Does a Learning Agent Rediscover Optimal Play and Where Does It Fall Short?

A study of reinforcement learning's characteristic behavior, told through blackjack

Tabular Monte Carlo control • 5M training hands • audited cell-by-cell against basic strategy

Part of AI Journey — Deep Learning & RL

github.com/arda-basarici/blackjack-rl

The question behind the question

A reinforcement-learning agent that learns blackjack from nothing but the win or loss at the end of each hand will, given enough play, converge toward expert play. That much is unsurprising. The interesting question is not whether it succeeds but where it fails, and why — because the places a learned policy diverges from a known optimum are where the mechanics of learning-from-experience become visible. This report uses blackjack as an instrument: the agent is a way of watching how a value-based learner allocates its attention, where it grows confident, and where it stays blind.

Blackjack is the right instrument for a specific reason. Its optimal policy — basic strategy, the value-maximizing action in every situation — is exactly computable and long established. That gives the study something most machine-learning problems lack: a complete, trustworthy answer key. Every decision the agent makes can be checked against the known-correct one, so a disagreement is never a matter of opinion — it is the agent measurably falling short of a solved problem, which means each shortfall can be traced to a cause rather than waved away.

From a simulator to an environment

This work is the second movement of a longer project. The first built a Monte Carlo simulator of blackjack from scratch — an engine that deals, plays, and resolves hands under fixed rules, validated until its house edge and outcome distributions matched the known mathematics of the game. Here the same engine takes on a second role: what was a tool for generating data becomes an environment for learning. The agent plays through it to improve, updating its own estimates from the rewards the engine returns. The continuity is deliberate — the engine was already trusted, and the optimal strategy the first project derived is precisely what makes this project's central question answerable. A study of where a learner falls short is only meaningful if you already know, exactly, what not falling short looks like.

The method, in one box

The agent is a tabular Monte Carlo control learner: it plays complete hands, sees only the terminal win or loss, and from that single delayed signal updates an estimated value for every decision it made. Its policy is to take, in each situation, the action it currently values most.

The audit places that policy cell-by-cell against basic strategy, classifying each disagreement by whether the two actions genuinely differ in value, sit within a hair of each other, or rest on too little experience to trust. Crucially, the audit is computed from the agent's trained value table and its *training* visit counts — not from the separate simulations used to measure house edge — so a cell's classification is a property of the trained policy and does not shift if the edge evaluation is run at a different size.

1. The aggregate lies; the structure is in the conditioning

The pooled 92.5% hides the result; conditioning on decision type reveals it — error is localized, not spread.

Pooled over every situation, the agent agrees with basic strategy 92.5% of the time unweighted, and 94.0% weighted by how often each situation occurs. Taken alone, that number is true and nearly useless. It averages the decisions the agent has effectively mastered — stand on a hard twenty, hit a low total — together with the handful it genuinely struggles with. The structure appears the moment agreement is conditioned on the kind of decision: disagreement runs at 1.7% on hard totals, 12.0% on soft totals, and 27.1% wherever doubling is an option.

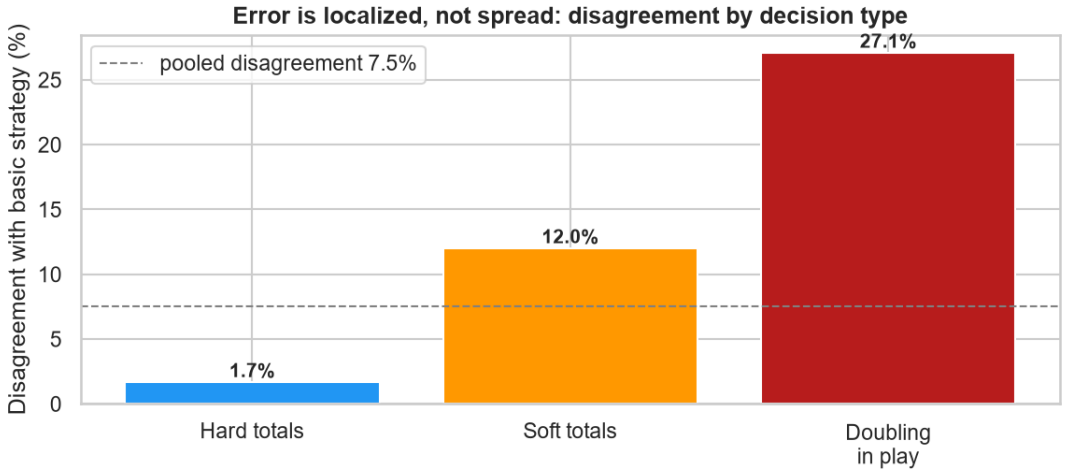

Figure 1: Disagreement with basic strategy, conditioned on decision type. The pooled average (dashed) hides a 16x spread between common and rare decisions.

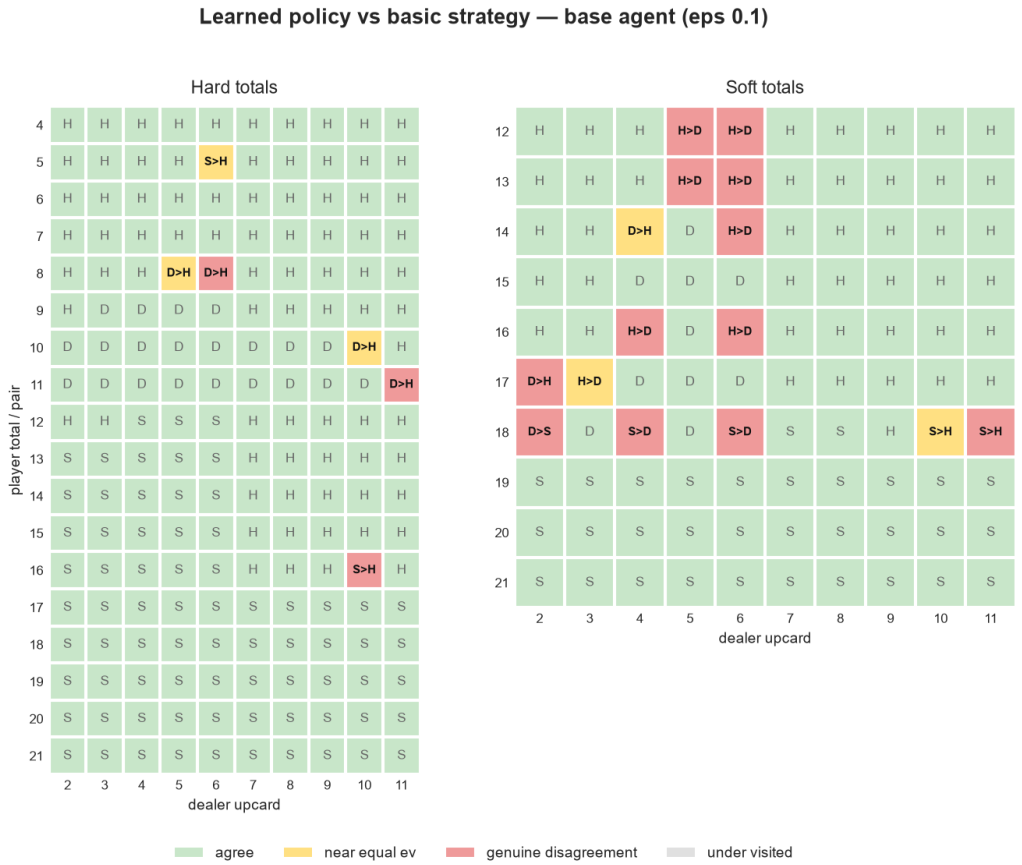

Figure 2: The learned base policy against basic strategy. Green agrees; yellow are near-ties; red are genuine disagreements. The red concentrates in the soft and doubling bands.

That spread is the first real result, and it is a general one wearing a blackjack costume. The agent has all but solved the common, frequently-revisited decisions and concentrated nearly all its error in the rare ones. This is the defining behavior of a learner whose knowledge is built from experience: what is seen often is learned well; what is seen seldom is learned poorly. A single headline metric is simply the wrong altitude from which to judge a learned policy — the informative view is always the conditioned one.

2. Reading the disagreements honestly: severity, not count

The disagreements span a twenty-fold range in severity; counting them as equal is the mistake a fixed threshold invites.

Having found that disagreements cluster, the natural next move is to count them — and that is the second place the obvious approach misleads. There are 15 genuine disagreements, but treating them as equivalent collapses two very different things: decisions where the agent's choice costs almost nothing, and decisions where it costs real value. The honest unit is severity — expected value forfeited — and measured that way the 15 span a twenty-fold range, from 0.021 to 0.407.

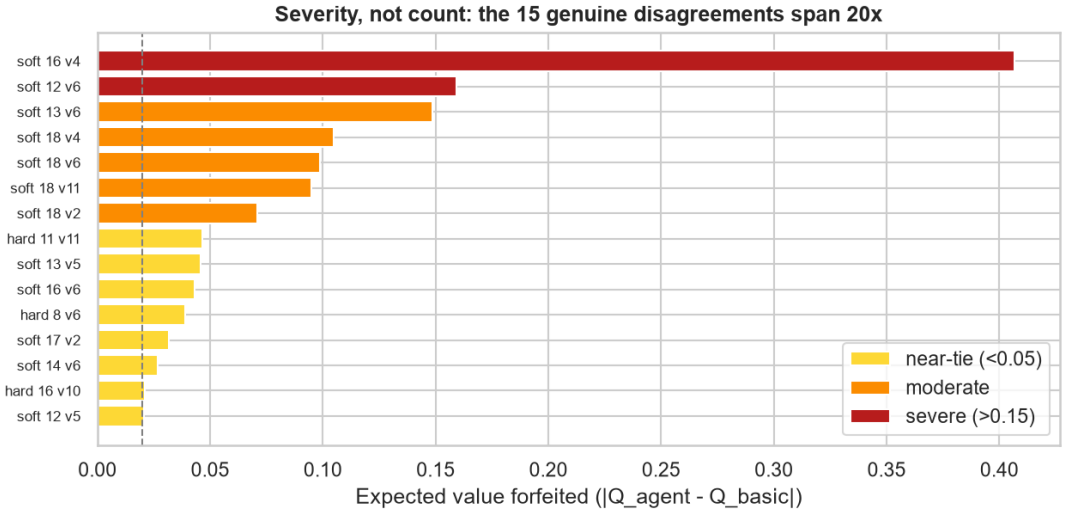

Figure 3: The genuine disagreements sorted by severity. A fixed 0.02 threshold (dashed) tips the most balanced decision in the game into the 'error' column on a single thousandth.

The most instructive case sits right at the boundary: hard sixteen against a dealer ten, the most famously balanced decision in the game, visited over 166,610 times, where the value of hitting and standing differ by only 0.021 — essentially a coin flip. A count leaning on a fixed cutoff labels this an error on a thousandth of a unit of value. Reporting the distribution dissolves the artifact, and surfaces a subtler point the next section tests: the largest gaps are not the agent confidently choosing wrong, but valuing an action it barely tried. The 0.407 outlier — soft sixteen against a four — is a case where the correct action was sampled in only about 3% of visits.

3. The mechanism: starvation, not stubbornness

The agent isn't wrong where it's confident — it's wrong where it barely looked.

A wrong choice can come from two very different places: the correct action was tried too rarely to be valued accurately (a coverage problem), or a wrong action was lifted by a few lucky early outcomes the

estimate could never shed (a memory problem). They are separable by a simple question: in the cells where the agent disagrees, how often did it actually try the action basic strategy recommends?

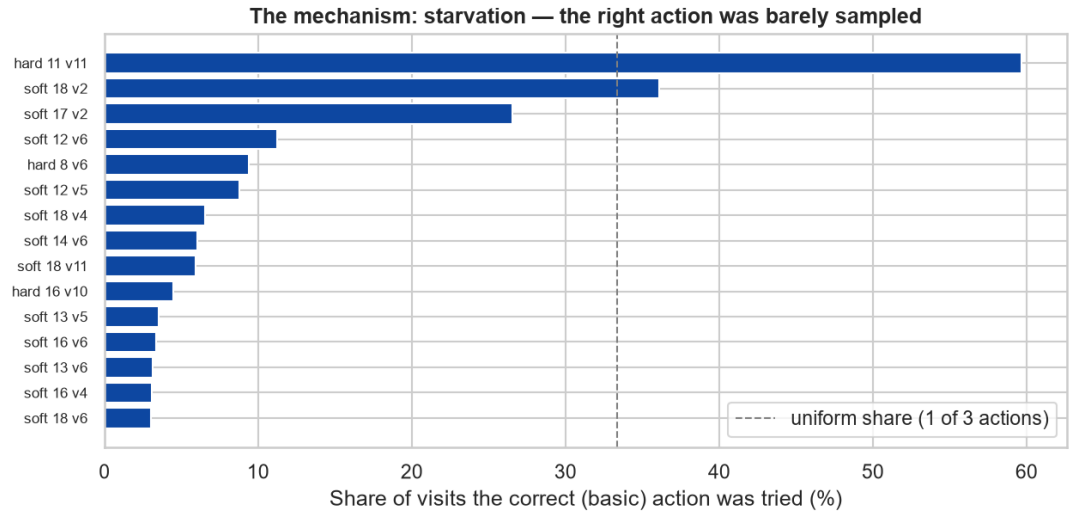

Figure 4: For each disagreement cell, the share of visits the correct action was tried. Almost everywhere it falls well below an even split — the agent never gathered the evidence to value it.

The answer is decisive. In nearly every disagreement the correct action was sampled only a small fraction of the time, so the agent never gathered the evidence to value it properly. This explains the localization with one stroke: under a policy that mostly exploits what it already believes, an action that looks unpromising early is tried less, which keeps its estimate pessimistic, which makes it look unpromising — a self-reinforcing starvation. The agent is not stubborn; it is uninformed, and uninformed in a structured way that traces directly back to how a value-based learner allocates its own experience. This is why exploration is not a tuning detail but the central design problem of this entire class of method.

4. More exploration relocates the error

Raising exploration corrects as many cells as it spoils: the error moves, it doesn't shrink.

If under-exploration causes the error, the intuitive fix is to explore more. The most counterintuitive result is that this does not work the way one expects. Raising the exploration rate from 0.1 to 0.3 does not shrink the disagreement; it moves it. Across the 280 shared situations, the higher rate corrects 11 decisions that were wrong and spoils 12 that were right — a net change of -1. The total is essentially unchanged; what shifts is which cells are wrong.

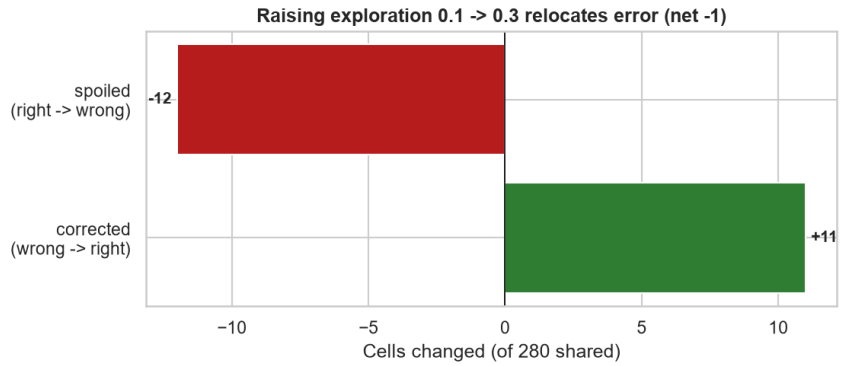

Figure 5: Raising exploration from 0.1 to 0.3, cell by cell. Coverage of the rare is bought with distortion of the common.

The reason is a genuine tension, not a flaw to be tuned away. The agent's value estimates reflect the policy it actually plays. Explore more, and the rare cells finally get coverage — but the common cells are

now evaluated under a policy that plays randomly thirty percent of the time, which biases their estimates downward. There is no fixed exploration rate that wins everywhere, because the very mechanism that informs the neglected decisions corrupts the well-understood ones. This is the on-policy learner's central bind in miniature.

5. Why the fix needs two parts: schedule and memory

Decaying exploration supplies the experience; constant-step memory keeps it from being anchored to a noisy start.

The resolution has two components. The first is obvious once the tension is named: explore heavily early, then decay exploration toward zero so the policy can settle into clean exploitation. But decaying alone is not enough. If each value is a simple running average over every outcome it ever saw, the noisy early returns are baked in permanently — the average cannot forget its own beginning. The cure is to weight recent experience more heavily, by updating with a constant step size rather than a true average, so each estimate tracks the improving policy instead of being anchored to its uninformed start.

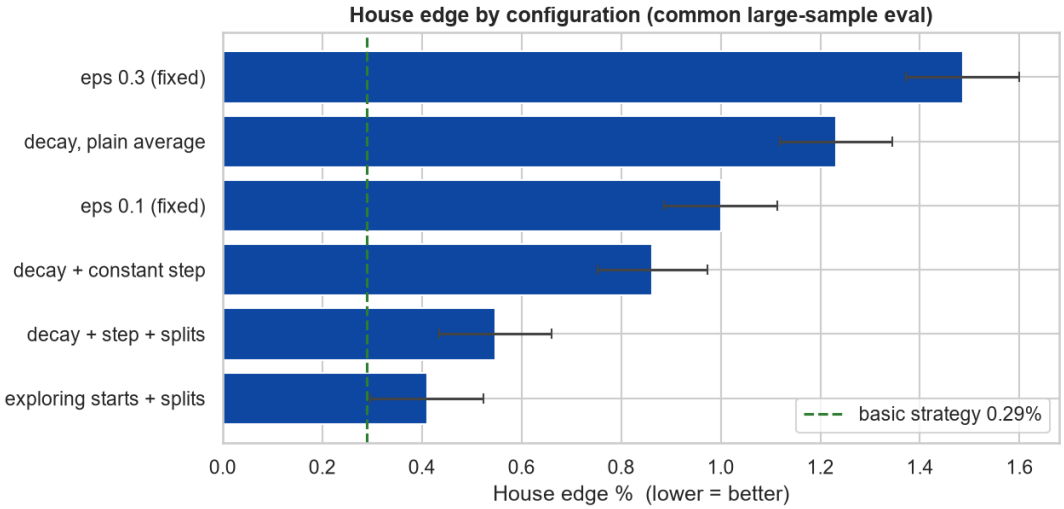

Figure 6: House edge by configuration, re-evaluated at a common large sample so the comparison is fair. Lower is better; the dashed line is basic strategy.

The effect is visible in the ledger of house edges. The decaying schedule with a constant step reaches roughly 0.86% against an optimal near 0.29%, while the same schedule under a plain running average sits at 1.23% and a high fixed exploration rate is worst at 1.49%; adding splits brings the best natural-play agent to 0.55%. Read against the measurement uncertainty of about ±0.11% on each estimate, the disciplined reading is narrow but real: recency-weighting and splits genuinely help, while the closest configurations differ by less than their error bars and cannot honestly be ranked against one another. That restraint is reinforced by a telling detail — at a smaller evaluation the top of the table inverted. The win was real only once the measurement was precise enough to see it.

6. The same story in a new register: splits

A new kind of decision, the same coverage signature — evidence the mechanism is general, not fitted.

Extending the agent to handle pairs is both a completion of the problem and a test of whether the mechanism is truly general. The agent is taught no rule; the situation is simply added to what it can perceive, and it must discover from experience whether and when to split. Adding splits improved the

edge, since correct splitting recovers value a no-split policy leaves on the table. The result is otherwise the same story in a new column: most splitting decisions are learned well; the errors concentrate in the rare low pairs — twos, threes, fours, and nines — each visited only around twenty-three hundred times, whose split action is tried far fewer times still. It is the identical coverage signature, now reproduced in a structurally different decision — and a mechanism that recurs across different decisions is more trustworthy than one fitted to a single case.

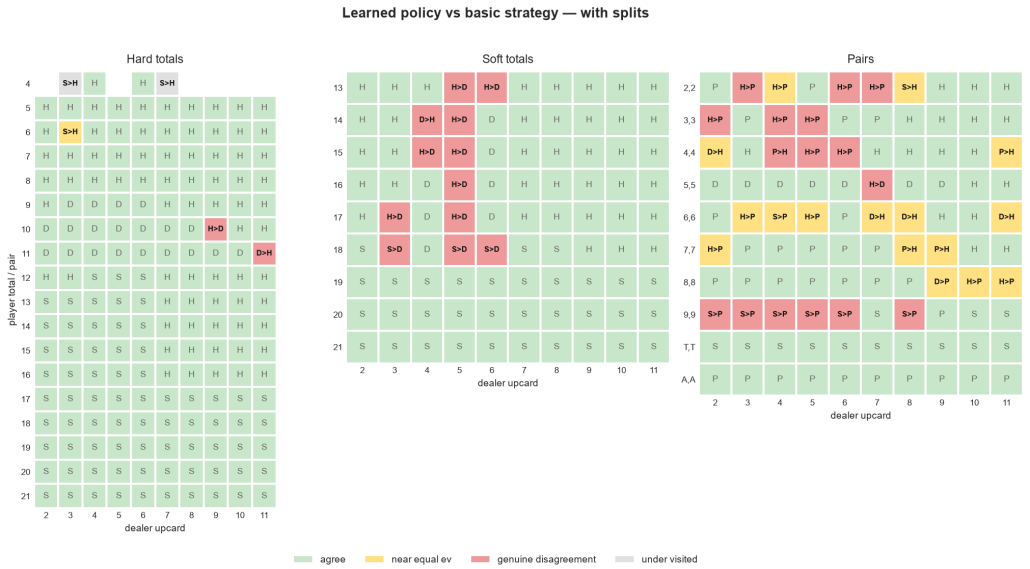

Figure 7: The policy with splits, before exploring starts. Most of the pairs column is learned (green); red marks the under-split rare low pairs the agent rarely experiences.

7. The capstone: making the explanation prove itself

A prediction locked before the run: force coverage, and the gap collapses to a floor of genuine ties.

Everything to this point argues that the residual error is, in the main, a coverage problem. An argument is not a demonstration, so the study closes with a test designed to be able to fail. If the residual is truly coverage, then forcing coverage should remove it. Exploring starts — the canonical Monte Carlo control variant — begins every hand from a deliberately chosen situation-and-action, giving every decision equal airtime regardless of how rarely natural play would reach it, then follows the learned policy. The prediction was fixed before the run: the coverage-driven disagreements should collapse toward optimal play, while the genuine near-ties should remain, since no amount of coverage can separate decisions that are truly tied. Had everything vanished, the coverage story would have been too clean to trust.

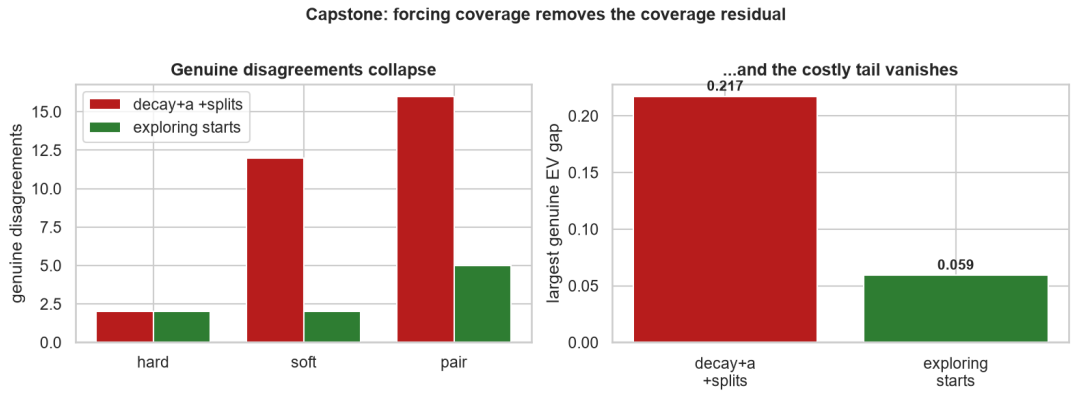

Figure 8: Forcing coverage collapses the genuine disagreements and erases the costly tail, leaving only near-ties.

The prediction held. The capstone runs against the splits agent, which carries 30 genuine disagreements; forcing coverage cut them to 9, and dropped the largest surviving gap from 0.217 to 0.059 — a tail of costly errors reduced to nothing but small ones. The cells that had been starved flipped to the correct action exactly where the mechanism said the agent had simply never looked: the rare low pairs that had been hit now split, the avoided doubles now double. What survived was precisely what was predicted to — 9 near-ties, each now backed by ample experience under forced starts (16,166–21,977 visits, far above their roughly twenty-three hundred in natural play) and still split down the middle because there is genuinely nothing to separate. The residual decomposes cleanly into two parts and only two: a reducible portion that was coverage, now collapsed, and an irreducible floor that is genuine indifference.

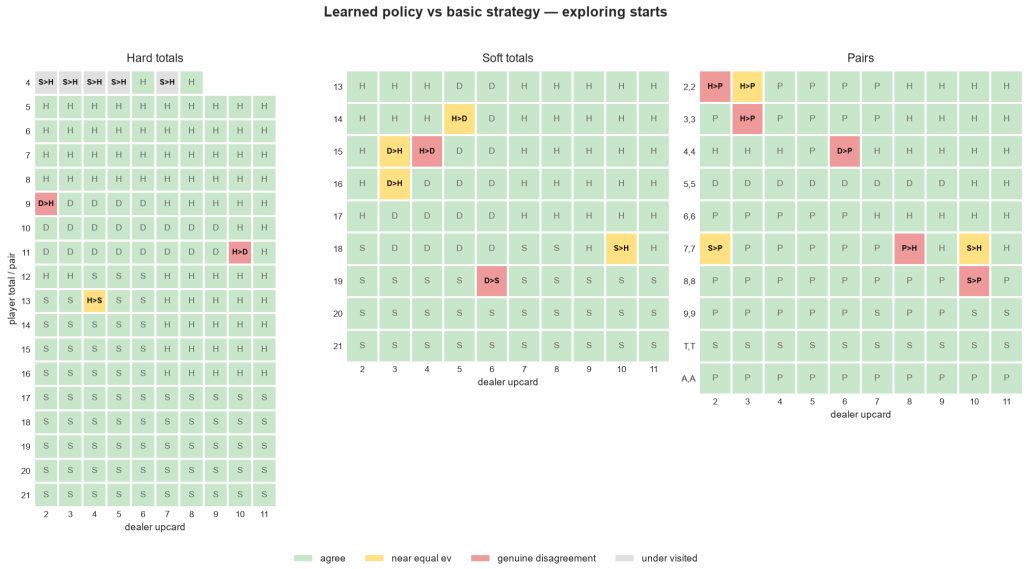

Figure 9: The exploring-starts policy. The red of Figure 2 is almost gone; what remains is yellow near-ties and a few thinly-visited deep states.

It is worth being honest about what the capstone did not do. Forcing coverage from two-card starting situations leaves the deepest, multi-card hands still thinly visited, so a small number of sparse states remain — 5, in this run — and coverage was therefore made near-uniform over decision points rather than over every reachable position. The experiment earned its claim; it did not manufacture a flawless sweep, and saying so is part of the same discipline that made the claim worth trusting.

What this study actually shows

Read as a whole, the report is not about blackjack and not about a particular agent. It is about a characteristic way that learning-from-experience behaves, made visible because a known optimum was available to measure against. A value-based learner masters what it sees often and stays blind where it looks seldom; its own policy shapes the data it learns from, so it can starve itself of the very experience it needs; naive attempts to fix this relocate the error rather than removing it, because coverage of the rare trades against accuracy on the common; the genuine fix pairs a decaying exploration schedule with memory that favors recent experience; and the residual that remains, once coverage is forced, is not failure but the irreducible floor of decisions that do not matter. None of these is specific to cards.

Where this goes next

The natural continuation moves from a setting where the optimum is known to one where it is not. Extending the agent toward card-counting and bet-sizing introduces a state the table can no longer cleanly

hold, and an objective that is no longer a single computable answer but a trade between expected value and the risk of ruin — a problem with no tidy ground-truth key. That shift is exactly where the disciplines practiced here — conditioning rather than pooling, severity over count, locked and falsifiable predictions, ranking only on distinguishable differences — stop being conveniences and become the only safeguards left. The value of having done this study against a solved problem is that those habits are now established before they are needed on an unsolved one.