.843	$<10^{-1136}$	$+82.984$	$<10^{-1139}$